Modal User Interfaces

- Modal
  - actions take on a different meaning depending on the current state or mode
  - e.g., dragging with mouse in a drawing program depends on the current tool
Example of Modal UIs

• Some dialog boxes
  – requiring action before anything else
  – why can this be bad?
• VI editor
  – command mode vs. insert mode
  – how do you know which mode you are in?
• Drawing/paint programs
• Palette-based programs
Problems with Modal UIs

• **Mode errors**
  – think you are in one mode but really in another
  – e.g., in vi (want “mu” -> “muddle”)
  – if in command mode by accident, deletes the line

• **Mode hides functionality you want**
  – e.g., to deal with a dialog box must switch modes

• **Constant mode switching may be slow**
  – e.g. Adobe Illustrator
  – lots of tools in palette
  – One solution is keyboard shortcuts
    • (not a great solution)
Are Modal UIs bad?

• Not necessarily
  – can help make a large interface easier to use
    • do not need so many different commands

• Only bad if done wrong
  – modal dialog boxes
  – modes that are not visible (*)
    • palettes are a fine use of modes
Speech User Interfaces
UIs in the Pervasive Computing Era

- Future computing devices won’t have the same UI as current PCs
- Wide range of devices
  - Small or embedded in environment
  - Often with alternative I/O & w/o screens
  - Information appliances

I-Land vision by Streitz, et. al.
Motivation

- Smaller devices -> difficult I/O
  - People can talk at ~90 wpm (high speed)
- “Virtually Unlimited” set of commands
- Freedom for other body parts
  - Imagine you are working on your car and need to know something from the manual
- Natural
  - Evolutionarily selected for speech
  - Not for reading, writing or typing
When to use Speech

- Mobile
- Hands-busy
- eyes-busy
- Assistive Technologies
Why are they hard to get right?

• Speech recognition far from perfect
  – Imagine mouse with 5-20% error rate
• Speech UIs have no visible state
  – Can’t see what you have done before
  – Can’t see effect of commands
• Speech UIs are hard to learn
  – Can’t easily explore interface
Why are they hard to get right?

• Isolated, short words difficult
• Segmentation
  – Recognize speech
  – Wreck a nice beach
• Spelling
  – mail vs. male
  – need to understand language
• Context is necessary
Speech UIs require

- Speech recognition
  - the computer understanding what the customer is saying.

- Speech production (or synthesis)
  - the computer talking to the customer.
Designing Speech UIs

• Speech UI no-no’s
  – modes
    • no feedback
    • certain commands only work when in specific states
  – deep hierarchies (aka voice mail hell)
• Verbose feedback wastes time/patience
  – only confirm consequential things
  – use meaningful, short cues
• No Barge-In Support
  – Must wait for UI to finish
Designing Speech UIs

- Too much speech is tiring
- Speech takes up working memory
  - can cause problems when problem solving
- Establish shared context
  - Make sure people know
    - what type of tool they are using
    - where they are in the conversation
Designing Speech UIs

Pacing

• recognition delays are unnatural
  – make it clear
• barge-in lets user interrupt like in real conversations
• progressive assistance
  – short error messages at first
  – longer when user needs more help
• Implicit confirmation
  – include confirm in next command
Disadvantages of Speech UIs

Close to Home
John McPherson
Disadvantages of Speech UIs

• Disruptive
• Privacy Concerns
• Recognition Errors
• Multiple Verbal Tasks (Interference)
• Context Errors
Future:

Future UIs for Information Access

• Star Trek style UI
  – verbally ask the computer for info or services
  – Hard: it requires perfect speech recognition & unambiguous language understanding
Future:
MultiModal Interaction

- Multimodal interfaces use different kinds of input (e.g., pen and speech) together
- Achieves "put that there"
VoicePen: Augmenting Pen Input with Simultaneous Non-Linguistic Vocalization
Muscle-Computer Interaction

Figure 1: Artist rendering of a forearm band with EMG sensors that could be used for muscle-computer interfaces.
Future:

Context-Aware Applications

• Apps are aware of context
  – User location
  – What they are doing
  – Who is around
  – What is appropriate / relevant
Kate Everitt’s Research

- Use **physical context** to assist speech recognizer

- $\alpha$- WISP tags detect objects in use

- Activate different grammars based on state of objects
Questions

- When would you use a speech UI?
- What speech UIs have you encountered?
- Have they been good?
- How have speech UIs changed?
- What are the problems with Speech UIs?
Summary

• Speech UIs
  – May permit more natural computer access
  – Allows us to use computers in more situations
  – Are hard to get to work well
    • Lack of visible state, tax working memory, recognition problems, etc.
    • Multimodal UIs address some of the problems with pure speech UIs.
Exercise

Would you use a speech UI for the following?
Why or why not?

1. Banking system
2. Registration/Enrollment for University
3. Internet browser for blind users
4. Remote service manual for traveling repairman
5. Database management system
Motivation for Speech UIs: Pervasive Information Access

I-Land vision by Streitz, et al.
Information access via speech

Read my important email