Human Factors Theory: Part III

Part of the Human Computer Interaction course Notes

2008-2009

Where we are

- Model of how our mind works
- Cognition
  - Mental models
  - Problem solving
  - Learning
- Attention and memory
- Perception (visual and auditory)
- Motor skills
- **Social science, dialog with computer**
- Design guidelines

Social science

- How humans behave towards each other
- How humans behave when they interact with machines
- Why they react in certain ways

- Human and computer dialogs
  - Studies found that it’s similar to human-human dialogs

Dialog design

- Natural and simple, avoid technical jargons
- Consistent in terms of style

“To err is human; to forgive is by design.”

- Provide legal choices.
  - Gray out illegal choices.
- Provide key completion.
- Recall is harder, recognize is easier
- Consider user effort

Treat errors in positive and helpful manners

- Mistakes
  - Arise from conscious deliberations that lead to an error instead of the correct solution
- Slips
  - Unconscious behavior that gets misdirected en route to satisfying goal
    - e.g. drive to store, end up in the office
- Shows up frequently in skilled behavior
  - usually due to inattention
  - often arises from similarities of actions

slip: impreciso, trascurato

Designing rules

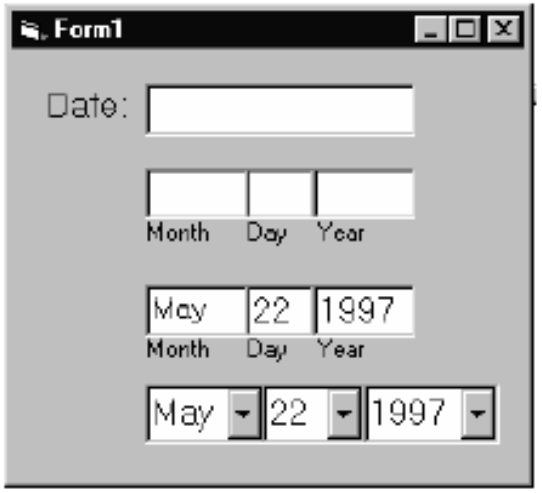
- General rules
  - Prevent errors before they occur
  - Detect and correct errors when they do occur
  - User correction through feedback and undo

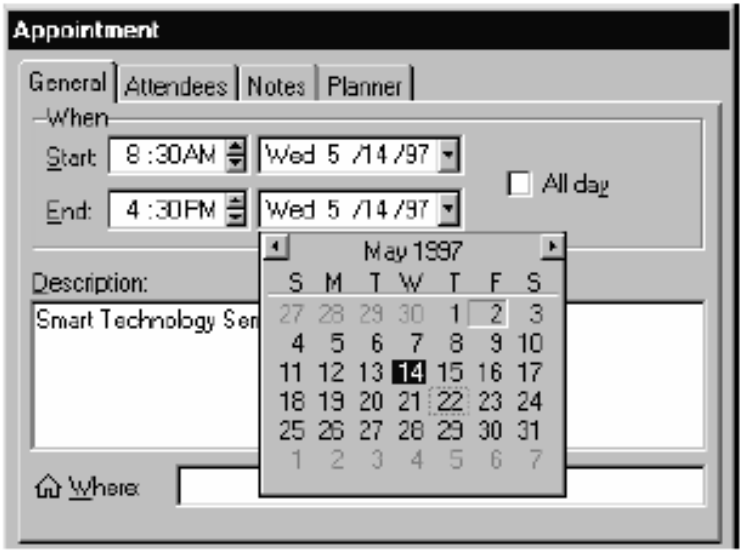
Examples

- mode errors
  - have as few modes as possible (preferably none)
  - make modes highly visible
- capture errors
  - instead of confirmation, make actions undoable
  - allows reconsideration of action by user
- loss of activation
  - if system knows goal, make it explicit
  - if not, allow person to see path taken
- description errors
  - in icon-based interfaces, make sure icons are not too similar

Treat errors in positive and helpful manners

- Prevent errors
  - try to make errors impossible
  - modern widgets: only “legal commands” selected, or “legal data” entered





- Provide reasonableness checks on input data
  - on entering order for office supplies 5000 pencils is an unusually large order. Do you really want to order that many?

Provide help

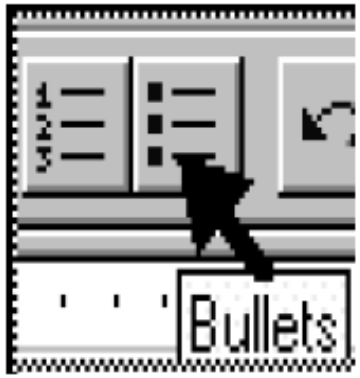
- Help is not a replacement for bad design!
- Simple systems: walk up and use; minimal instructions
- Most other systems: feature rich
  - some users will want to become “experts” rather than “casual” users
  - intermediate users need reminding, plus a learning path

Types of help

- Tutorial and/or getting started manuals
  - short guides that people are likely to read when first obtaining their systems
    - encourages exploration and getting to know the system
    - tries to get conceptual material across and essential syntax
- on-line “tours”, exercises, and demos
  - demonstrate very basic principles through working examples

Types of help

- Reminders
  - short reference cards
    - expert user who just wants to check facts
    - novice who wants to get overview of system's capabilities
  - keyboard templates
  - shortcuts/syntactic meanings of keys; recognition vs. recall; capabilities
  - tooltips
  - text over graphical items indicates their meaning or purpose

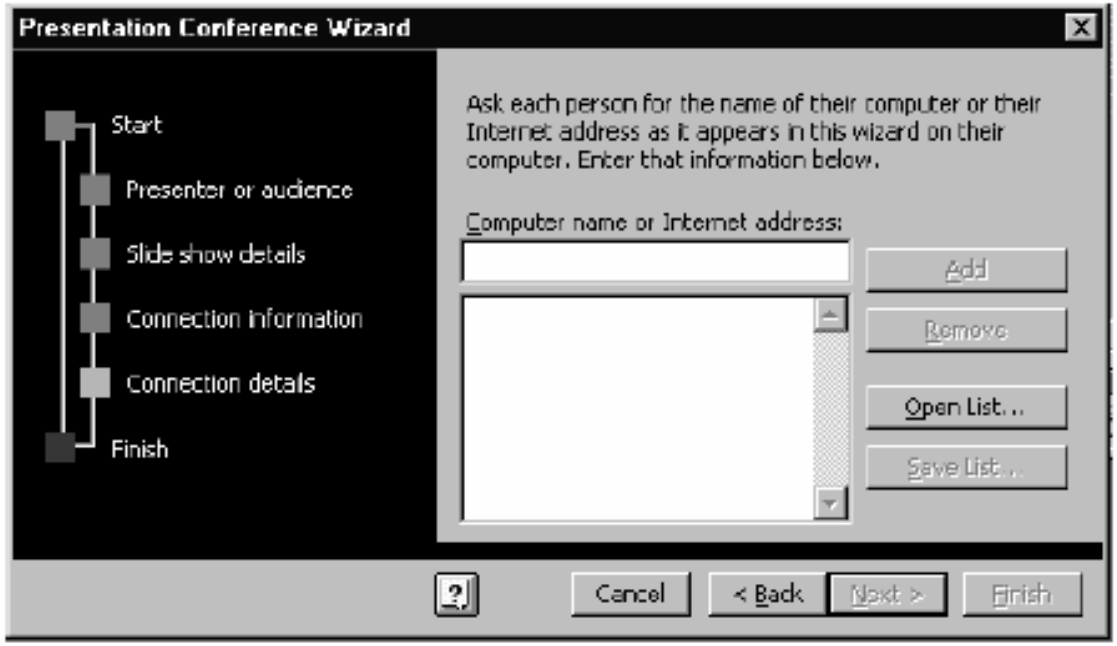


Types of help

- Context-sensitive help
  - System provides help on the interface component the user is currently working with
    - Tool tips
    - Macintosh “balloon help”
    - Microsoft “What’s this” help
      - brief help explaining whatever the user is pointing at on the screen

Types of help

- Wizards
  - walks user through typical tasks
  - *but* problematic if user gets stuck



Types of help

- Tips
  - migration path to learning system features
  - also context-specific tips on being more efficient
  - must be “smart”, otherwise boring and tedious

Error message design

- Specificity
  - Too simple, or general -> not enough information
  - Condemning -> frustrating

Poor	Better
Syntax error	Unmatched left parenthesis
Illegal entry	Type first letter: Send, Read, or Drop
Invalid data	Data range from 1 to 31
Bad file name	File names must begin with a letter

Show correct actions and use positive tone

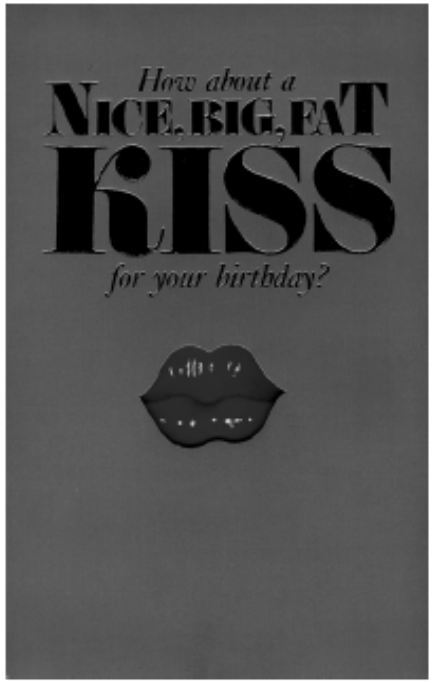
- Poor:
  - Disastrous string overflow. Job abandoned.
- Better:
  - String space consumed. Revise program to use shorter strings or expand string space
- Poor:
  - Undefined Labels
- Better:
  - define statement labels before use

Where we are

- Model of how our mind works
- Cognition
  - Mental models
  - Problem solving
  - Learning
- Attention and memory
- Perception (visual and auditory)
- Motor skills
- Social science, dialog with computer
- **Design guidelines**

The KISS principle

- design simplicity should be a key goal and unnecessary complexity avoided.



Keep It Simple and Stimulating

- Principle categories
  - ensure correct mental models
  - facilitate problem solving
  - maximize learning
  - minimize memory load
  - ease information processing
  - provide positive dialogs
- This is simply not so simple to do.
- Many designers cannot help giving users all the widgets they can offer.
- Featuritis

KISS principle

Alan Kay put it correctly :

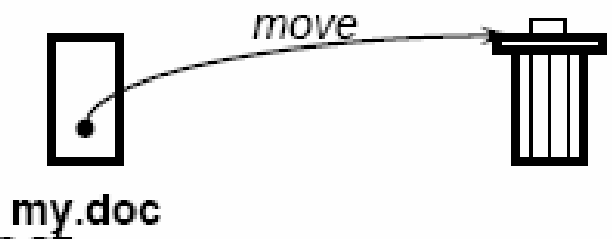
"*Interaction designers can make an effort to keep simple tasks easy for the user, and to make complex tasks possible*"

The Metaphor of Direct Manipulation

- Direct Engagement
  - the feeling of working *directly* on the task
- Direct Manipulation
  - An interface that behaves as though the interaction was with a real-world object rather than with an abstract system
- Central ideas
  - visibility of the objects of interest
  - rapid, reversible, incremental actions
  - manipulation by pointing and moving
  - immediate and continuous display of results
- Almost always based on a metaphor
  - mapped onto some facet of the real world task semantics)

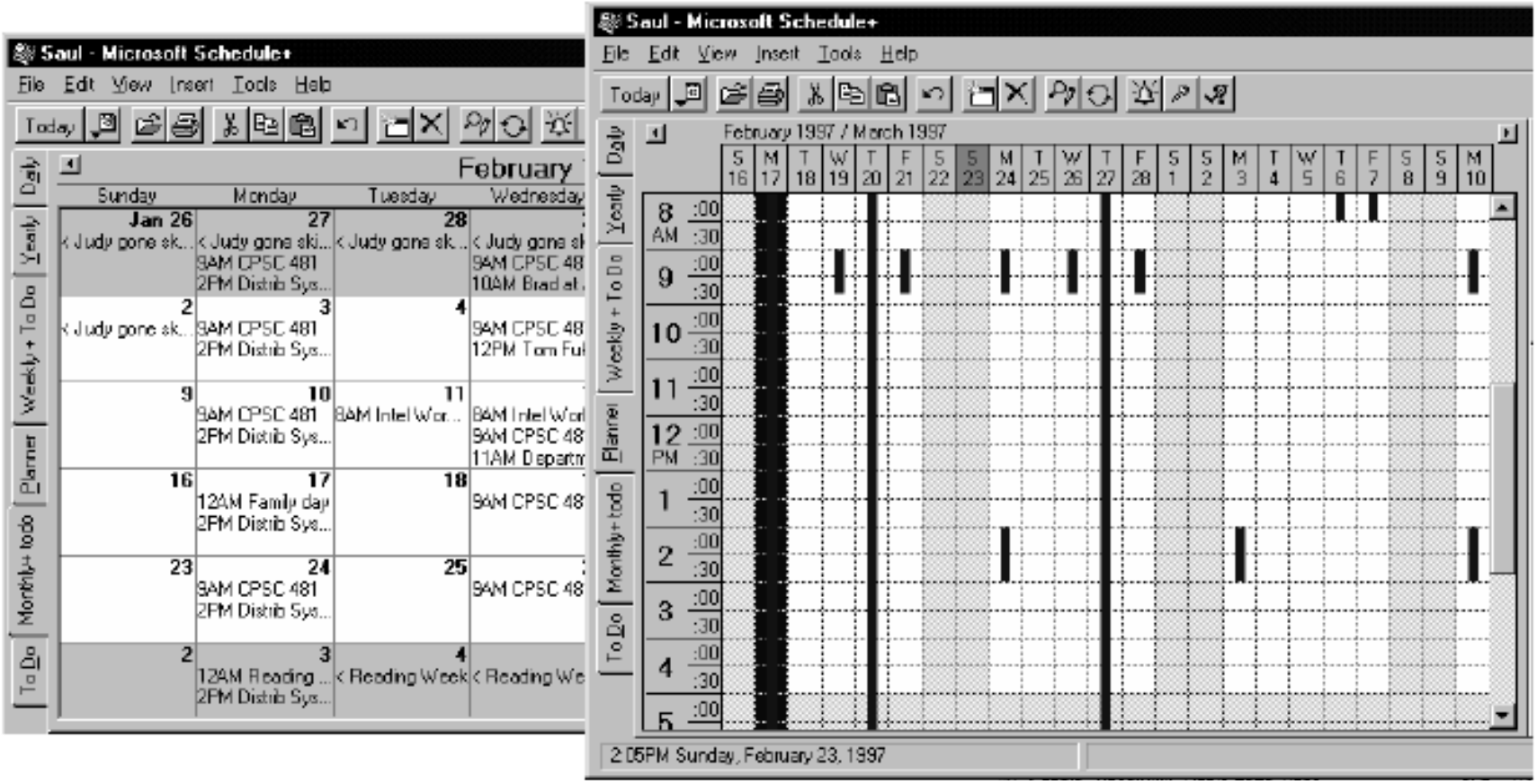
Object-Action *vs* Action-Object

- Select object, *then* do action
  - interface emphasizes 'nouns' (visible objects) rather than 'verbs' (actions)
- Advantages
  - closer to real world
  - modeless interaction
  - *actions* always within context of object
    - inappropriate ones can be hidden
  - *generic commands*
    - the same type of action can be performed on the object
    - eg drag `n drop:
      -



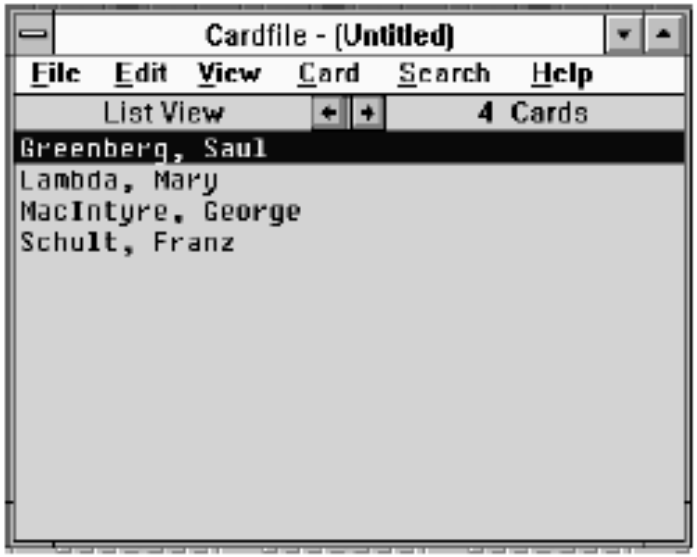
Direct manipulation

- Representation directly determines what can manipulated

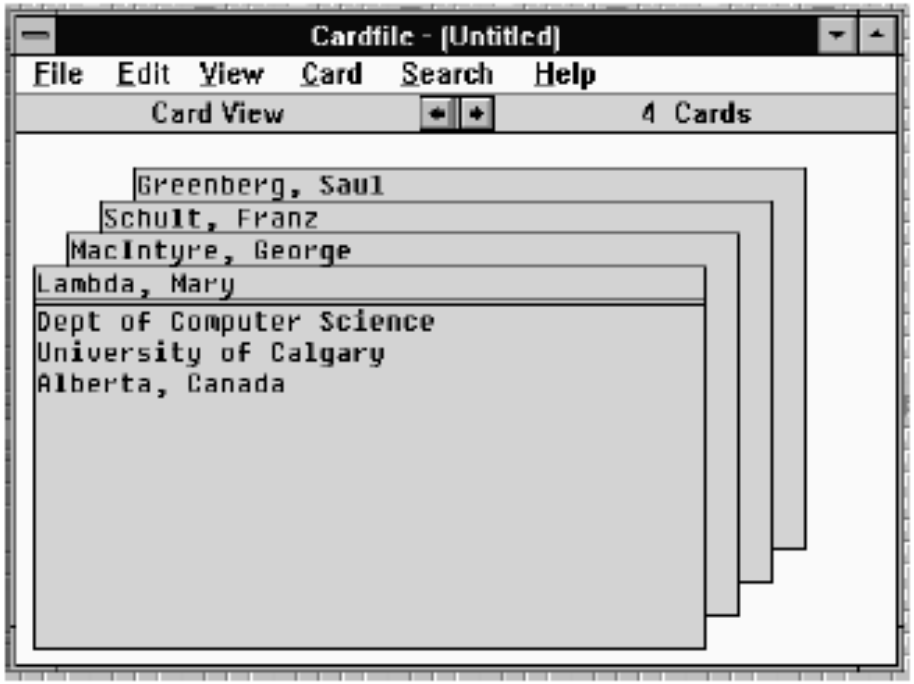


Phone list

List metaphor



Rolodex metaphor

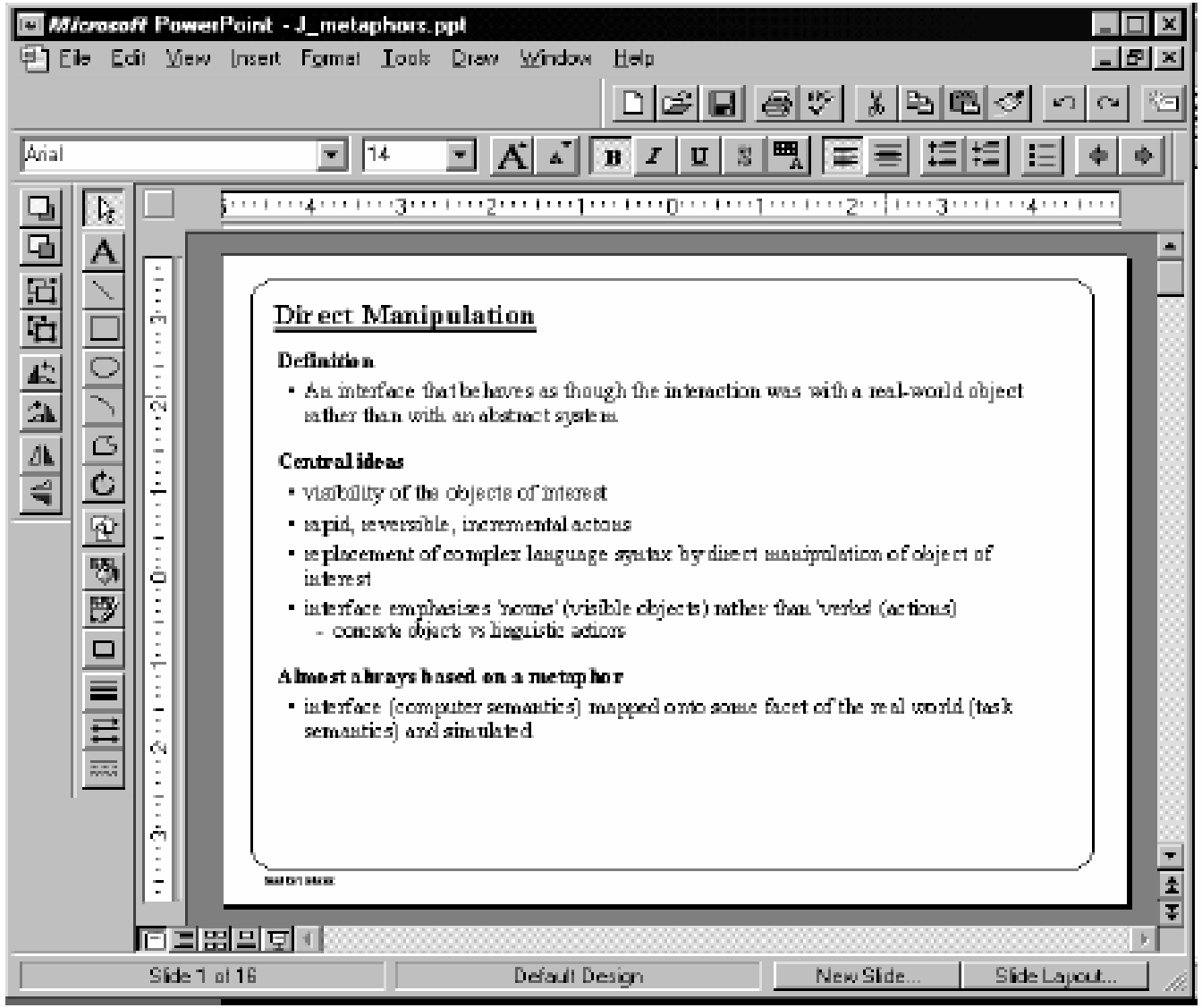


Is direct manipulation the way to go?

- Some Disadvantages
  - Ill-suited for abstract operations
    - spell-checker?
  - Tedium
    - manually search large database vs query
  - Task domain may not have adequate physical/visual metaphor
  - Metaphor may be overly-restrictive

- Solution: Most systems combine direct manipulation and abstractions
  - word processor:
    - WYSIWYG document (direct manipulation)
    - buttons, menus, dialog boxes (abstractions, but direct manipulation “in the small”)

Conventional Applications: A Mix



Guidelines continued

- Provide a good conceptual model
  - allows users to predict consequences of actions
  - communicated through the image of the system
- Make things visible
  - relations between user's intentions, required actions, and results should be
    - sensible
    - consistent
    - meaningful (non-arbitrary)
  - make use of visible affordances, mappings, and constraints
    - remind person of what can be done and how to do it

Summary

- Good Representations
  - captures essential elements of the event / world
  - deliberately leaves out / mutes the irrelevant
  - appropriate for the person, their task, and their interpretation
- Metaphors
  - uses our knowledge of the familiar and concrete to represent abstract concepts
  - need not be literal
  - has limitations that must be understood
- Direct manipulation
  - visibility of the objects of interest
  - rapid, reversible, incremental actions
  - manipulation by pointing and moving
  - immediate and continuous display of results