Categorizing Knowledge

Jeff Offutt

http://www.cs.gmu.edu/~offutt/

SWE 632
User Interface Design and Development
Shneiderman, Ch. 2

Syntactic & Semantic Knowledge

Syntactic vs Semantic

1) Actions – things that can happen
2) Objects – things that exist

1) Computer semantic knowledge
2) Task semantic knowledge

Structured
Independent of machine, OS
Stable in memory

Varied
Dependent on device, machine, and OS
Rote memorization
Easily forgotten

© Jeff Offutt, 2002-2011

Syntactic & Semantic

- Finding files
  - find . -name "*.ppt"
  - Start – Find – Files or Folders …
- Searching within files
  - grep "cat*" filename
  - Ctrl-F

How do you describe this?

- Regular expression?
- Wild card?

© Jeff Offutt, 2002-2011

Syntactic & Semantic

- Syntax knowledge is about how to manipulate things, without regards to the concepts
  - How to type, without knowing language
  - How to turn a radio on, without understanding the sounds
- Semantic knowledge is about the concepts
  - The language we type in
  - The meaning of the music and the words
- "thank you", "xie xie", "gracias", "cám ón" all have the same semantics, but very different syntax

© Jeff Offutt, 2002-2011

Computer and Task Semantic

- Task-semantic knowledge is about how to perform an activity
- Computer-semantic knowledge is about how the internals of the computer works – the hardware, software platform, etc.
- Editing a paper is task-semantic, and comp-semantic is about how the file is stored on disk, what printing is, etc.

We don't acquire comp-semantic knowledge by simply using the computer

© Jeff Offutt, 2002-2011

Users and Knowledge

Consider users in terms of all three types of knowledge

- Novice
  - No syntactic
  - Little comp-semantic
  - Little task-semantic

  Few options, slower, lots of feedback, tutorials, careful defaults
- Knowledgeable intermittent
  - Difficulty retaining syntax
  - Good retention of semantic

  Help the memory, recognition, consistency, on-line help
- Frequent user (expert)
  - Familiar with syntax and semantics

  Speed, shortcuts, customization

  feedback is intrusive

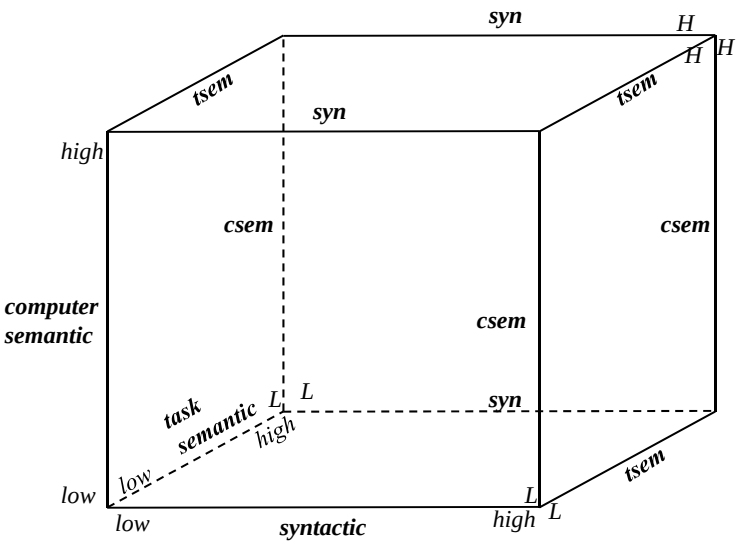
© Jeff Offutt, 2002-2011

Users and Knowledge

- This model is too simplistic
- The three types of knowledge are independent and form a 3-space …

© Jeff Offutt, 2002-2011

The Knowledge 3-Space



© Jeff Offutt, 2002-2011

Understand the Users

It is important to know who the user is

- Work experience
- Computer experience
- Age
- Education
- Reading skills
- Language skills
- Work environment
- Task frequency
  - ⋮

© Jeff Offutt, 2002-2011

Modes of Interaction
(Types of Interfaces)

- Menu (selection)
- Form fill (selection)
- Command language
- Natural language
- Direct manipulation – GUI, WUI

© Jeff Offutt, 2002-2011

Eight "Golden" Principles for Dialogue Designers

Valid for any type of interface

1. Consistency
   - Design in, then **evaluate**
   - Changes break consistency …
2. Shortcuts
   - Users must be able to find them
3. Feedback
   - Not just error messages
4. Yield Closure
   - Have a clearly defined end-point in the interaction

© Jeff Offutt, 2002-2011

Eight "Golden" Principles

5. Error Handling
   - Clearly tell me what was wrong
   - Only make me redo the part that was wrong
6. Undo
   - If the operation cannot be "undone," use hesitation
7. Put the user in charge
   - Inexperienced users may be intimidated when the software makes decisions
   - Experienced users want to control the flow
8. Reduce the STM load

© Jeff Offutt, 2002-2011

Eight "Golden" Principles

9. Design for the USER
   - First
   - Last
   - Then test it

© Jeff Offutt, 2002-2011

Preventing Errors

- People often make mistakes
- Faster computers can increase errors

- Prevention strategies :
  - Flow : Users make fewer mistakes when the flow through the UI makes sense
  - Education : Better error messages can reduce errors

© Jeff Offutt, 2002-2011

Software Seatbelts

- The software can prevent the user from making dangerous choices
- If the dangerous choice must be available, allow it with a hesitation ("are you sure?")
  - Exiting editors with changed, unsaved text
    Should not hesitate if unchanged!
  - rm *.o … rm * .o
    rm –i *.o forces hesitation (alias rm "rm –i")
  - Do not put choices on menus that should not be used

Simplifying Displays
Five Objectives

1. Consistency
   - Format, terminology, abbrevs, titles
   - Easier to repair than to build in!
2. Efficient assimilation
   - Neat columns
   - Left justification of characters
   - Right justify integers
   - Spacing, etc

Simplifying Displays
Five Objectives

3. Reduce memory load
   - Do not cross screens
   - No long sequences of interaction

4. Input should look like output
   - Put dashes in phone numbers and credit card numbers

5. Allow output displays to be flexible

© Jeff Offutt, 2002-2011

Highlighting

- Highlighting can appear in many ways :
  - **bold**
  - marking
  - size
  - font
  - CAPITALIZE
  - inverse video
  - blinking
  - color
  - audio
- **BE GENTLE!! HIGHLIGHTING ALWAYS LOOKS STRONGER TO THE USER THAN TO THE DESIGNER!!!**

© Jeff Offutt, 2002-2011

Rules for Data Entry

- Be consistent !
- Minimize user inputs
  - Single character choices
  - Reduce device switching (keyboard – mouse)
  - No redundancy (addresses …)
- Minimize memory load
- Make data entry similar to data display
- Be flexible : Again [YN] (N)?

© Jeff Offutt, 2002-2011

Summary

All an interface designer has to do is

- Be polite
- Be considerate
- Be clear

© Jeff Offutt, 2002-2011