

# Multimedia im Netz

Wintersemester 2010/2011

## Part I

# Web Technologies for Interactive Multimedia

# 2 Media on the Web

- 2.1 Media Embedding in HTML
- 2.2 Client and Server Functionalities
- 2.3 Media Players for the Web

# Embedding a YouTube Video

```
<object width="500" height="315">  
<param name="movie" value=  
"http://www.youtube.com/v/_oBuE66majc&hl=de&fs=1&rel=0&border=1">  
</param>  
<param name="allowFullScreen" value="true"></param>  
<param name="allowscriptaccess" value="always"></param>  
<embed src=  
"http://www.youtube.com/v/_oBuE66majc&hl=de&fs=1&rel=0&border=1 "  
type="application/x-shockwave-flash" allowscriptaccess="always"  
allowfullscreen="true" width="500" height="315"></embed></object>
```

- Redundant information
  - Nested “object” and “embed” tags
- Adobe Flash runtime code referenced
  - MIME type “application/x-shockwave-flash”
  - Movie player program, parameterized

# Embedding Media

- Media embedding requires:
  - Media data (a file)
  - Player software
- Typical media data:
  - Sound files (e.g. .wav, .midi)
  - Movie files (e.g. .avi, .mov)
  - Programs to be executed on a virtual machine
    - » Java applets
    - » Flash runtime code (Shockwave Flash, .swf)
- Browser integration:
  - Built-in: Browser "knows" about player for media type
  - Plug-in: Flexible association between player and media type

# <embed> Tag in HTML

- <embed> tag refers to browser *plugin*
  - Introduced by Netscape with browser version 2.0
  - **Outdated, not** part of the HTML standard
- Example:  
`<embed src="yippee.wav" width="140" height="60">`
- Plugin:
  - Separate program to handle special file types
    - » E.g. Flash player plugin handles .swf files
  - Located on client
- Important attributes:
  - **src**: Data to be embedded (URI or local file)
  - **width, height** etc.: Control of appearance
  - **autostart**: Determines whether playback starts immediately
  - **pluginspage**: Where to find information on the plugin software
  - **pluginurl**: Where to find the plugin software

# <object> Tag in HTML

- **<object>** : Generic solution to embed arbitrary data files
  - Part of HTML 4.0 and XHTML 1.0 standards, supported by Microsoft
  - Supports media files, files to be opened with separate application software, files to be opened with plugin software, executable programs (e.g. Java applets or ActiveX controls)
  - Not well supported in all browsers

- Example (modern standard-conform style):

```
<object data="nibbles.swf"  
  type="application/x-shockwave-flash"  
  width="600" height="400">  
  <param name="movie" value="nibbles.swf">  
  <param name="quality" value="high">  
</object>
```

- Important attributes:
  - **data**: Data to be embedded (URI or local file)
  - **width**, **height** etc.: Control of appearance
  - **type**: MIME type of data
- Nested tag **<param>** to convey arbitrary name/value pairs

selfhtml.org

# More on the <object> Tag in HTML

- Further attributes:
  - **classid**: May be used to specify the location of an object's implementation via a URI. It may be used together with, or as an alternative to the `data` attribute, depending on the type of object involved.
    - » Specifies the version of the player software to be used
    - » In practice often platform specific, e.g. ActiveX registry values
  - **codebase**: Specifies the base path used to resolve relative URIs specified by the `classid`, `data`, and `archive` attributes. When absent, its default value is the base URI of the current document.
    - » In practice, misused to specify the location of the player software (like `pluginurl`)
  - **codetype**: Specifies the content type of data expected when downloading the object specified by `classid`.
    - » MIME type for code of player (not data)
- <object> tag with child tags in its body:
  - Uses the inner HTML code as display alternative

<http://www.alistapart.com/articles/flashsatay/>

# Combining `<embed>` and `<object>`

- Problems:
  - Older browsers:
    - » Microsoft IE ignores `<embed>`
    - » Netscape/Mozilla ignores `<object>`
  - Current browsers:
    - » `<object>` as shown above works on all platforms
    - » However, Microsoft IE does not allow streaming of the data (but loads all data first)
- Pragmatic solution:
  - Enclosing an `<embed>` tag in an `<object>` tag (see above)
  - Recommended for Flash, stable
  - Not (X)HTML standard conform!
- Complex solution for Flash, standard conform:
  - Use portable `<object>` code from above
  - Load a container movie which then loads the target movie

<http://www.alistapart.com/articles/flashsatay/>



# HTML 5

- HTML Version 5
  - Draft W3C standard (most recent draft 19 October 2010!)
  - Developed in parallel to XHTML 1.0
    - » XHTML 2.0 development has been stopped
- HTML 5 is partially supported already by many modern browsers
- HTML 5 contains standardized and simple media embedding tags
  - audio
  - video
  - embed

# Audio Embedding in HTML 5

- Example:

```
<html> ...
  <body>
    ...
    <audio src="nightflyer.ogg" autoplay>
      Your browser does not support the <code>audio</code> element.
    </audio>
```

- Attributes (examples):
  - autoplay: Playback starts automatically
  - controls: Control UI elements are made visible
  - loop: Plays in an endless loop
  - preload: Hints about preloading expectations
- Subelement <source>:
  - Alternative way to specify data source
  - Multiple occurrence is possible, first supported version is taken

# Video Embedding in HTML 5

- Example:

```
<html>
  <body>
    <video controls>
      Your browser does not support the <code>video</code> element.
      <source src="big_buck_bunny_480p_stereo.ogg" type="video/ogg">
      <source src="big_buck_bunny_480p_surround-fix.avi">
    </video>
```

- Additional Attributes compared to <audio> (examples):
  - height, width: Dimensions of video image
  - poster: Image to be shown until first frame becomes available
- Events (can be handled e.g. with JavaScript, examples):
  - empty
  - canplay
  - ended
  - abort
  - volumechange

## <embed> in HTML 5

- HTML 5 contains a standardized version of the <embed> element
- Purpose:
  - Embed arbitrary content played back via plug-in software
- Examples:
  - Flash content
  - Java applets
- Not intended for media playback

## Side Remark: HTML 5 vs. Flash

- HTML 5 establishes a clear alternative to Flash:
  - Simple audio and video playback
    - » Makes usage of Flash video for video portals unnecessary
    - » Open issue: File format/compression (H.264 ?)
- HTML 5 provides a `<canvas>` element
  - Surface for graphics programming
  - Interactive content can be written in JavaScript

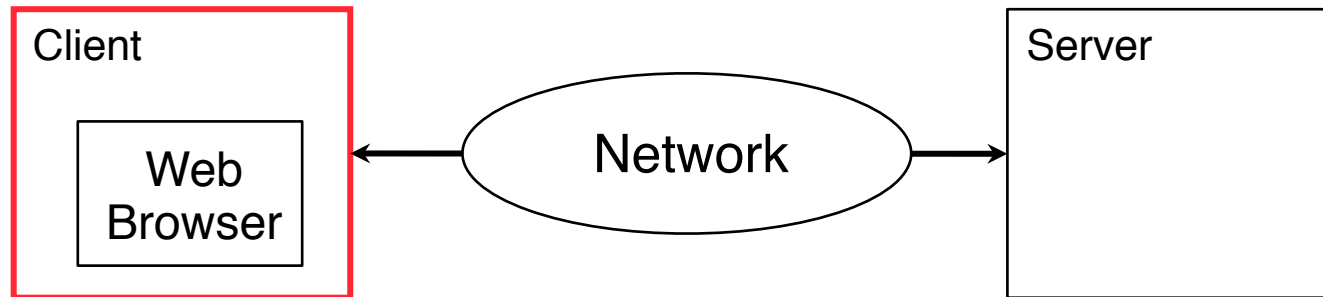
# 2 Media on the Web

2.1 Media Embedding in HTML

2.2 Client and Server Functionalities

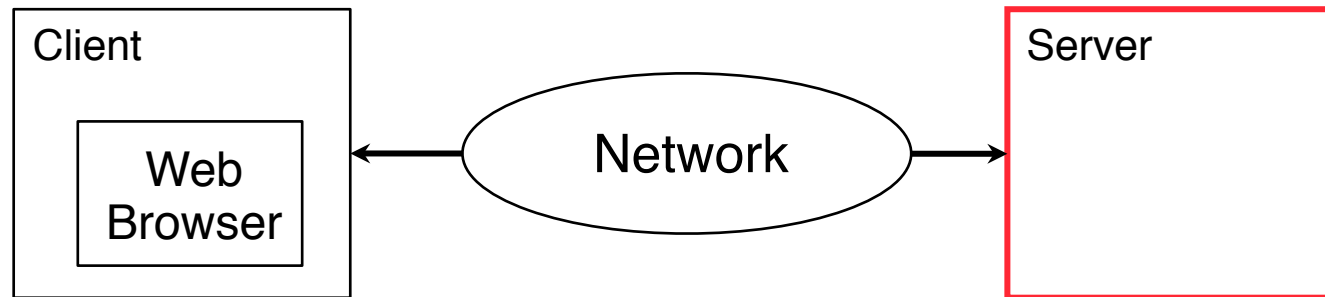
2.3 Media Players for the Web

# Functionalities Supported by Client Only



- Media rendering:
  - Recognition of media file types
    - » MIME registry of browser
  - Local media playing software
    - » Plugins or separate programs
- Interactivity:
  - Local interactions
    - » Highlighting, dynamic menus etc.

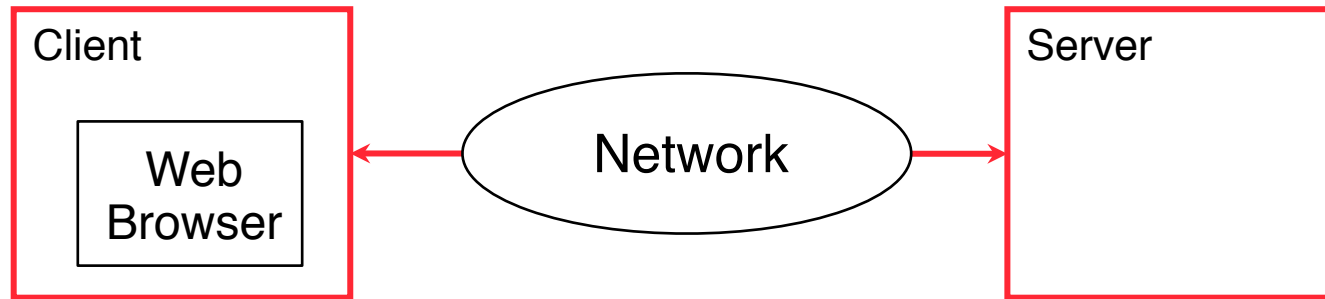
# Functionalities Supported by Server Only



- Media rendering:
  - Storage of media files and meta-information
  - Indexing and querying
- Interactivity:
  - Interactions with server-side effect
    - » E.g. database updates (registration, buying, ...)
  - Interactions with global effect for all users
    - » E.g. adding a comment, uploading a video



# Functionalities Supported by Client & Server



- Media streaming:
  - Playback of incomplete content in client
  - Playout in defined order from server
  - Synchronization, rate control, buffering
  - Flow control (stop, start, pause)
  - Adaptation to network conditions
- Interactivity:
  - Near real-time interactions
    - » E.g. status notifications, data ticker

# 2 Media on the Web

2.1 Media Embedding in HTML

2.2 Client and Server Functionalities

2.3 Media Players for the Web

# Audiovisual Players

- Player software for rendering a single audiovisual presentation
  - Audio or video file formats (e.g. .mpg, .mp3, .m4a, .mov, .avi, .wma, .wmv)
- Either browser plugin or separate application
  - Sometimes both co-exist (e.g. QuickTime, Flash)
- Examples:
  - Microsoft Media Player
  - Apple QuickTime Player
  - RealPlayer
  - Xiph.org players for open media formats

# Universal Multimedia Players

- Provide a platform for arbitrary interactive multimedia applications
  - Including media playback, but also highly interactive applications like games
- Provide an authoring platform independent of execution platform
  - Usually using a virtual machine for execution
- May include components for playback of audiovisual presentation
  - Eg. Player component in Flash playing Flash Video (.flv)
- Examples:
  - Adobe Shockwave, playing Adobe Director (.dcr) files
  - Adobe Shockwave for Flash, playing .swf files
  - Microsoft Silverlight, playing .scr files
  - Java applets, playing .class/.jar files
- Video on the Web is currently dominated by universal multimedia formats
  - Platform independence, versatility
  - Situation may change with the advent of HTML 5