Multimedia

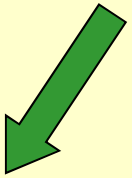
Applications

What is Multimedia Useful For?

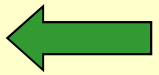
- Education
  - Tutoring systems
  - Encyclopaedias
  - Instruction manuals
- Information
  - Tourist information
  - Museums
  - Art galleries
- Entertainment
  - Games
  - Art

Spelling Fusion

Multimedia Definition



"Multimedia is the seamless integration of text, sound, images of all kinds and control software within a single digital information environment."



Tony Feldman, multimedia consultant
(quoted in England and Finney, "Managing Multimedia")

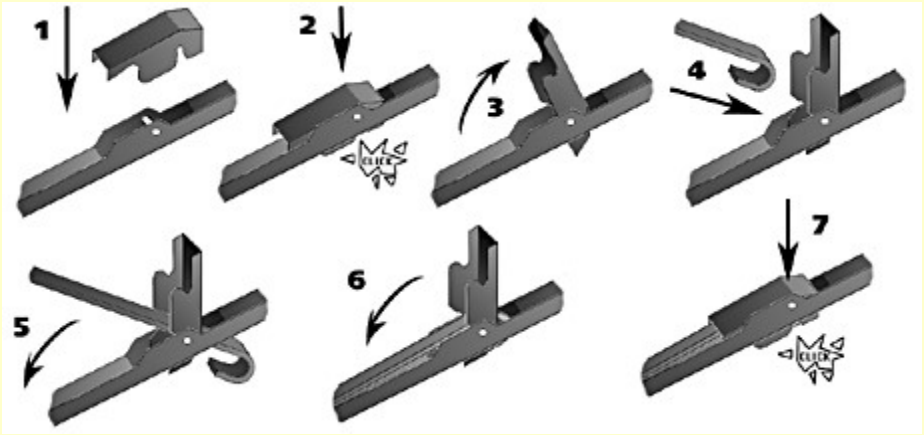
Communicating Information

- Modalities
  - Vision, hearing, touch
- Channels of communication
  - Within a single modality
  - e.g. speech, sound effects, music
- Medium = coordinated channels
  - May be multimodal
  - e.g. animation + sound track
  - e.g. picture + caption

- From Elsom-Cook, "Principles of Interactive Multimedia", McGraw-Hill, 2001

A Simple Example

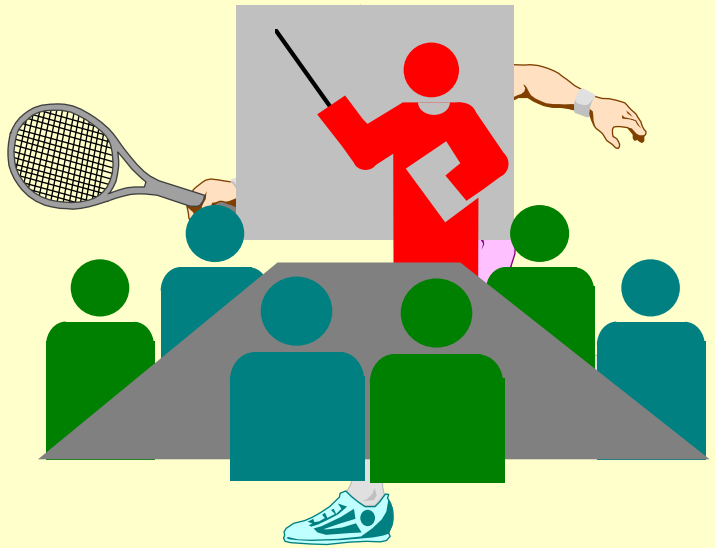
- Which is easier to follow?



- From England and Finney, "Managing Multimedia"

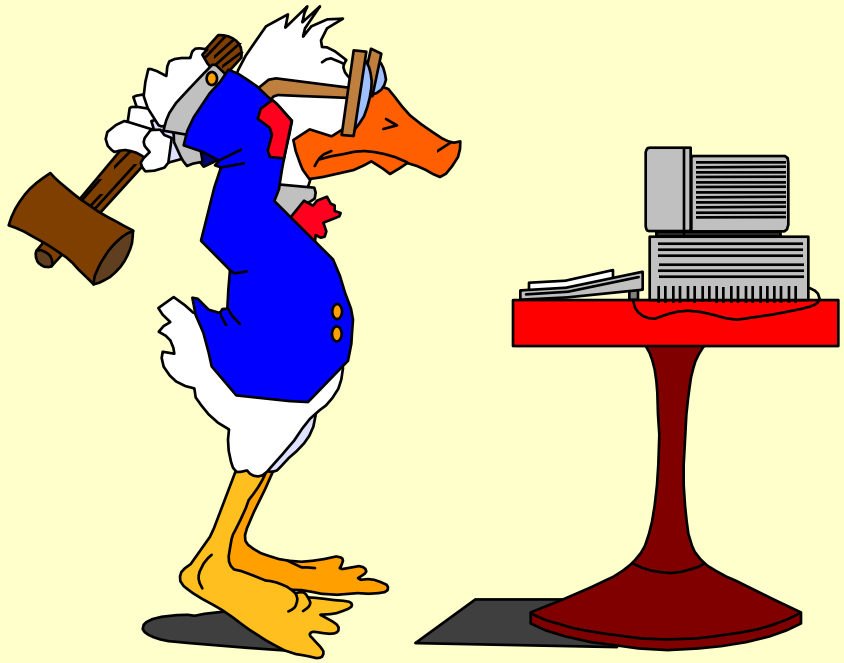
Pros of Multimedia

- Grab and hold attention
  - humans have a limited attention span
- Alternative media for the same message
  - text, sound and images
- Combined media can enhance message
- Interactivity
  - Doing aids learning



Cons of Multimedia

- Poor design leads to total confusion!
- Reliance on multimodal input/output acts against people with disabilities
- Lack of suitable computer
  - still certainly possible!



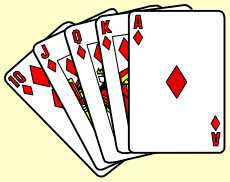
How Do I Build A Multimedia Presentation?

- Design process
  - considered in “design” lectures
- Content produced using disparate software/hardware
  - drawing / animation packages
  - word processing
  - image capture with scanners / cameras / videos
  - sound recording / generation
- Presentation produced using an authoring tool(s)
  - brings elements together
  - adds control and navigation

How is Multimedia Delivered?

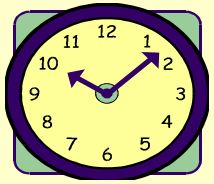
- Offline
  - Installations / kiosks
    - single site with known hardware
  - CD-ROM / DVD / Software download
    - multiple sites with no control over target hardware
- Online
  - Communication over networks
    - limited bandwidth
  - User may require suitable software (plug-in)
  - Feedback and interaction possible

Styles of Presentation

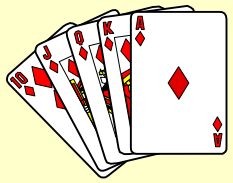


CARD-BASED

EVENT-BASED



TIME-BASED

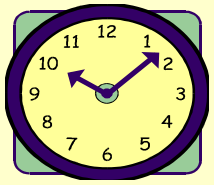


CARD-BASED

- Presentation consists of 2D pages (page-based)
- Elements arranged in the way text and images are laid out in books and magazines
  - text, images, videos, sound
- Time-based elements
  - occupy a fixed space
  - controls to start/stop playback
- Links between pages: *hypermedia*
- E.g. HyperCard, ToolBook, PowerPoint, World-Wide Web (HTML)

EVENT-BASED

- Presentation is an *event-driven* system
  - user does something (e.g. clicking a button) and something happens in response (a movie is played, perhaps)
  - contemporary GUIs
- Associating *actions* with *events*
- Events initiated by user (mouse clicks, key presses etc) or generated internally (movie finishes, time passes etc)
- Actions (behaviours) predefined or scripted (playing a movie, changing images, doing the unexpected…)
- Authoring tools include Adobe's Authorware or programming languages: Java, Javascript etc



TIME-BASED

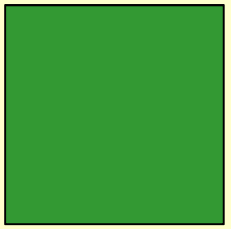
- *Time* is the organising principle
- Elements arranged on a timeline
  - presented in sequence (e.g. like a slide show)
  - parallelism: more than one thing may be going on at once
  - synchronization (e.g. display text while video is playing)
- May still incorporate some user control
- Adobe's Flash is time-based tool for web and standalone presentations

The Complete Presentation

- Actually most multimedia presentations contain aspects of all three presentation styles
  - Elements laid out in space
  - Elements change under user control / navigation
  - Elements change over time
- Development process may consider different aspects in sequence eg web pages:
  - Layout content of web pages in space
  - Add links to create interactivity
  - Add animated elements eg slide show

Authoring with HTML+CSS+JS

- All aspects of the three presentation styles can be implemented by a combination of:
  - Hypertext markup language (HTML)
  - Cascading style sheets (CSS)
  - Javascript (JS)
- We will use HTML+CSS+JS for multimedia development in CSCU9N5:
  - Practicals
  - Assignment



Page Structure with HTML

- blocks provide a convenient way to define logical parts of your page
  - Menu
  - Main content
  - Side bar
  - Heading bar
- Give each a unique identifier
  - ID attribute
- Position a with CSS
- Dynamically change content with JS

Multimedia with HTML

- <audio> and <video> tags
  - Native browser support for specific (limited) file formats
  - Browser can supply user controls
  - Customise control with JS
- Graphics with <canvas>
  - Drawing via JS libraries
  - Browsers support standard 2D graphics library
  - 3D graphics and animation available via custom 3rd party JS libraries

CSS

- Positioning to generate page layout
  - We will use mostly positioning with absolute coordinates within a fixed “page” size
- Style
  - All aspects of element style should be done with CSS
- Animation
  - CSS does enable animation of style and position of an element
  - Transitions
  - Key frames

Animation with CSS

- Transitions

```
div {
    width: 100px; height: 100px;
    background: red;
    transition: width 2s, height 2s, transform 2s;
}
div:hover {
    width: 300px; height: 300px;
    transform: rotate(180deg);
}
```

Animation with CSS

- Key frames

```
div {
    width: 100px; height: 100px;
    background-color: red;
    animation-name: example;
    animation-duration: 4s;
}

@keyframes example {
    from {background-color: red; width: 100px;}
    to {background-color: yellow; width: 200px;}
}
```

JavaScript (JS)

- Dynamic changes to content and appearance
- Event handling

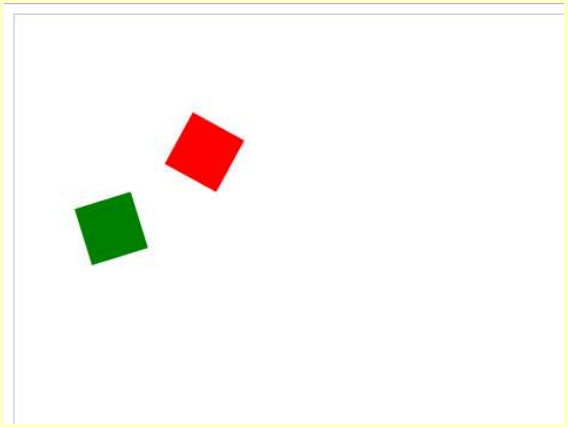
```
<button id="mybutt" onclick="chngtxt()">Click</button>
<script>
function chngtxt() {
document.getElementById("mybutt").innerHTML = "Done";
}
</script>
```

- Graphics and animation via Canvas
- Calculations

Animation with JS

- Animation on a canvas is achieved by:
  1. Defining drawing operations in a function
  2. Clear the canvas
  3. Draw objects slightly differently from previous time eg location, size, rotation
  4. Call function repeatedly at a defined time interval:

```
setInterval(animate, 30);
```



End of Lecture