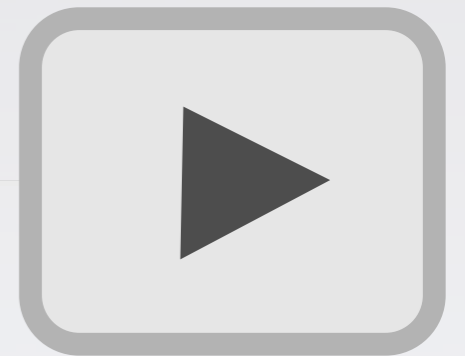


The Accessible Video Interface

Gabriel [McGovern](#) - Portland Community College



Slides, notes, code and questions



HTTP

www.gabrielmcgovern.com



MailTo

gabriel.mcgovern@pcc.edu



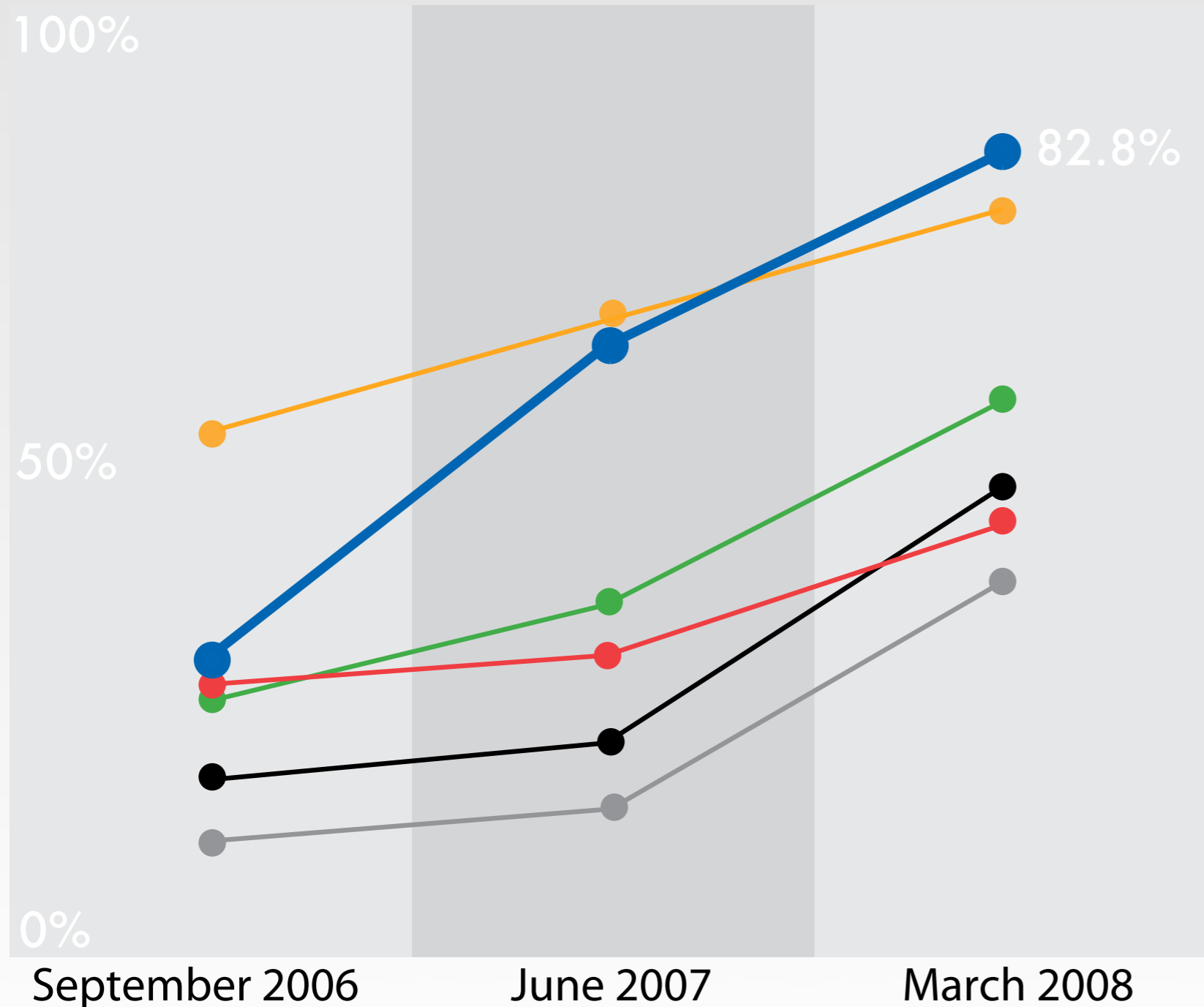
Twitter

[@gabrielmcgovern](https://twitter.com/gabrielmcgovern)

Internet killed the TV star



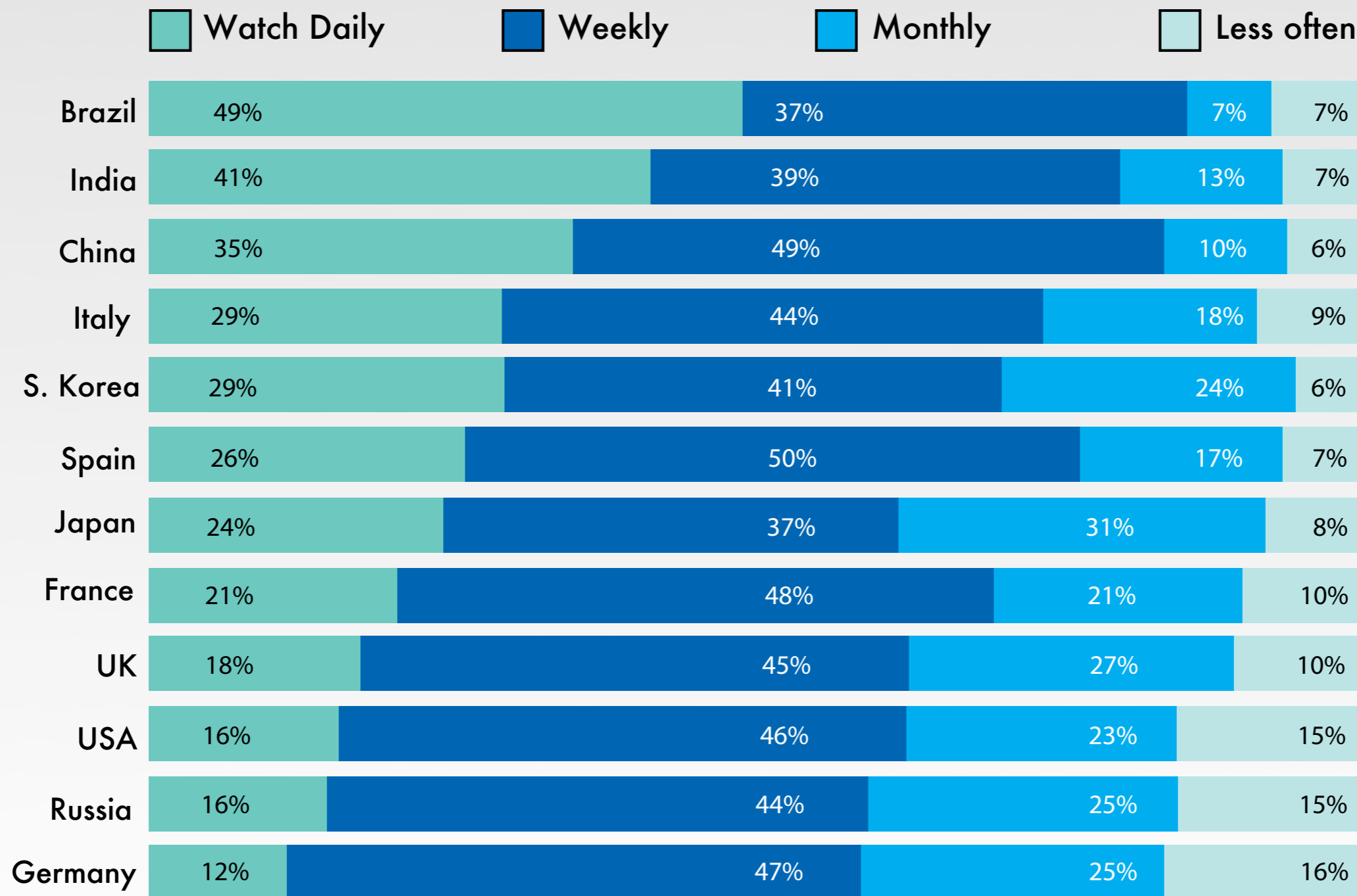
Internet killed the TV star



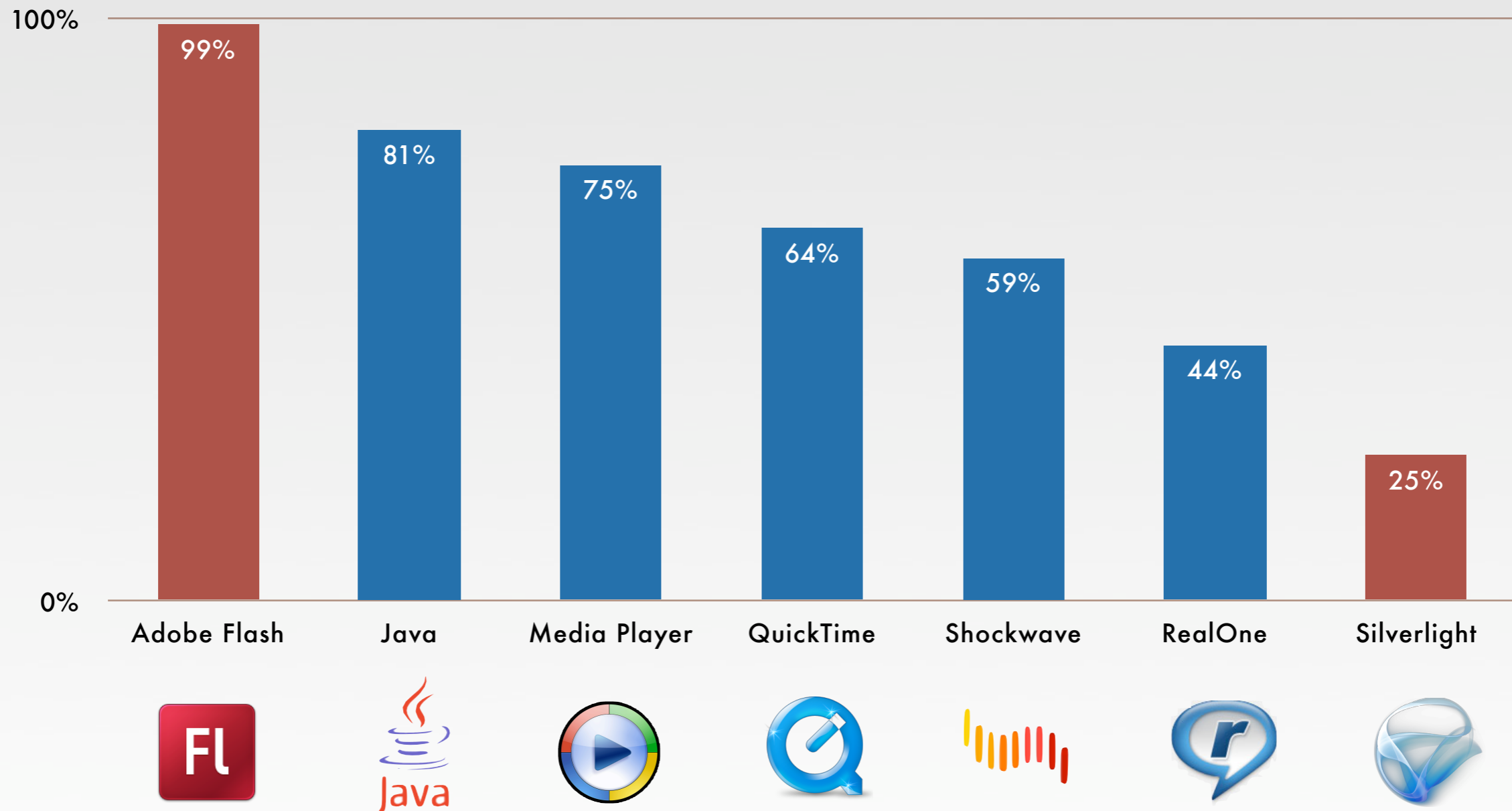
“Thinking about using the Internet, which of the following have you ever done?”

- Watch video clips online
- Read blogs
- Join a social network
- Download a podcast
- Start a blog
- Subscribe to an RSS feed

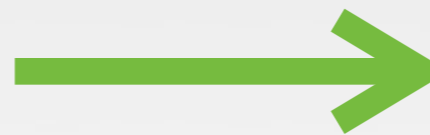
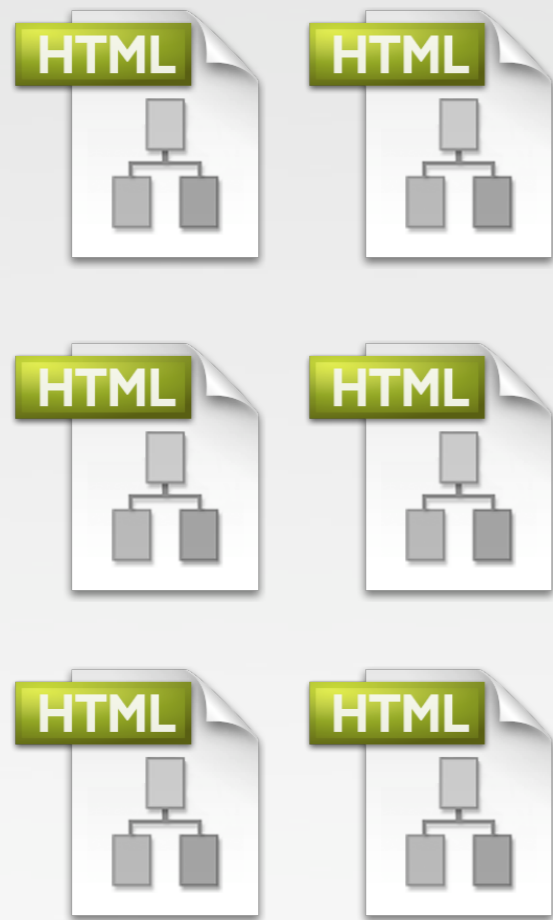
Internet killed the TV star



Internet killed the TV star



Internet killed the TV star



Requirements



Accessible

The degree to which a system can be used by the population.



Sustainable

The ability to maintain a system or process indefinitely.



Customizable

The capacity to adapt the system to meet your changing needs.

Requirements: Options

Hosted Video



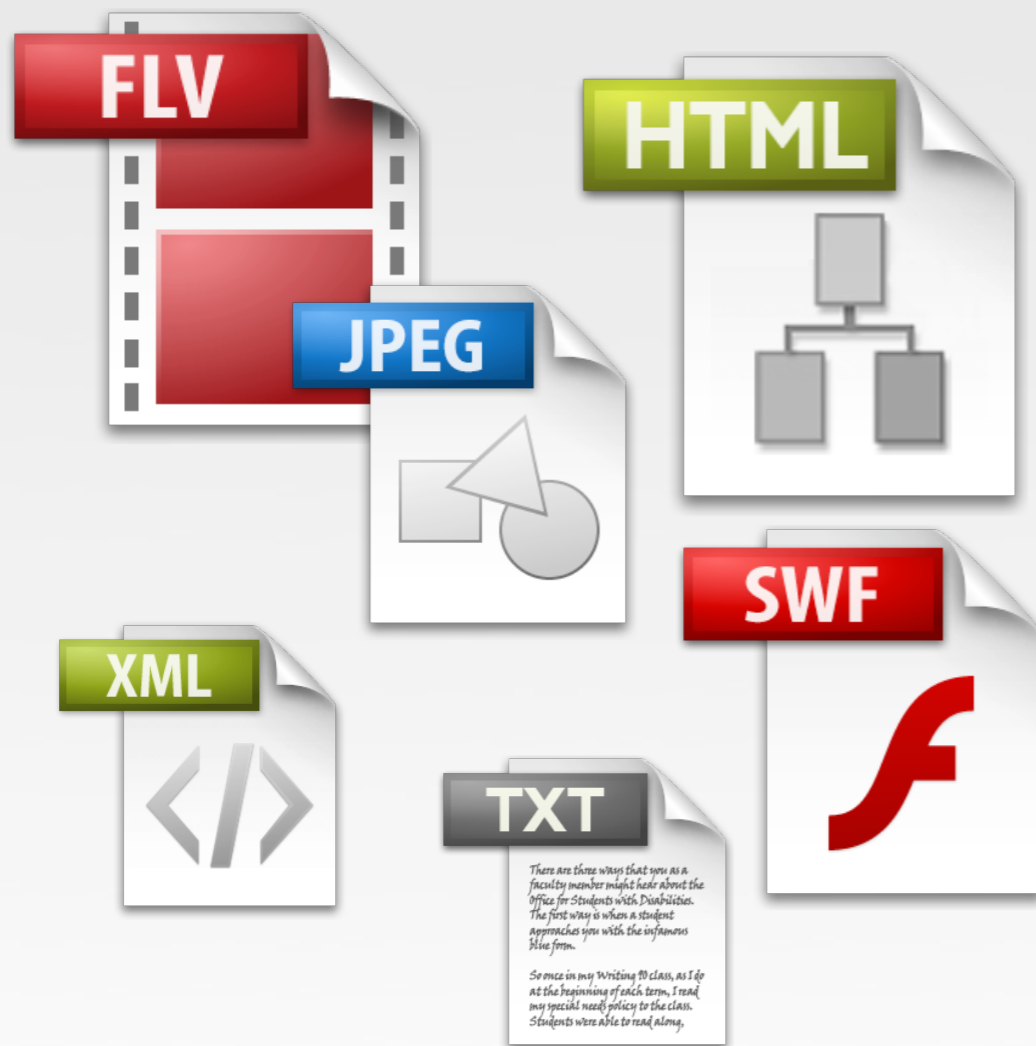
Custom Interface



Requirements: Options

	Hosted Video	Custom Interface
Ease of use	High	Low
Accessible	Med	High
Sustainable	Med	High
Customizable	Low	High

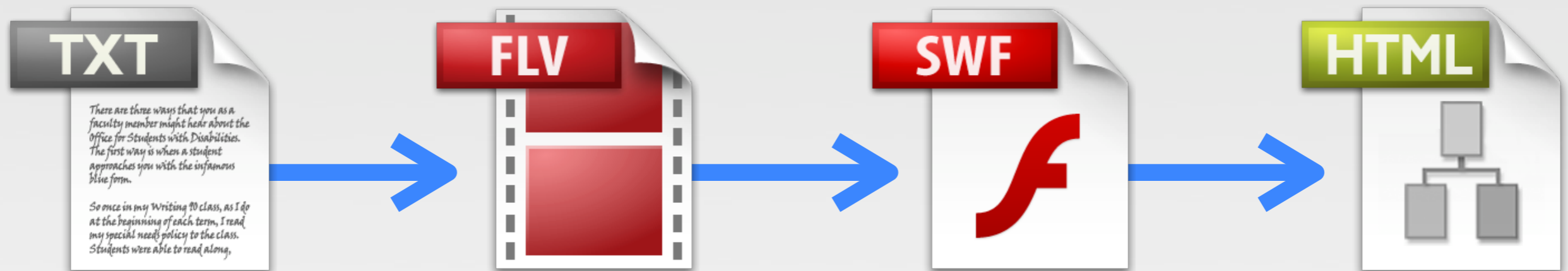
Custom Interface: What's Involved?



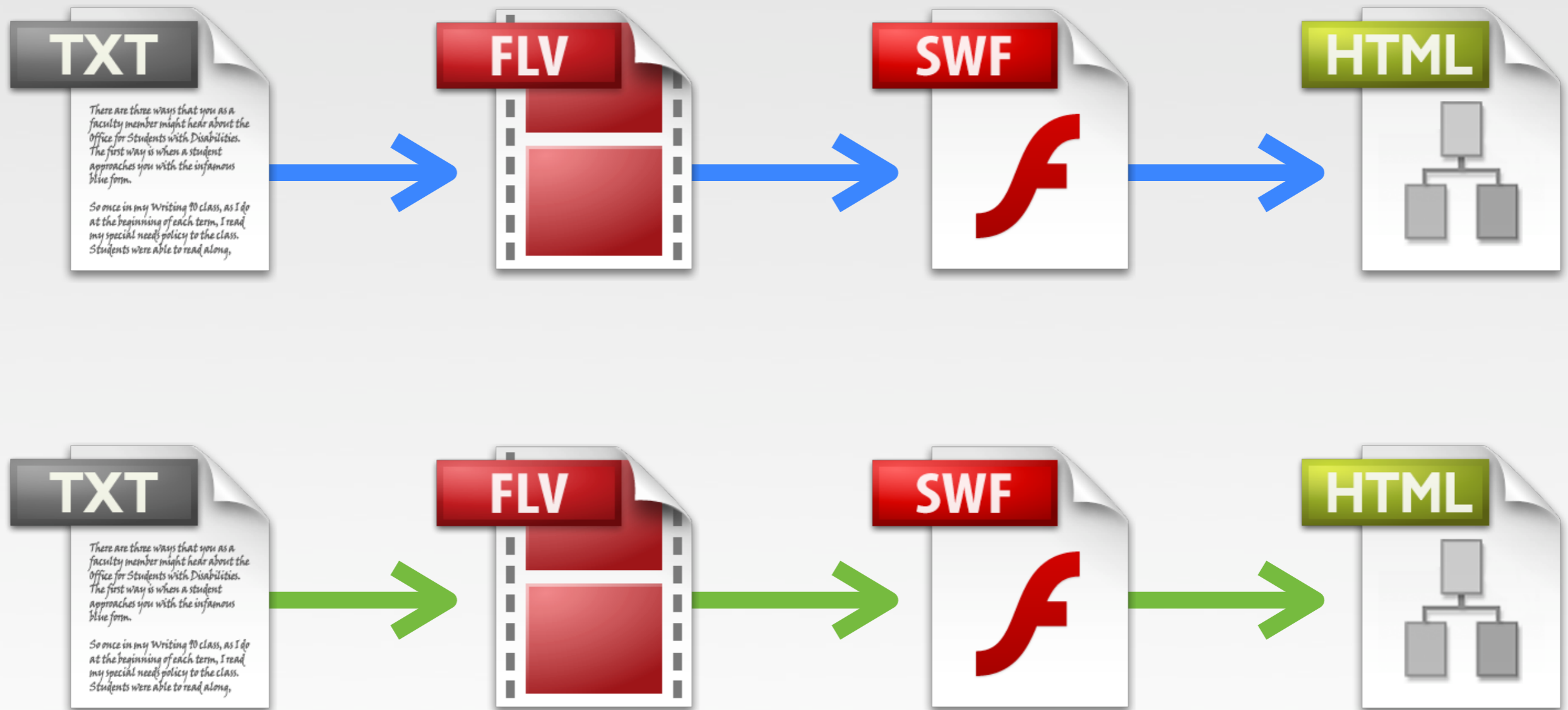
actions	• • •	•
controls	• • •	•
cc	• • •	•
image	• • •	•
video	• • •	•
background	• • •	•

```
deo.*;  
s:Object = LoaderInfo(this.loaderI  
xt:String = flashVars["title_text"  
eo:String = flashVars["path_video"  
String = flashVars["path_cc"];  
String = flashVars["path_image"]  
9  
10 if(!path_video){  
11     titleText.text = "Error: No Video";  
12 }  
13 else if(!path_image){  
14     setupVideo();  
15 }  
16 else {  
17     titleText.text = "Loading Video...";  
18     imagePane.source = path_image;  
19 }  
20  
21 var nStart:Number;  
22 function progressHandler(event:Event):void {  
23     nStart = getTimer();  
24 }  
25 function completeHandler(event:Event):void {  
26     var nSeconds:Number = (getTimer() - nStart  
27     var kBTotal:Number = imagePane.bytesTotal  
28     var nKbps:Number = (kBTotal * 8) / nSecond  
29     var nBps:Number = Math.floor(nKbps * 1024);  
30  
31     setupVideo(nBps);  
32
```

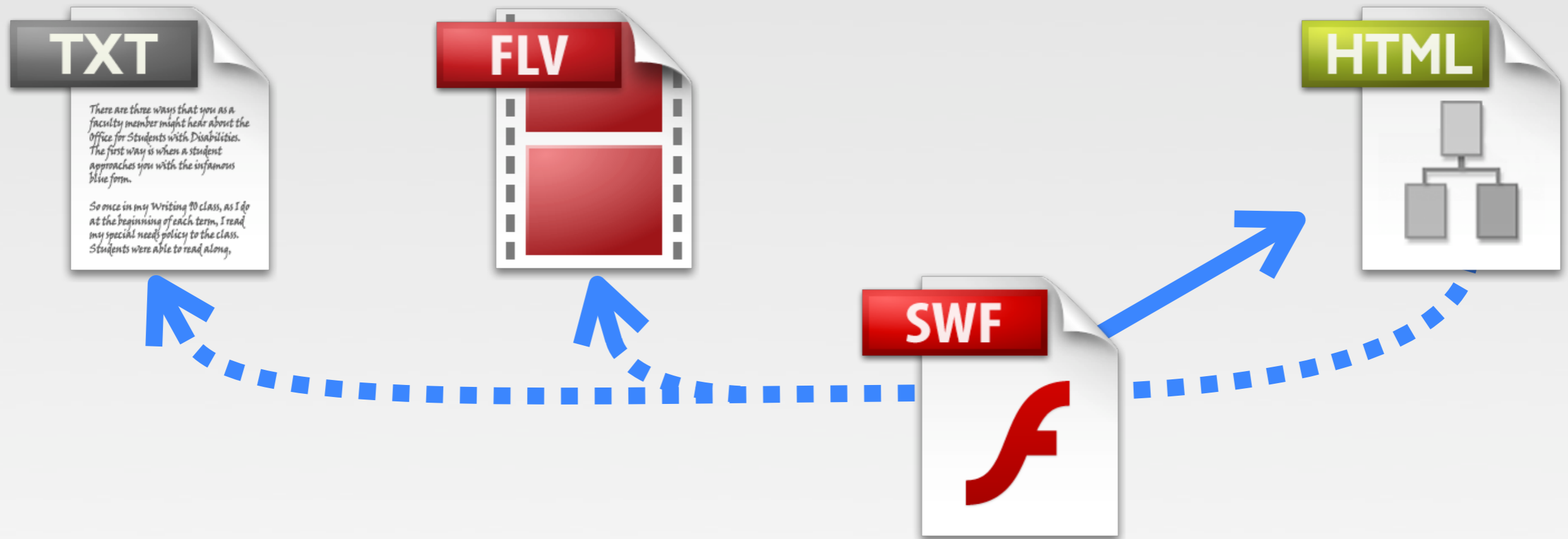
Gather the Pieces:



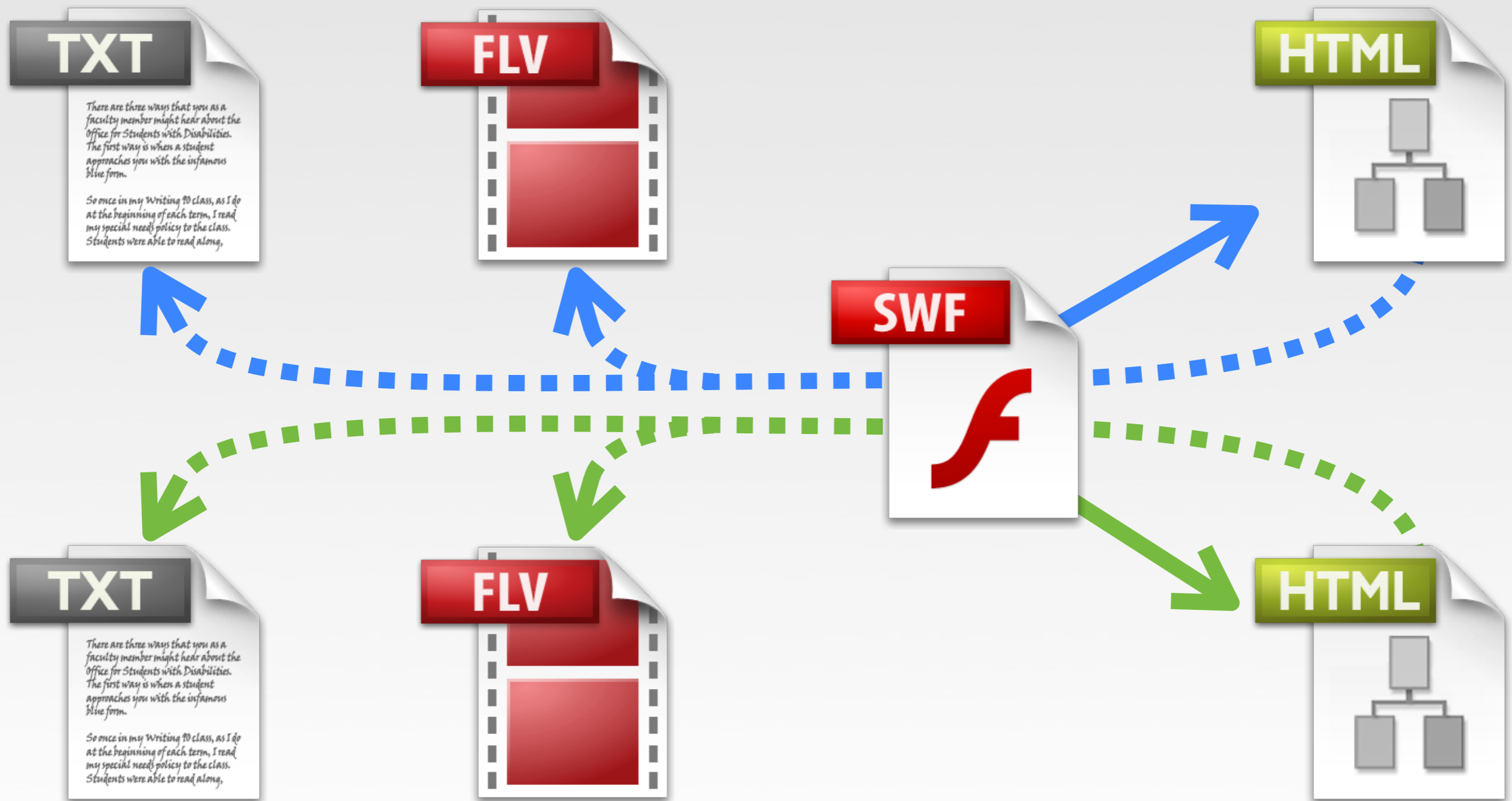
Gather the Pieces:



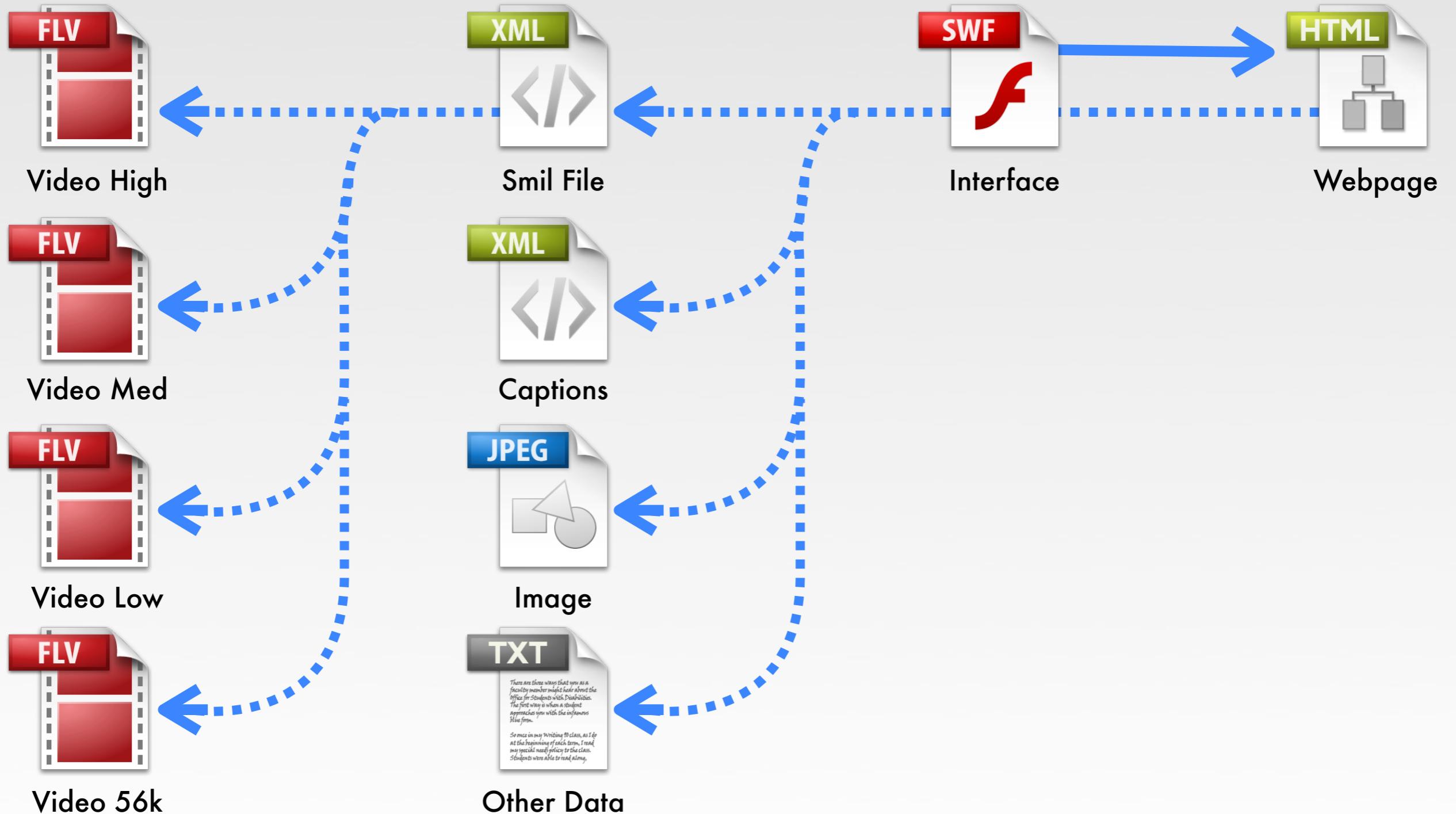
Gather the Pieces: Interface



Gather the Pieces: Interface



Gather the Pieces: Interface



Gather the Pieces: Compress the Video

Flash Video Encoder


To start encoding, drag video files into the queue or click Add.

Source File	Settings	Status
.../about/bond/media/pres-welding.mov	Flash 8 - High Quality (700kbps) - Advanced	Encoding
.../about/bond/media/pres-welding.mov	Flash 8 - Medium Quality (400kbps)	Waiting
.../about/bond/media/pres-welding.mov	Flash 8 - Low Quality (150kbps) - Advanced	Waiting
.../about/bond/media/pres-welding.mov	Flash 8 - Custom Profile - Advanced	Waiting

Add...
Duplicate
Remove
Settings...
Stop Queue

Source file: .../gabrielmcgovern/Sites/www.pcc.edu/about/bond/media/pres-welding.mov
Output file: /Users/gabrielmcgovern/Sites/www.pcc.edu/about/bond/media/pres-welding.flv
Video codec: On2 VP6 Audio codec: MPEG Layer III (MP3)
Video data rate: 700 kbps Audio data rate: 128 kbps (stereo)

Elapsed time: 00:05:45
Time left: 00:40:46



Elapsed queue time: 00:05:46



Gather the Pieces: Create the smil.xml

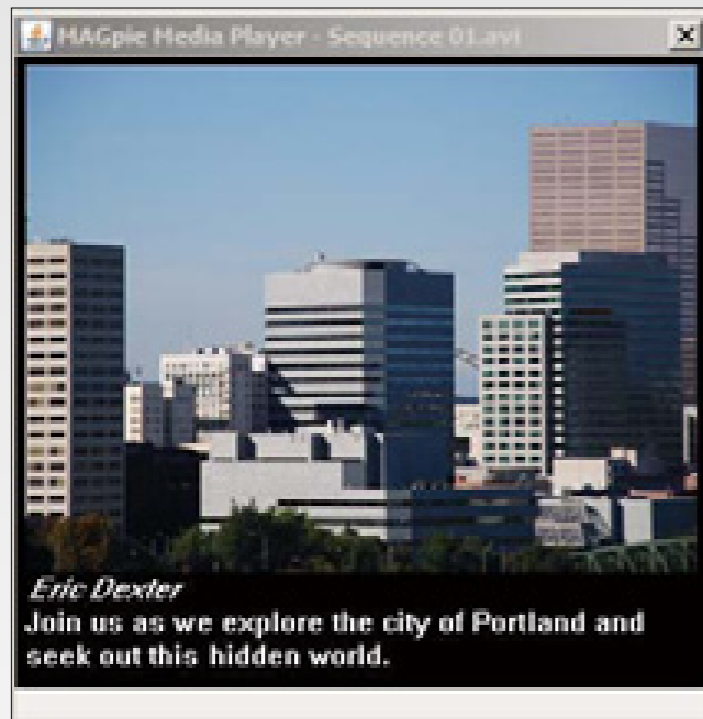
XML



```
1 <smil>
2   <head>
3     <!-- path for progressive download -->
4     <meta base="/resources/web/documentation/video/media/" />
5     <layout>
6       <root-layout width="320" height="240" />
7     </layout>
8   </head>
9   <body>
10    <switch>
11      <!-- Show video: "base+src" When users connection is: "system-bitrate"-->
12      <video src="eric-vp6-700kb.flv" system-bitrate="700000" />
13      <video src="eric-vp6-400kb.flv" system-bitrate="400000" />
14      <video src="eric-vp6-150kb.flv" system-bitrate="150000" />
15      <!-- If connection is even slower, just use the following -->
16      <ref src="dexter/eric-vp6-56kb.flv" />
17    </switch>
18  </body>
19 </smil>
```

```
<!-- path for streaming -->
<meta base="rtmp://flash-r1.vitalstreamcdn.com/portlandpcc_vitalstream_com/_definst_/public/dexter/" />
```

Gather the Pieces: Now for the Captions



Export Playback Help

XML

Plain Text...

QuickTime - SMIL 1.0 format

RealPlayer - SMIL 1.0 format

Windows Media Player - SAMI format

Adobe Flash - W3C DFXP

MAGpie - eric.maggie

File Tracks Edit Style Captions Audio Export Playback Help

0:00:00.00

Caption - Captions

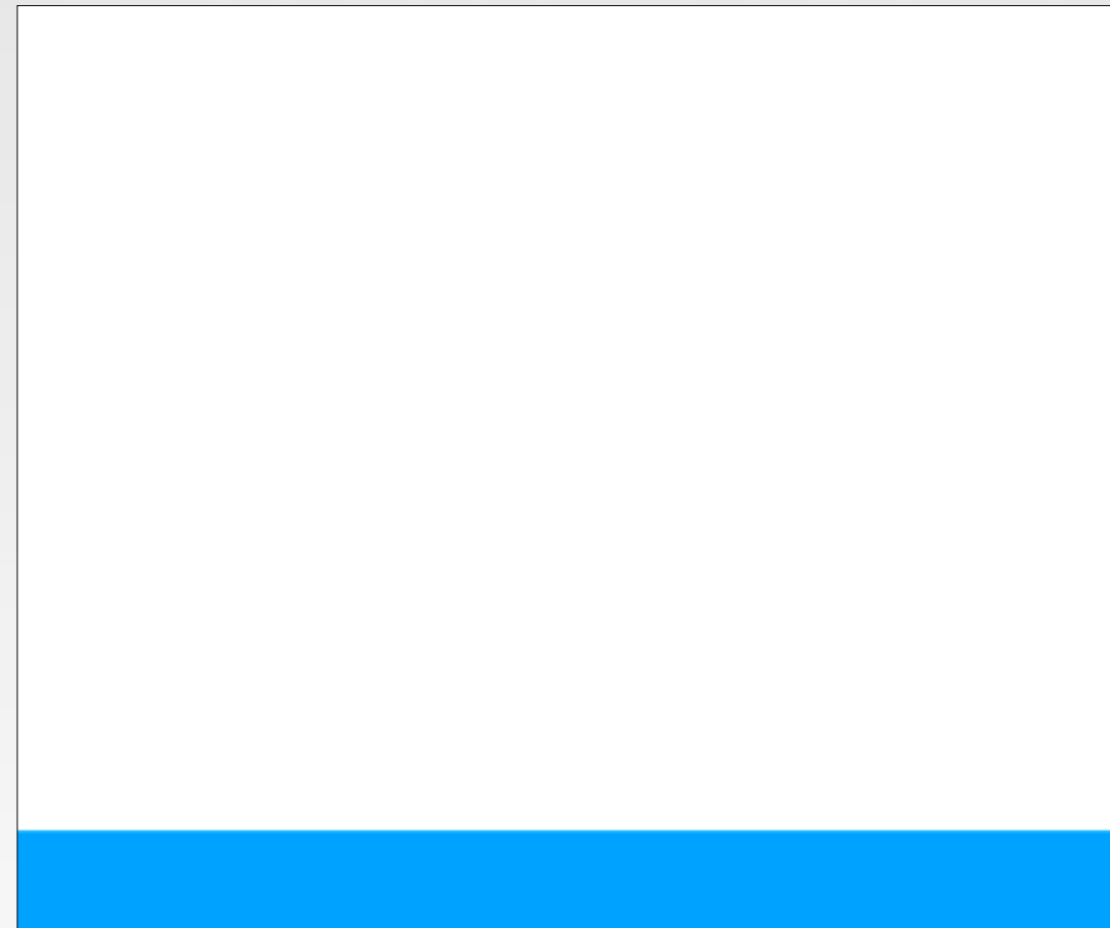
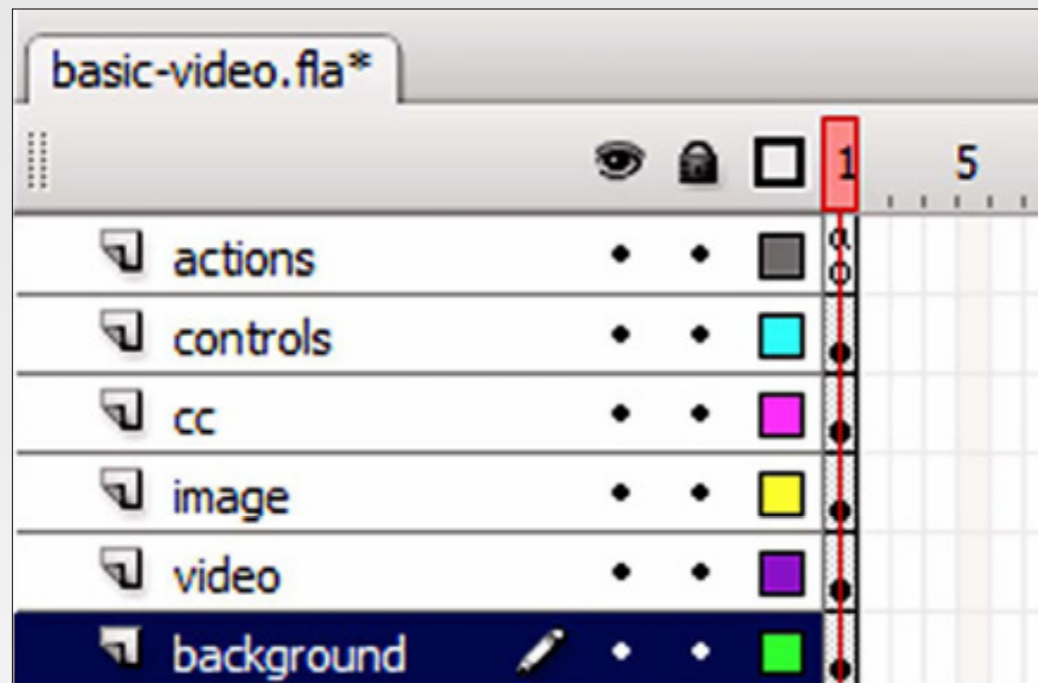
SansSerif 12 White Black

Row	Start Time	End Time	Speaker	Caption
1	0:00:00.00		Eric Dexter	J
2	0:00:02.02		Eric Dexter	The city of Portland is bustling metropolis of escalators, buses and skyscrapers.
3	0:00:08.40		Eric Dexter	However, to those who look for it, there is another hidden world ...
4	0:00:12.40		Eric Dexter	The world at our feet.
5	0:00:22.00		Eric Dexter	Join us as we explore the city of Portland and seek out this hidden world.
6	0:00:57.70		Eric Dexter	Our journey begins at Oaks Bottom wildlife refuge.
7	0:01:10.06		Eric Dexter	Found just a few minutes from the heart of Portland,
8	0:01:13.00		Eric Dexter	the refuge 140 acres of wetland reclaimed from a landfill.
9	0:01:19.50		Eric Dexter	This preserve does not just provide refuge for animals;
10	0:01:23.00		Eric Dexter	it provides refuge for humans seeking escape from the demands of city
11	0:01:48.34		Eric Dexter	We come here not as passive observers, but as participants.
12	0:02:16.25		Eric Dexter	The spider takes a defensive posture in response to our voices,
13	0:02:20.65		Eric Dexter	raising its legs with every word.
14	0:02:29.00		Eric Dexter	Watch as the spider removes a seedpod that has fallen into its web.
15	0:03:00.00		Eric Dexter	The next stop on our journey introduces us to the nutria.
16	0:03:15.00		Eric Dexter	First brought to Oregon by the fur trappers for a replacement to the beaver,

Gather the Pieces: **Poster Image**



Construct the Interface



Construct the Interface: Components



FLVPlayback



FLVPlaybackCaptioning



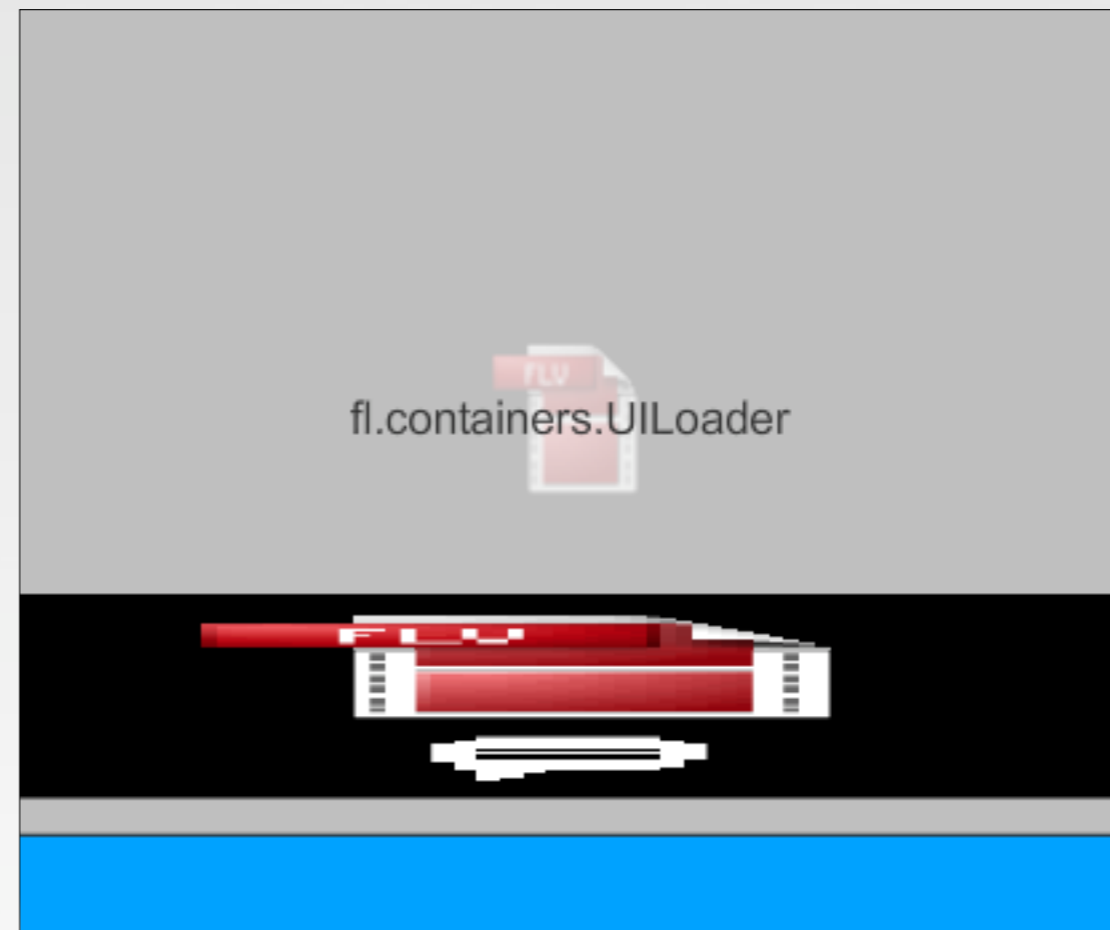
UILoader

Instance of: FLVPlaybackCaptioning

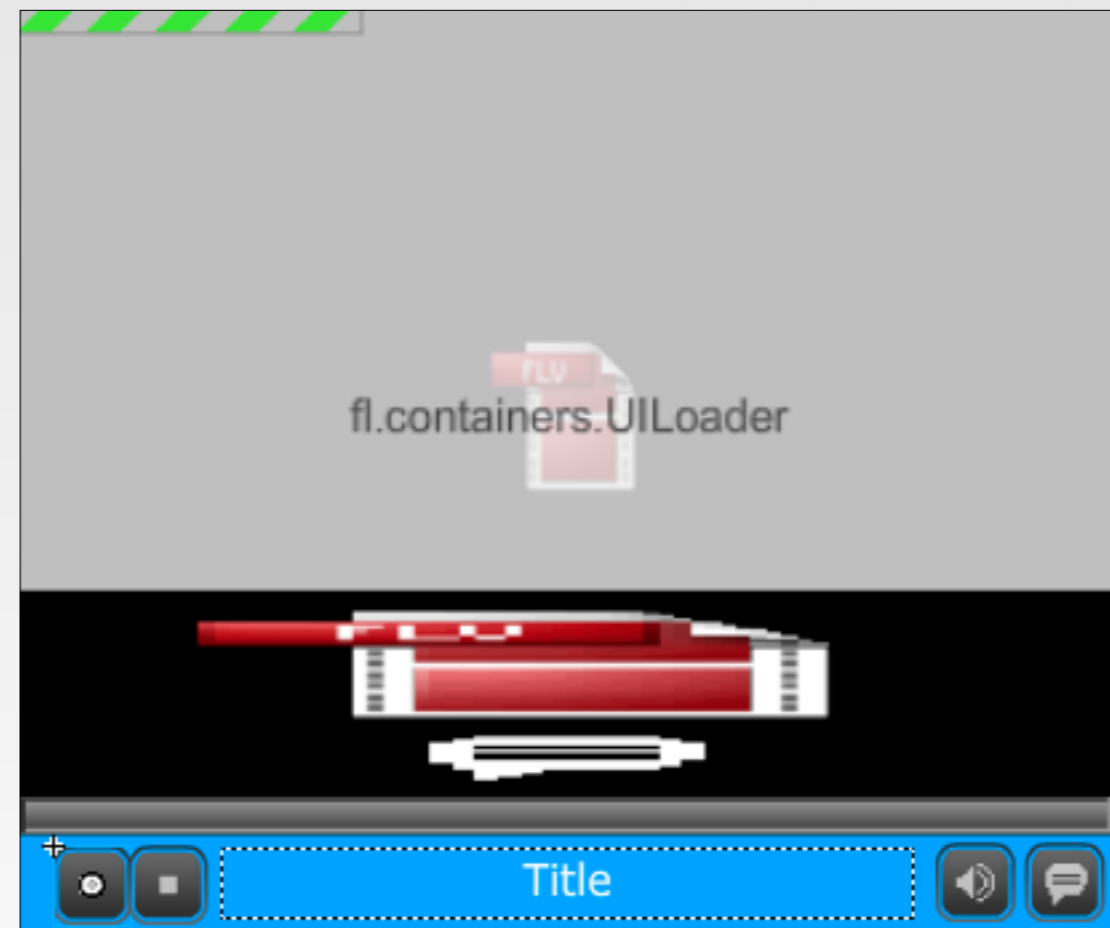
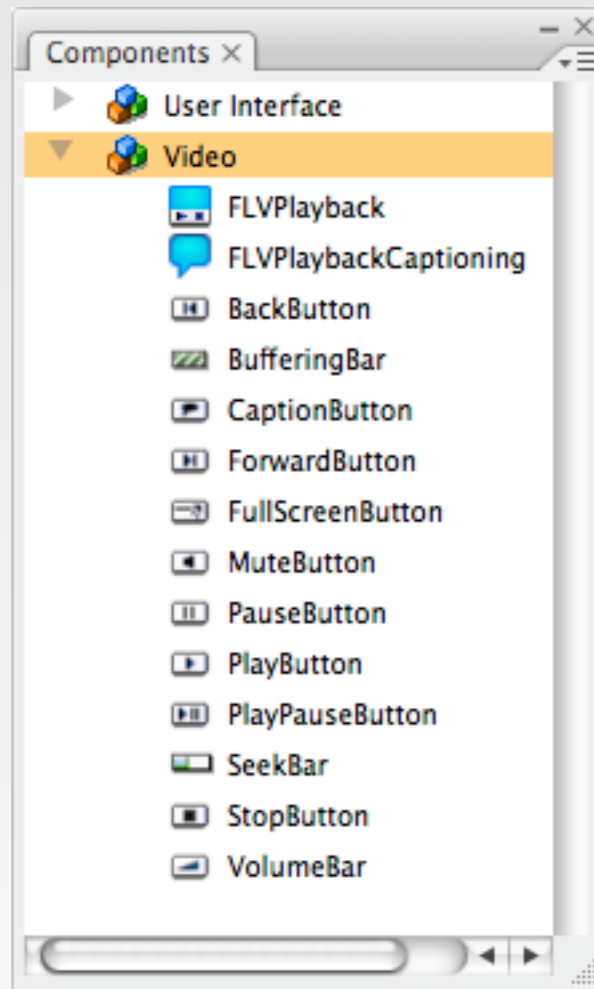
ccPane

W: 320.0 X: 0.0

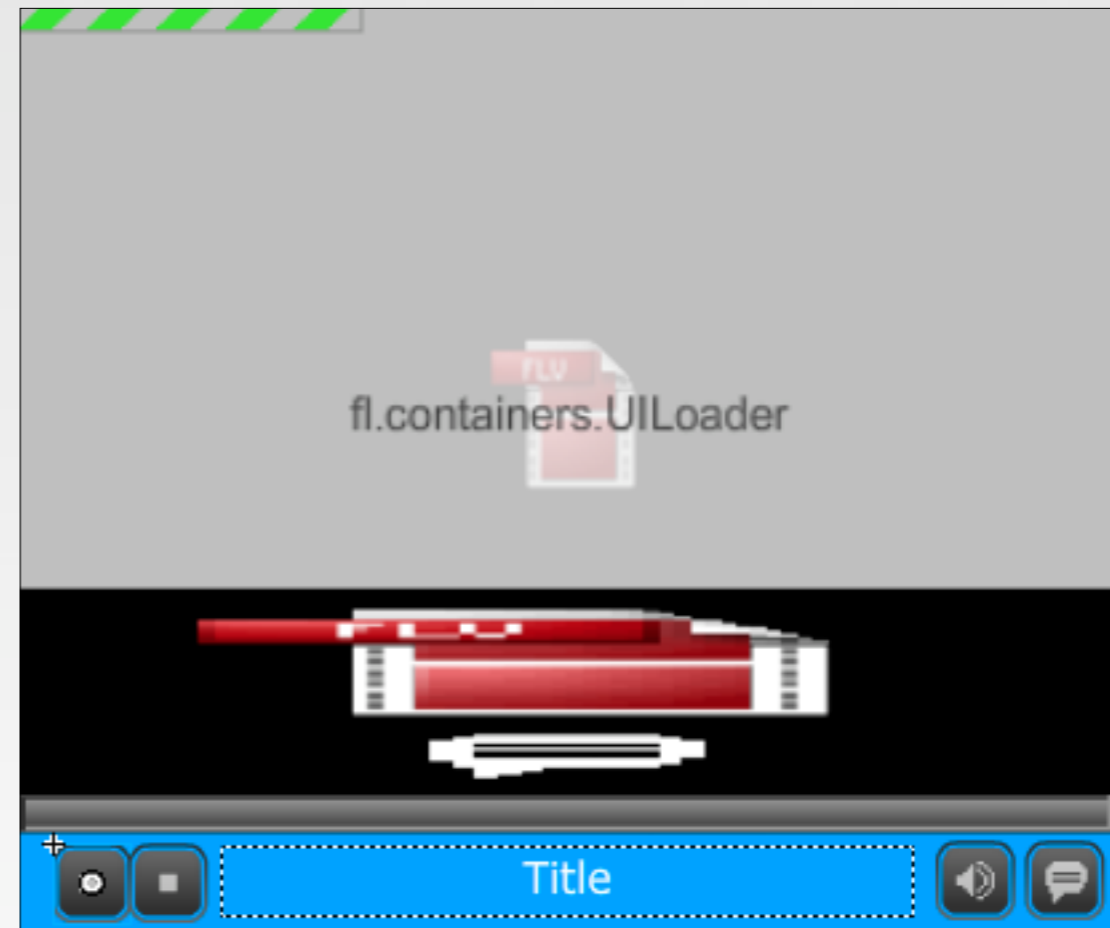
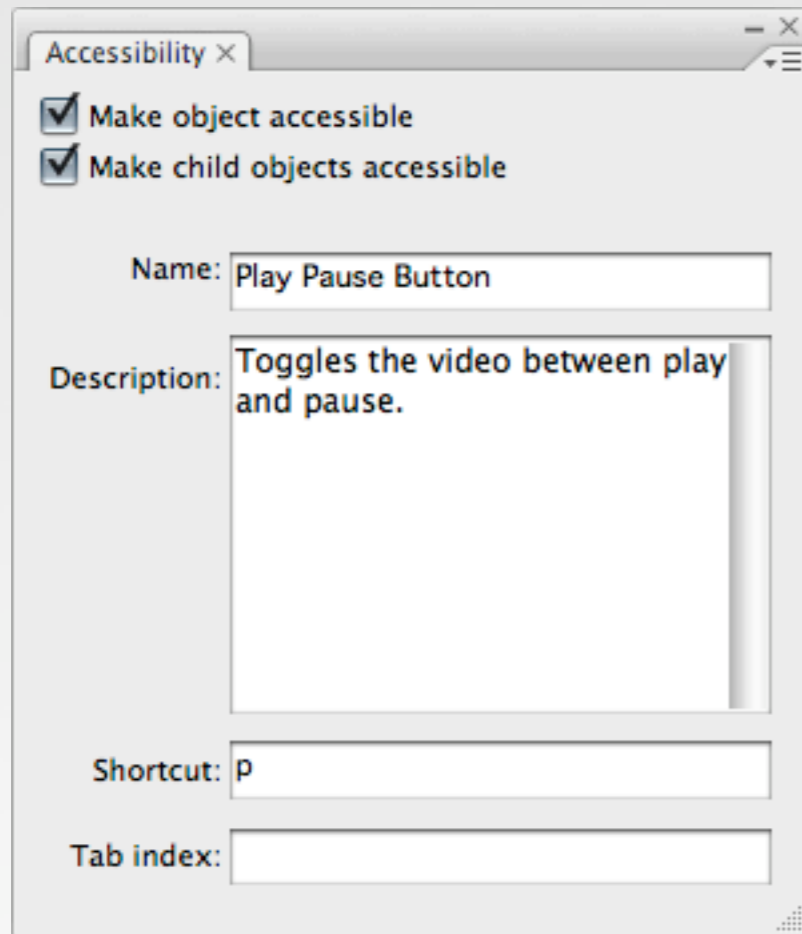
H: 60.0 Y: 170.0



Construct the Interface: Add Controls



Construct the Interface: Accessibility Panel



Construct the Interface: **ActionScript**

For what the components can't do directly:

Read variables from the html.

Set up the video and handle video events.

Estimate the bandwidth.

Add the access key behaviors.

```
72 // Set up access keys
73 function keyPressed(event:KeyboardEvent):void {
74     //trace("Key.getCode = " + event.charCode);
75     switch ( event.charCode ) {
76         case 101:
77             videoPane.stop();
78             break;
79         case 112:
80             if(videoPane.state == "playing")
81                 videoPane.pause();
82             }
83         else{
84             videoPane.play();
85         }
86         break;
87         case 114:
88             videoPane.seek(videoPane.playheadTime);
89             break;
90     }
91 }
92 stage.addEventListener(KeyboardEvent.KEY_DOWN, keyPressed);
```

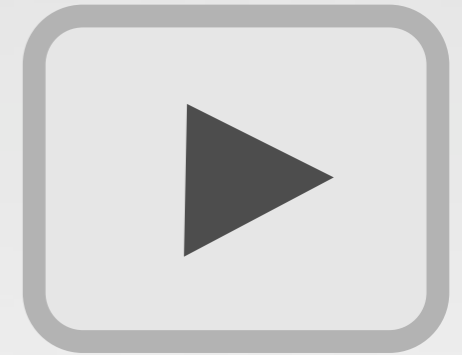
Beyond the basic



Share and share alike

www.gabrielmcgovern.com

The sample files, source code and more.



Special thanks to:

The National Center for Accessible Media

Geoff Stearns, the author of SWFObject

Robert Reinhardt for his quick help on the GoToAndPlay forum

Everyone on the PCC web team

Screenshots and data:

Chart data from 'Power to the PeoPle SoCIAI MeDIA trACKer' by www.universalmccann.com

www.youtube.com, tv.boingboing.net, www.comedycentral.com

Remix icon from creativecommons.org