Flash, Video & Audio

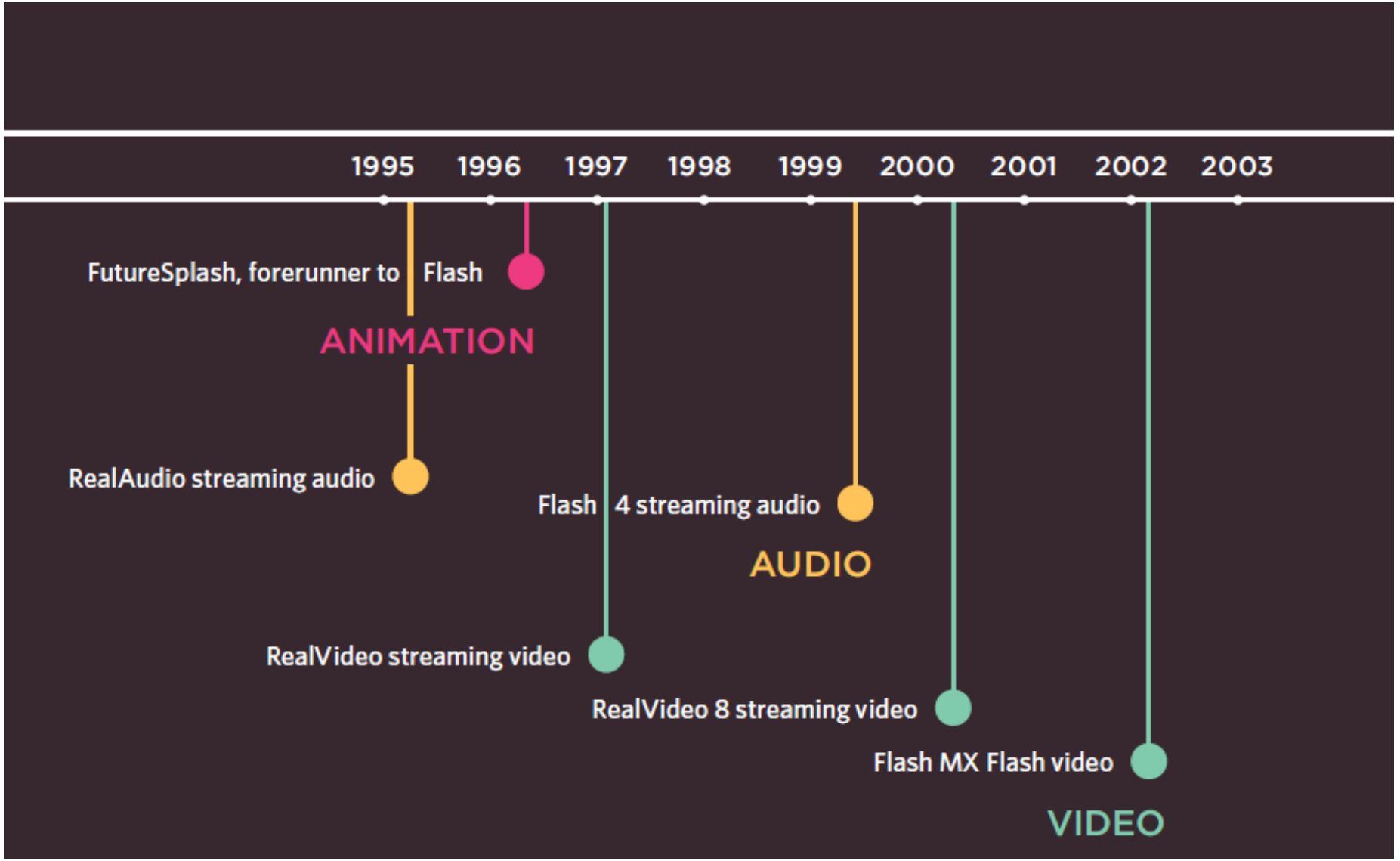
Flash is a very popular technology used to add animations, video, and audio to websites. This chapter begins by looking at how to use it in your web pages.

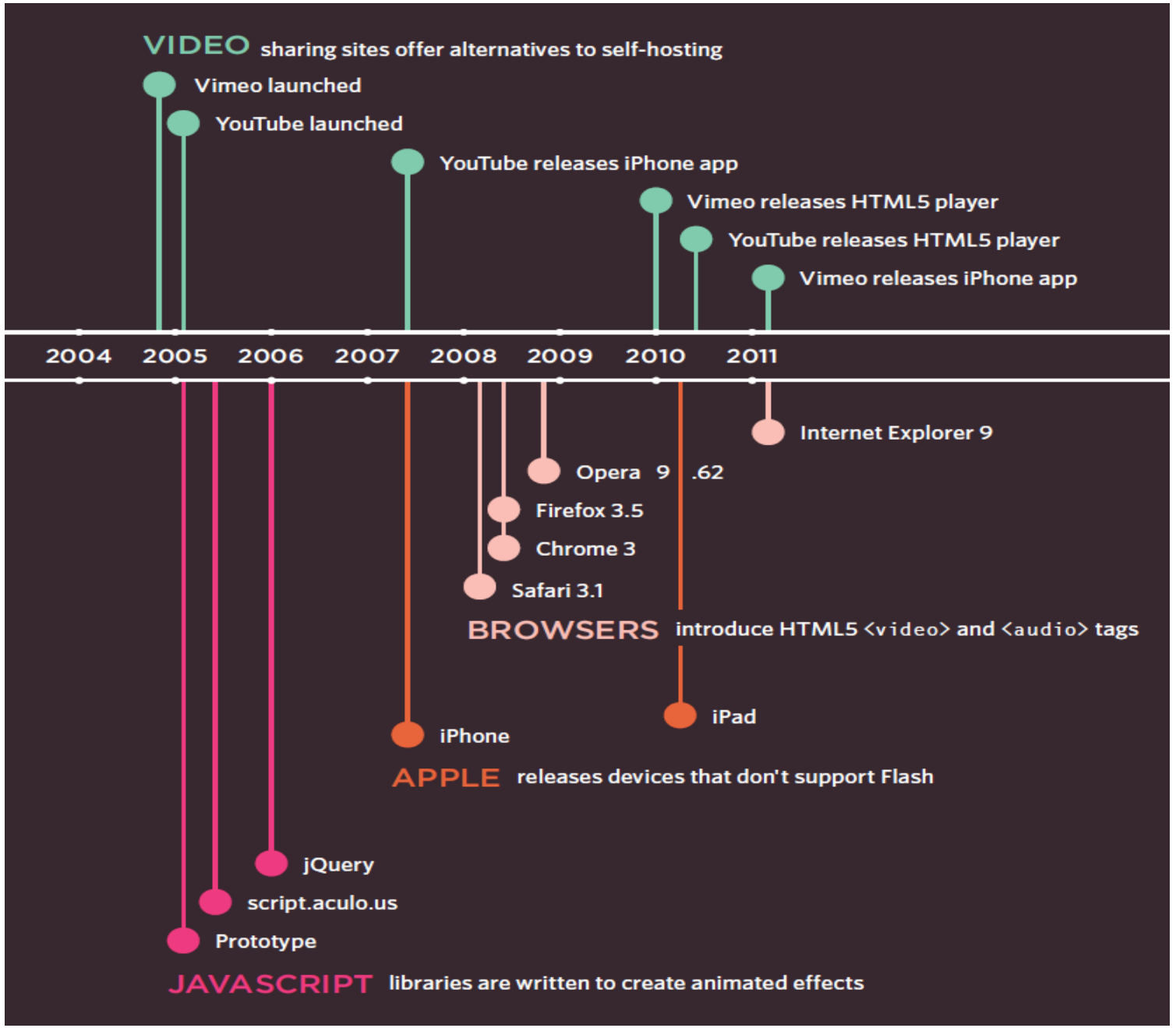
Since the late 1990s, Flash has been a very popular tool for creating animations, and later for playing audio and video in websites.

When you create a Flash file in the Flash authoring environment, it is saved with the .fla file extension.

Flash using

- **Since 2005, a number of factors have meant that fewer websites are written in Flash or even use elements of Flash in their pages.**
- **In 2005-6, a set of JavaScript libraries were launched which made it easier for people to create animated effects using JavaScript.**
- **When Apple launched the iPhone in 2007 and later the the iPad in 2010, they took the decision not to support Flash.**
- **In 2008, browsers started to support HTML5 <video> and <audio> tags**


1995
1996
1997
1998
1999
2000
2001
2002
2003
FutureSplash, forerunner to Flash
ANIMATION
RealAudio streaming audio
Flash 4 streaming audio
AUDIO
RealVideo streaming video
RealVideo 8 streaming video
Flash MX Flash video
VIDEO


VIDEO sharing sites offer alternatives to self-hosting
Vimeo launched
YouTube launched
YouTube releases iPhone app
Vimeo releases HTML5 player
YouTube releases HTML5 player
Vimeo releases iPhone app
2004
2005
2006
2007
2008
2009
2010
2011
Internet Explorer 9
Opera 9 .62
Firefox 3.5
Chrome 3
Safari 3.1
BROWSERS introduce HTML5 <video> and <audio> tags
iPad
iPhone
APPLE releases devices that don't support Flash
jQuery
script.aculo.us
Prototype
JAVASCRIPT libraries are written to create animated effects

Adding a Flash Movie to Your Web Page

```
<!DOCTYPE html>
<html>
  <head>
    <title>Adding a Flash Movie</title>
    <script type="text/javascript"
      src="http://ajax.googleapis.com/ajax/libs/
      swfobject/2.2/swfobject.js"></script>
    <script type="text/javascript">
      swfobject.embedSWF("flash/bird.swf",
      "bird", "400", "300", "8.0.0");</script>
  </head>
  <body>
    <div id="bird"><p>An animation of a bird taking
      a shower</p></div>
  </body>
</html>
```

In this example, the SWFObjectscript is hosted on Google's servers. We **include** the script in this web page using the first of the two **<script>** elements. The type attribute is used on the **<script>** element to indicate that the script inside is written in JavaScript. The **src** attribute tells the browser where to find the script. The second **<script>** element is used to tell the browser about the Flash movie, as well as which element it should replace. This element is actually telling the

chapter-09/adding-a-flash-movie.html HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Adding a Flash Movie</title>
    <script type="text/javascript"
      src="http://ajax.googleapis.com/ajax/libs/
      swfobject/2.2/swfobject.js"></script>
    <script type="text/javascript">
      swfobject.embedSWF("flash/bird.swf",
      "bird", "400", "300", "8.0.0");</script>
  </head>
  <body>
    <div id="bird"><p>An animation of a bird taking
      a shower</p></div>
  </body>
</html>
```

SWFObject script **five** pieces of information, which are in the brackets:

1. The **location** of the **.swf** file:
flash/bird.swf
2. The element that the Flash movie should **replace**, specified by the **value** of the **id** attribute on the ** ** element:
bird
3. The **width** of the Flash movie:
400 px
4. The **height** of the Flash movie:
300 px
5. The minimum **version** of the Flash player needed to view the movie:
Flash Player 8

Understanding Video Formats and Players

Format

Movies are available in many formats (BluRay, DVD, VHS, to name a few). Online, there are even *more* video formats\ (including AVI, Flash Video, H264, MPEG, Ogg Theora, QuickTime, WebM, and Windows Media).

Players

Recently browsers have evolved to support the HTML5 **<video>** tag (which renders players and plugins obsolete).

Using Hosted Video Services

The easiest way to add a video to your site is to upload the video to a site like YouTube or Vimeo and use the features provided on their site to embed the video in your page.

Advantage and disadvantage

- Hosted video sites (such as YouTube) provide players that work with the majority of web browsers.
- You do not need to worry about encoding
- Web hosting companies often charge extra if you use a lot of bandwidth.
- Your video will be available on the site of the hosted service,.
- Some services will limit what your video is allowed to include.
- Some hosted services will play their own adverts before your video will begin

Preparing a Flash Video for Your Site

1

CONVERT YOUR VIDEO INTO FLV FORMAT

2

FIND AN FLV PLAYER TO PLAY THE VIDEO

3

INCLUDE THE PLAYER & VIDEO IN YOUR PAGE

HTML chapter-09/adding-html5-video.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Adding HTML5 Video</title>
  </head>
  <body>
    <video src="video/puppy.mp4"
      poster="images/puppy.jpg"
      width="400" height="300"
      preload
      controls
      loop>
      <p>A video of a puppy playing in the snow</p>
    </video>
  </body>
</html>
```

In HTML5 you do not need to supply values for all attributes, such as the controls, autoplay, and loop attributes used with the <video> element. These attributes are like on/off switches. If the attribute is present, it turns that option on. If the attribute is omitted, the option is turned off.

If the browser does not support the <video> element or the format of video used, it will display whatever is between the opening <video> and closing </video> tags.

preload

This attribute tells the browser what to do when the page loads. It can have one of three values:

none

The browser should not load the video until the user presses play.

auto

The browser should download the video when the page loads.

metadata

The browser should just collect information such as the size, first frame, track list, and duration.

<video>

The <video> element has a number of attributes which allow you to control video playback:

src

This attribute specifies the path to the video. (The example video is in H264 format so it will only work in IE and Safari.)

poster

This attribute allows you to specify an image to show while the video is downloading or until the user tells the video to play.

width, height

These attributes specify the size of the player in pixels.

controls

When used, this attribute indicates that the browser should supply its own controls for playback.

autoplay

When used, this attribute specifies that the file should play automatically.

loop

When used, this attribute indicates that the video should start playing again once it has ended.

<source>

To specify the location of the file to be played, you can use the <source> element inside the <video> element. (This should replace the src attribute on the opening <video> tag.)

You can also use multiple <source> elements to specify that the video is available in different formats.

(Due to a bug on the iPad, you should provide the MP4 video as the first format. Otherwise, it might not play.)

src

This attribute specifies the path to the video.

type

You should use this attribute to tell the browser what format the video is. Otherwise, it will download some of the video to see if it can play the file (which will take time and bandwidth).

chapter-09/multiple-video-sources.html **HTML**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Multiple Video Sources</title>
  </head>
  <body>
    <video poster="images/puppy.jpg" width="400"
      height="320" preload controls loop>
      <source src="video/puppy.mp4" type='video/
        mp4;codecs="avc1.42E01E, mp4a.40.2"' />
      <source src="video/puppy.webm" type='video/
        webm;codecs="vp8, vorbis"' />
      <p>A video of a puppy playing in the snow</p>
    </video>
  </body>
</html>
```

RESULT

codecs

The codec that was used to encode the video is supplied within the type attribute. Note the use of single quotes, as well as double quotes in the type attribute, when it is supplied.

If the browser does not support the <video> element or the format of video used, it will display whatever is between the opening <video> and closing </video> tags.

Adding HTML 5 Audio to Your Pages

<audio>

HTML5 introduced the <audio> element to include audio files in your pages. As with HTML5 video, browsers expect different formats for the audio.

The <audio> element has a number of attributes which allow you to control audio playback:

src

This attribute specifies the path to the audio file.

controls

This attribute indicates whether the player should display controls. If you do not use this attribute, no controls will be shown by default. You can also specify your own controls using JavaScript.

autoplay

preload

chapter-09/adding-html5-audio.html **HTML**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Adding HTML5 Audio</title>
  </head>
  <body>
    <audio src="audio/test-audio.ogg"
      controls autoplay>
      <p>This browser does not support our audio
      format.</p>
    </audio>
  </body>
</html>
```

RESULT

HTML chapter-09/multiple-audio-sources.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Multiple Audio Sources</title>
  </head>
  <body>
    <audio controls autoplay>
      <source src="audio/test-audio.ogg" />
      <source src="audio/test-audio.mp3" />
      <p>This browser does not support our audio
      format.</p>
    </audio>
  </body>
</html>
```

RESULT

src type

<source>

It is possible to specify more than one audio file using the <source> element between the opening <audio> and closing </audio> tags (instead of the src attribute on the opening <audio> tag).

This is important because different browsers support different formats for audio files.

MP3: Safari 5+, Chrome 6+, IE9

Ogg Vorbis: Firefox 3.6, Chome 6, Opera 1.5, IE9

So you would need to supply two audio formats to get coverage across all recent browsers that support the <audio> element. You could also provide a Flash alternative for older browsers that do not support the <audio> element.

Exercise

Video Element:

Create a new HTML file named video.html.

Insert a video file named video.mp4 in the same directory as your HTML file.

Add a <video> element with width, height, and controls attributes to video.html.

Test the video player in your web browser to ensure it works correctly.

Audio Element:

Create a new HTML file named audio.html.

Insert an audio file named audio.mp3 in the same directory as your HTML file.

Add an <audio> element with controls attribute to audio.html.

Test the audio player in your web browser to ensure it works correctly.