

# Direct Manipulation

- Common pattern in a wide variety of successful interactive systems:
  1. Visibility of objects and actions of interest
  2. Rapid-response, reversible, incremental actions
  3. Actions have a “noun-verb” structure vs. older “verb-noun”
- Very effective interaction style *if implemented well* — and frequently, that is a *huge* “if”

## User Responses to Well-Done Direct Manipulation

- Mastery of the interface
- Competence in performing tasks
- Ease in learning originally and in assimilating advanced features
- Confidence in the capacity to retain mastery over time
- Enjoyment in using the interface
- Eagerness to show off the interface to novices
- Desire to explore more powerful aspects

# Word/Document Processing

- “Line” vs. “display” editors (e.g. *ed*/*edlin*, *vi*)
- Then came WordStar (for consumers) and *emacs* (for technical types)
- Then came WYSIWYG editors — original MacWrite, these days almost always Microsoft Word
- Direct manipulation characteristics: page-like display; matching screen/printed versions; visible and natural cursor motion; visual/graphical controls (rules, tab marks, icons); immediate display of results; rapid response; easily reversible actions

## The Spreadsheet

- Invented in 1979 by Harvard Business School students — VisiCalc: 254 rows and 63 columns
- Lotus 1-2-3 in the ‘80s, then Microsoft Excel
- More direct manipulation characteristics: direct cell entry/editing; immediate recalculations of modified formulas and values; genuine 2-dimensional and spatial layout; insert, move [drag], and delete columns, rows, or cells; point-and-click of cell targets or ranges

# Spatial Data Management and Display

- “As-is” displays of geographies and spaces — computerized maps are most popular
- Pointer-dragging corresponds directly to motion within the map (or space)
- Zoom-in or point-and-zoom to display more detailed information
- Promising newcomer: Google Maps — high degree of interactivity over a network connection

## Video Games

- From the beginning (literally, with PONG), video games have been all about direct manipulation
- Rapid-response correspondence between game controller manipulation and video game response (character, spaceship, weapon, other objects)
- Many actions are reversible within the rules of the game (movement, open/close doors, add/drop objects)
- New elements: full 3D engines, social element brought about by network play
- Some differences: games are meant to entertain and challenge — not applicable to other types of software

# Computer-Aided Design and Manufacturing (CAD/CAM)

- Computer screen representation of objects being designed (buildings, floor plans, circuits), plus additional information and actions not normally available if working in the “real world”
- Quick review/revision of multiple design alternatives is a particularly compelling feature for CAD/CAM
- Intriguing historical note: Sketchpad (watch for it in the upcoming Alan Kay video)
- Interesting Alan Kay term: “intimacy”

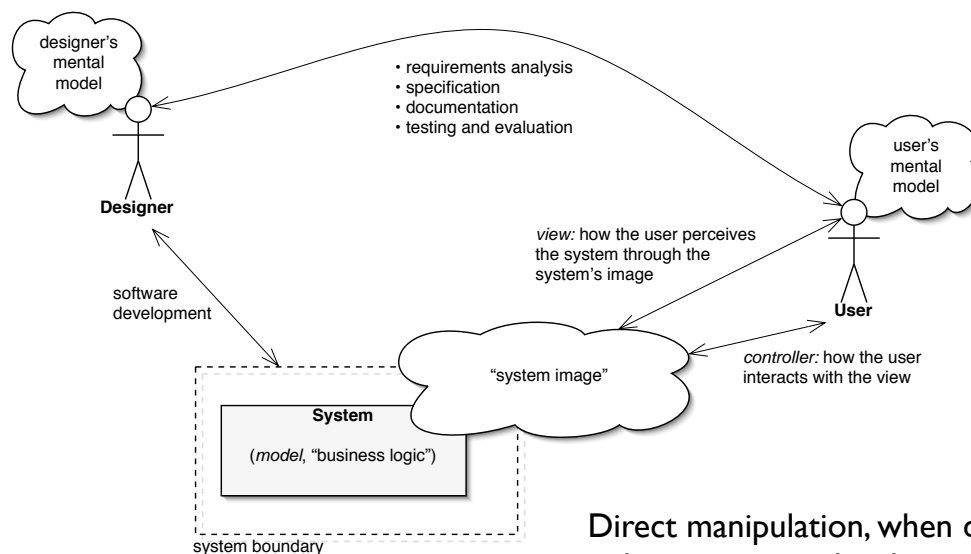
## “Office Automation” — or, Computers in General

- Direct manipulation is a key factor in the personal computer revolution: Apple Macintosh then Microsoft Windows, as preceded by Xerox Star and Apple Lisa
- Which came first, object-oriented programming or direct manipulation? Hard to say — the first full-blown object-oriented programming language, Smalltalk, was expressly developed to make programming of direct manipulation UIs easier
- Current GUI “widgets” use direct manipulation extensively — windows, buttons, sliders, etc.

# Why Does It Work?

- “Principle of virtuality” (Nelson 1980) — reality is represented in a way that can be affected by the user
- “Principle of transparency” (Rutkowski 1982) — “the tool itself seems to disappear”
- Corresponds to Norman’s seven stages: direct manipulation simultaneously bridges both gulfs of execution and evaluation
- Many explanations from psychology and education: representation is key to understanding (note the longevity of the abacus)

It all boils back down to this:



based on Don Norman's model, with some UML, software engineering, use case modeling, and MVC tweaks

Direct manipulation, when done right, represents the shortest distance between a user's understanding of a domain and the presentation of that domain by a system image

# Caveat Manipulator

Of course, direct manipulation is not without its flaws or challenges — otherwise HCI would be solved!

Problem	Solution Approach
not very accommodating of visually- or motor-impaired users	screen readers, accessibility guidelines, full keyboard interface
can occupy a great deal of screen space, frequently requiring scrolling	devise compact representations that retain advantages; use Tufte's micro/macro view principle
developer's visual representation may not correspond to the user's	mix text with images; extensive testing; standardization
analogy may be misleading, incomplete, or incorrect — “metaphor shear”	implement full and complete metaphor; even more testing

- The keyboard/mouse gap — the user interface version of context switching; not to be underestimated
  - ◆ Design interactions that allow lengthy “modes” — keyboard or mouse for long contiguous periods
  - ◆ Allow “one-hand-each” interfaces: e.g. 1st-person 3D shooters
  - ◆ Provide both mouse and keyboard methods for the same action — DM is *not* synonymous with mouse control (but stay efficient!)
- **Implementation!** Direct manipulation is significantly harder to implement than other implementation styles
  - ◆ Manipulation sequences — frequently requires precise mouse and keyboard event handling
  - ◆ Performance — effective direct manipulation requires a near-instantaneous perception/action loop (< 100ms, ideally < 10ms)
  - ◆ Reversibility — if there are no natural inverse actions, a generalized undo framework is key (look up the Command design pattern)
  - ◆ Technology — some platforms (notably HTML) do not naturally support direct manipulation

# Avoiding Metaphor Shear

- From intuition to intention — theory to the rescue!
- Stages-of-action theories: closely matches direct manipulation activity loop (particularly Norman's *execution* and *evaluation* phases)
- Special mention: Norman's framework *affordances, constraints, and mappings*
- GOMS and widget-level theories can help tune direct manipulation efficiency concerns
- Grammars place rigor/structure on the “manipulation language,” thus enhancing consistency

## The OAI Theory and Direct Manipulation

- Shneiderman's breakdown of direct manipulation:
  1. Continuous representations of objects and actions of interest with meaningful visual metaphors
  2. Physical actions or presses of labeled components, instead of complex syntax
  3. Rapid, incremental, reversible actions whose effects on objects of interest are visible immediately
- Surprise surprise — very close match between direct manipulation and object-action-interface

# Personal Notes on Direct Manipulation

- Abstract concepts (and domains) are harder to “DM-ize” than domains that correspond to the physical world — proceed with more caution here
- Do not underestimate the necessity for rapid response in direct manipulation: if raw latency cannot be ensured, then the user interface must be ready to provide progress feedback; or, ensure that back-end or business logic is reliable and fast
- Thorough specification is a key aid to efficient implementation of DM: particularly, state diagrams, screen prototypes and storyboards

## Direct Manipulation: Not for Users Only

- ...as evidenced by the popularity of visual programming environments such as HyperCard, Visual BASIC/Studio, Macromedia Flash, Mac OS X Interface Builder, and NetBeans
- Much of Eclipse’s popularity can be linked to direct manipulation as well: continuous compilation, immediate and clickable errors and warnings, predefined refactoring operations

# Extensions to Direct Manipulation

- 3D interfaces
  - ◆ Very compelling and appealing
  - ◆ Some current projects: Sun's Project Looking Glass; the open-source Croquet project
  - ◆ But watch out for getting "too real" — part of direct manipulation's appeal is that it simplifies the representation of the domain
- Teleoperation — natural extension of DM also
- Virtual and augmented reality — full circle, returning direct manipulation back to the real world