



Class 6 CS545

Gregg Vesonder
Stevens Institute of Technology

©2020 Gregg Vesonder

1

Welcome to Class 6



Roadmap

- Log
- Ethics
- Design
- Interaction Styles
- Readings this class: Stone et.al., Chapters 10-13, Norman Chapters 3 & 6

©2020 Gregg Vesonder

2

Log Book

- “He or she that codes, decides!”

NOT!

- Remember this:
`asdfghjkl;' qwertyuiopzxcvbnmmmm`

©2020 Gregg Vesonder

There is an attitude that is very alive in developer circles that the person who codes makes the final decision on feature aspects including usability. In fact I have had coders bluntly tell me that they decide on the interface. This should NOT be the case (now you could predict that response). Notwithstanding the fact that it should not be the usability experts that decide the interaction (why? Hint user driven design). However the usability professional should insure that the coder is staying true to the guidelines, including the “non visible” ones. When I was doing usability work for command line interfaces my first test, which achieved some notoriety, was to rake my hand over the keyboard and hit return. The last line of this slide shows the result of doing this. For the first few weeks the entry would invariably cause the program to crash. What I was checking was making sure that they would do reasonable things during error conditions, accepting illegal input and buffering it while notifying the user that it was an incorrect response and perhaps displaying what the legal responses were.

Social Impact Statement

- Necessary for some systems, some countries
 - Potential to elevate fears or force designers to compromise, strong form in Scandinavian countries.
 - Produce it early in the development process, so that adjustments can be made
- Outline for the statement:
 - Describe new system, and benefits, identify stakeholders
 - Address concerns and potential barriers: changes in job function, layoffs, privacy issues, diverse access, ...
 - Outline development schedule and migration plans including training, environmental impact
- <http://heim.ifi.uio.no/~kristen/>, Kristen Nygaard, social implications

©2020 Gregg Vesonder

When doing user design and experimentation we have to understand what is legal and what is required legally and this varies per country. This is important if you want to market internationally— of course, however there are also moral issues involved with human experimentation and design.

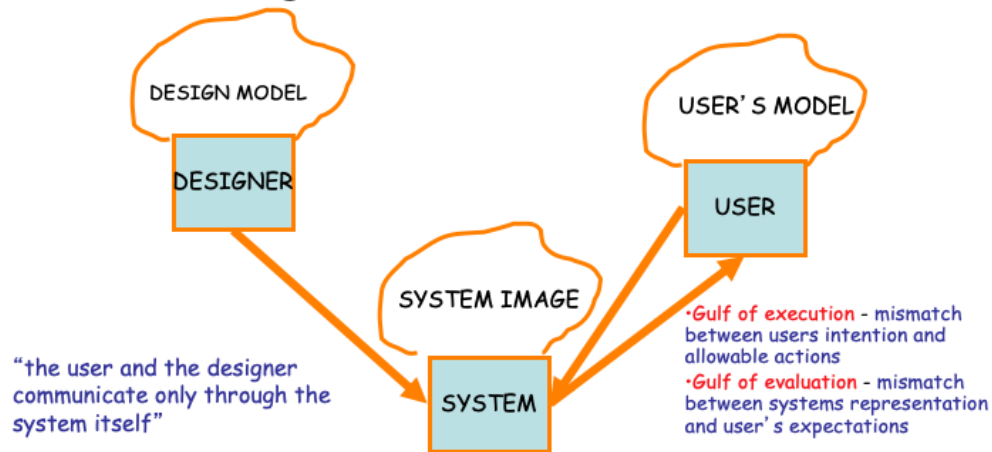
Milgram Experiment

It actually happened.

©2020 Gregg Vesonder

This occurred in 1963 at Yale. Hopefully it will be an aid to insure that you will be careful to respect your participants rights. Often showing extremes highlights the issues. It has been agreed that all HCI courses should cover ethics and experimentation. You can access the video in canvas.

Knowledge in the World and in the Head



The Design Challenge

©2020 Gregg Vesonder

So what devices do we use to convey information from the system to the user and then permit the user to interact with the system.

Interaction Styles

Style	Advantages	Disadvantages
Direct Manipulation	Visually presents task concepts, easy learning, easy retention, avoids errors, encourages exploration, high subjective satisfaction	Hard to develop, requires graphics display & pointing device
Menu Selection	Shortens learning, reduces keystrokes, structures decision making, can use dialog management tools, easy support of error handling	Danger of many menus, slows frequent users, consumes screen space, requires rapid display rate
Form Completion	Simple data entry, modest training, convenient assistance, use of form management tools	Consumes screen space
Command Language	Flexible, power users, user initiative, creation of macros (customizing)	Poor error handling, long training, memorization
Natural Language aka Anthropomorphic	Relieves burden of learning syntax	Clarification dialog, more keystrokes, contest is hard, unpredictable

©2020 Gregg Vesonder

I presented this table earlier in the year. These are the basic interaction styles in HCI we will review each of them in a bit more depth in this lecture and the next.

Direct Manipulation

- Shneiderman coined the concept
- Visibility of objects and actions of interest
- Rapid, reversible incremental actions
- Replacement of typed commands by a pointing action to the object of interest
- Enhancements:
 - Virtual reality - users are in an immersive environment in which the normal surroundings are blocked out by a head mounted display presenting artificial worlds
 - Augmented reality - keeps users in normal surroundings but adds a transparent overlay with information such as names of party goers
 - Tangible user interfaces - gives users physical objects to manipulate so as to operate the interface - drag and drop

©2020 Gregg Vesonder

The first we will cover is direct manipulation, which is slowly becoming more prevalent in gaming, in fact almost every gaming console has some means of motion capture..

Second Life



©2020 Gregg Vesonder

Second Life is a nice free version of a DM interface and is worth exploring. Note that it can be as nasty as the real world at times, so let the user be cautious. I would like to develop a software engineering space in Second Life or some other virtual reality space when I have time. The virtual reality space continues to morph with no consensus in sight. Oculus Rift certainly has created a stir - <http://www.oculusvr.com/>

User Aspects

- Mastery of the interface
 - Competence in performing tasks
 - Ease in learning originally and in assimilating advanced features
- Confidence in long term retention
- Fun
- Eagerness to show to novices
- Exploration

©2020 Gregg Vesonder

This is fairly clear – it is straightforward to learn the interface because it is “natural.”

Early Data

	Average task completion times	Average errors
Original Apple Macintosh	4.8 min	0.8
MS-DOS	5.8 min	2.0

©2020 Gregg Vesonder

This was an early study of the original Apple Macintosh's direct manipulation interface versus MS-DOS's command line interface, performing tasks such as creating, copying and renaming files. Note that Mac users did it faster and with less errors.

Principles

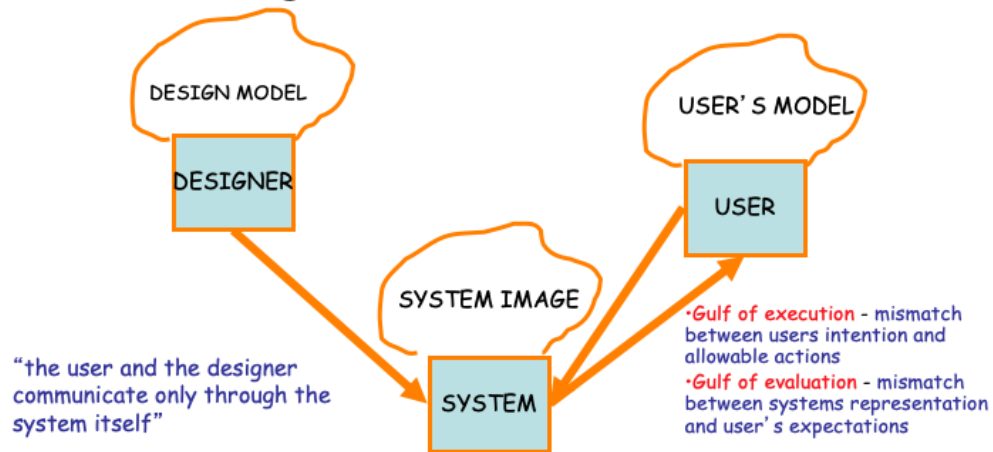
- **Principle of virtuality** - a representation of reality that can be manipulated
- **Principle of transparency** - the user is able to apply intellect directly to the task, the tool effectively disappears
- Bridges gulfs of execution and evaluation
- Gets even more interesting with Augmented Reality



Peak.ar

Some principles for using direct manipulation. It should be as natural as manipulating objects in the real world. It gets even more interesting with augmented reality interfaces where this direct manipulation intrudes with the real world.

Knowledge in the World and in the Head



The Design Challenge

©2020 Gregg Vesonder

Just to remind you 😊



Aspects of Direct Manipulation

- (principles)
- Continuous representation of objects and actions of interest with meaningful visual metaphors
- Physical actions or presses of labeled buttons, instead of complex syntax
- Rapid, incremental, reversible actions whose effects on the objects of interest are available immediately
- In this milieu error messages are rarely needed

©2020 Gregg Vesonder

Fairly straightforward – the last point is worth comment. With direct manipulation errors are evident – you drop or “break” objects or go down the wrong path. Error messages however can be used even in this milieu.

Enhanced 3D Interfaces

- Provide overviews (maps) so that user can see big picture
- Allow teleportation (point to point with animation)
- Provide history
- Enable remote collaboration
- Users have control over explanatory text and there are details on demand
- Enable landmarks to show from a distance
- Zooming
- Multiple coordinated views

©2020 Gregg Vesonder

More design points of such interfaces. You should have views that scale to the complete world. Again Second Life has done a great job of grappling with these issues and finding acceptable user interface solutions.

Teleoperation

- Direct manipulation in personal computers + process control (operator control physical processes in complex environments).
- Example is telemedicine
- Several issues:
 - Time delay
 - Operation delay
 - Incomplete feedback (haptic issues)
 - Unanticipated interferences (do not control all of the remote “stage”)



anybots.com

©2020 Gregg Vesonder

Another form of direct manipulation is directly manipulating devices through the computer. This is already done in medicine, astronomy and for the spirit and opportunity rovers on Mars. Of course there was the Sheldon bot, http://www.youtube.com/watch?v=_bloeBpSeU4 . Ignore the sub-titles

Enhancements to VR

- Head position sensing
- Hand position sensing - data glove
- Hand held manipulatives
- Force feedback and haptics (touch and force)
- Sound input and output - HDTV
- Other sensations tilting, ... Disney/Universal rides
- Collaborative and competitive environments

©2020 Gregg Vesonder

The title explains it all – of course 3D also has a place.



In Summary: DM Benefits

- Visual representation of the world of action, objects and actions are shown, analogical reasoning is tapped
- Rapid, incremental and reversible actions
- Replacement of typing with pointing and selecting
- Immediate visibility of results of actions

©2020 Gregg Vesonder

Some benefits of Direct Manipulation

In Summary: Benefits Over Commands

- Control/display compatibility
- Less syntax reduces error rates
- Errors are more preventable
- Faster learning and higher retention
- Encourages **exploration**
- **Games, social spaces, remote control**

©2020 Gregg Vesonder

More specifically in relation to command language interfaces. One negative is that they are often more verbose, take longer than command language equivalents.

Norman on Exploration

- Explorable systems are easier to learn and use, encourages the users to experiment and learn the possibilities
- Three requirements for explorable systems:
 - In each state of the system the user must readily see and be able to do the allowable action (sound familiar)
 - The effect of each action must be both visible and easy to interpret
 - **Actions should be without cost.** When an action has an undesirable result, it must be readily reversible or the action should be difficult to do, nonexplorable. However most actions should be cost- free, explorable, discoverable
- The point cannot be over stressed: make the computer system invisible

©2020 Gregg Vesonder

As you might suspect, Norman has a few opinions on these interfaces. Indeed the last point is that the systems should be immersive – in effect you are actually “in world.”

In Summary: DM Concerns

- Increased system resources
- Some actions may be cumbersome
- Macro techniques are often weak
- History and other tracing may be more difficult (replays)
- Visually impaired users may have more difficulty
- You will see more of it – Wii was really groundbreaking

©2020 Gregg Vesonder

Of course there are issues with DM. Mostly the benefits outweigh the issues and you will see much more of it in the future.

Menus, Form Completion, Dialog Boxes

- Our everyday world on the web
 - Information Architecture
- Many issues worthy of experimentation:
 - Depth versus **breadth - studies support it**
 - Frequency, recency, positional constancy (graying for focus)
 - Serial position curves
 - But beware of vertical scrolling – changes what is recent
 - Possibly adapt menu design to cohorts - task analysis
 - Position markers in long interactions to provide information on where you are and the ability to move backwards

©2020 Gregg Vesonder

These interaction styles are the staple of the web. Note that there is still considerable design issues regarding placement and focus. Always be cognizant of what is “above and below the page fold.” Newspaper terms for what is on the first half of the front page – above the page fold – and what is on the second half. (New York Post is a whole other issue ;-).

Form Completion Guidelines

- Meaningful title
- Comprehensible instructions
- Logical grouping and sequencing of fields
- Visually appealing layout of forms
- Familiar field labels
- **Consistent, standardized** terminology, abbreviations
- Convenient cursor movement
- Error correction for fields
- Error prevention
- Error messages for unacceptable values
- Marking of optional fields
- Explanatory messages for fields
- Completion signal and % done

©2020 Gregg Vesonder

Fairly self explanatory – basically there is a lot of work to do this stuff well. HTML5 has made it easier with more extensive error correction. We will see more guidelines on completion signals and percent done later in the course.

Dialog Box Guidelines

- Internal Layout
 - Meaningful title, **consistent** style
 - Top left to bottom right sequencing
 - Clustering and emphasis
 - Consistency in layout, terminology, appearance
 - Standard buttons (ok, cancel)
- External Relationships
 - Smooth appearance and disappearance
 - Distinguishable but not overpowering boundary
 - Easy to make disappear
 - No overlap of required items (pop-ups)
 - Clear how to complete cancel (e.g., focus)

©2020 Gregg Vesonder

Again straight forward

Command and Natural Languages

- Language Design Goals
 - Precision
 - Compactness
 - Ease in reading and writing
 - Completeness
 - Speed in learning
 - Simplicity to reduce errors
 - Ease of retention over time

©2020 Gregg Vesonder

Using command languages, think Linux or MS-DOS command line, we are looking in the command language case for precision and compactness (many linux shell commands are short because their creators wanted to limit typing). While using Natural language should increase the speed of learning

CNL -2

- High level goals:
 - Close correspondence between reality and notation
 - Convenience in carrying out manipulations relevant to users' tasks
 - Compatibility with existing notations (regular expressions)
 - Flexibility to accommodate novice and expert users
 - Expressiveness to encourage creativity
 - Visual appeal

©2020 Gregg Vesonder

Some additional goals, and for command languages, it eases the burden on the cpu – we often do not consider it, but a valuable UI characteristic is responsiveness.

CNL-3

- **Constraints on Language**
 - Capacity for humans to record notation
 - Match between recording and display media
 - Convenience in expressing (including speech)

©2020 Gregg Vesonder

Most command language options should be easy to say, even if that means spelling it ("L S" for example when saying linux's directory command, ls).



Linux Gems

- `ls -f *.doc`
- `grep vesonder *.txt > namefiles`
- `cat datafile | uniq -c | sort > counts`
- `rm *`
- `enscript -2rgh $*`
- `man rm`

©2020 Gregg Vesonder

Speaking of ls, here are some linux command line commands! Often linux commands are an attempt to abbreviate the description of their function: ls for list, rm for remove, man for manual, Indeed cat is for concatenate. As raspberry pi and other single board, system on a chip computers become available it will be useful to understand command line languages.

Command Language Guidelines

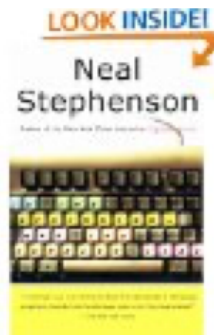
- Create explicit model of objects and actions
- Choose meaningful, specific, distinctive names
- Try to achieve hierarchical structure
- Provide consistent structure (hierarchy, argument order, action-object pairs)
- Support consistent abbreviation rules (prefer truncation to one letter)
- Offer frequent users the ability to create macros
- Consider menus on high speed displays
- Limit the number of commands and ways to accomplish the task (anti-PERL)
- Provide history and command line editing

©2020 Gregg Vesonder

Back in the day, there were conferences devoted to command languages. In my opinion command languages are used too infrequently these days. In many instances they would make an ideal high efficiency user interface. Scripting languages are interesting. I would suggest that JavaScript and Python are essential if you want to be a developer.

In the Beginning was the Command Line

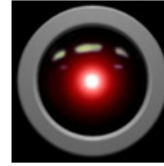
- <https://smorgasborg.artlung.com/CRYPTONOMICON.shtml>



©2020 Gregg Vesonder

A great personal history of the command line by a fantastic author, Neal Stephenson. You can get the entire text of the book at the url I provided. Snow Crash was **the** book for game developer's in the 90's and I think it is still one of his best works. In general he writes great (and lately long) series of novels, that have geek elements. For the computer security aficionados in the audience I recommend Cryptonomicon. Awesome.

Anthropomorphic



- Siri, Alexa (amazon Echo), Google Home, Microsoft Cortana (Windows 10)
- Human/human interaction is not necessarily an appropriate model for HCI
- Habitability issue - how users can know what objects and **interactions are appropriate**
- Expanding to gesture recognition, facial expressions, eye movements - even a mailing list on artificial emotion

©2020 Gregg Vesonder

I generally was not a fan of Natural Language interfaces, even though I worked extensively with them, actually with aspects of them – speech recognition and text to speech. Siri and Alexa changed all that. Siri and Alexa provides one of the first opportunities to use this technology effectively. One of the newer research efforts is machine perception and machine presentation of emotion. I would not be pleased if my robot got too angry with me – I do keep a hammer nearby in case my Roomba turns on me! The picture in the corner is of HAL from the film, 2001: A Space Odyssey – a can't miss film. To listen to some of HAL's more memorable excerpts, check out these audio clips: <http://www.ee.ryerson.ca/~elf/aso/>. If you have an iPhone and Siri try asking her to "Open the pod bay doors Hal." Do the same with Alexa if you know someone who has an amazon echo.

References

- Parnas, D.L. & Wurges, H. "Response to undesired events in software systems." In D.M.Hoffman and D.M.Weiss (eds), Software fundamentals: Collected papers by D.L. Parnas, Addison-Wesley, 2001, ISBN: 0-201-70369-6
- Preece, J., Rogers, Y. and Sharp, H. Interaction Design: Beyond human computer interaction, John Wiley, 2002, ISBN: 0-471-49278-7
- Endres, A. and Rombach, D. A handbook of software and systems engineering, Addison Wesley, 2003, ISBN: 0-321-15420-7.
- Gershenfeld, N. When things start to think, Owl Books, 1999(2000), ISBN:080505880X
- Morville and Rosenfeld, Information architecture for the world wide web 3rd Edition, O' Rielly, 2006, ISBN: 0-596-52734-9
- Stone, Jarrett, Woodroffe and Minocha User interface design and evaluation, Morgan Kaufmann, 2005