

Advanced Effects with HTML5 Media Technologies

Session 604

Vicki Murley

Safari Technologies Evangelist

These are confidential sessions—please refrain from streaming, blogging, or taking pictures



<audio>



<video>

HTML5



`<audio>`



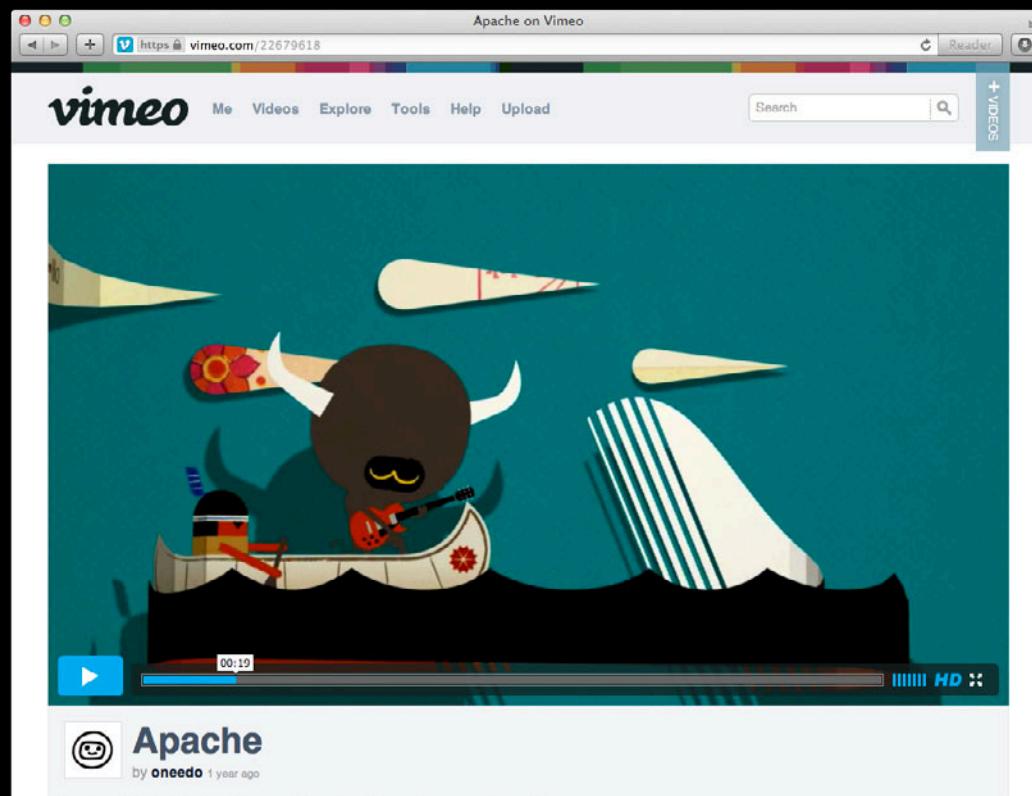
`<video>`

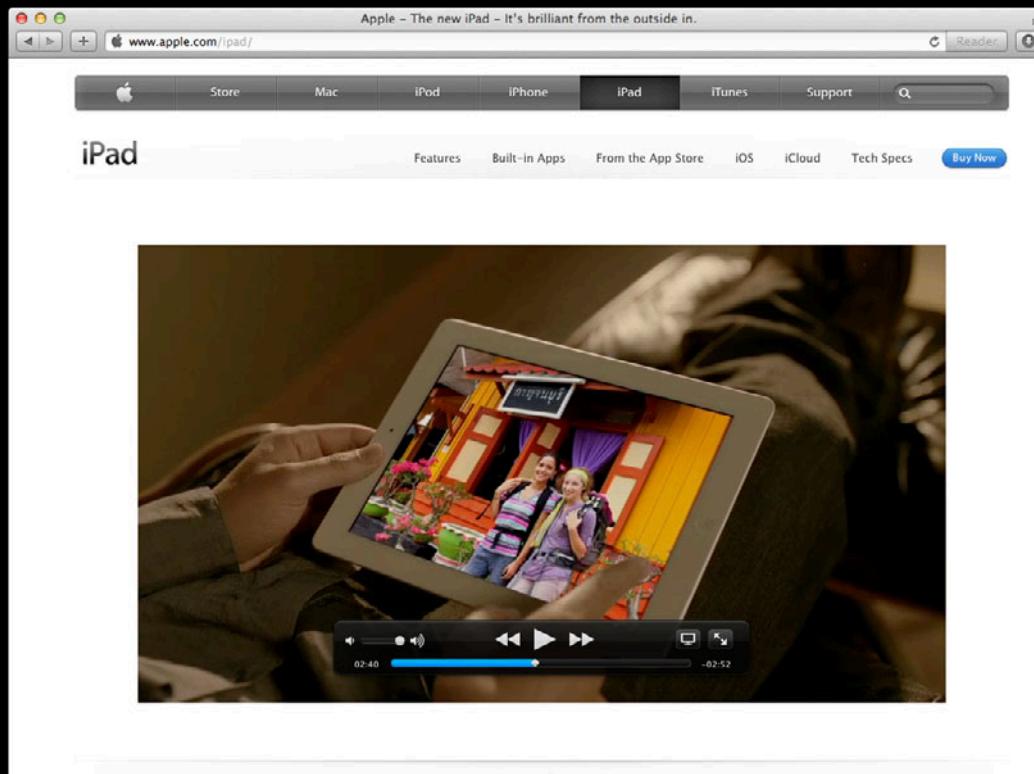
HTML5

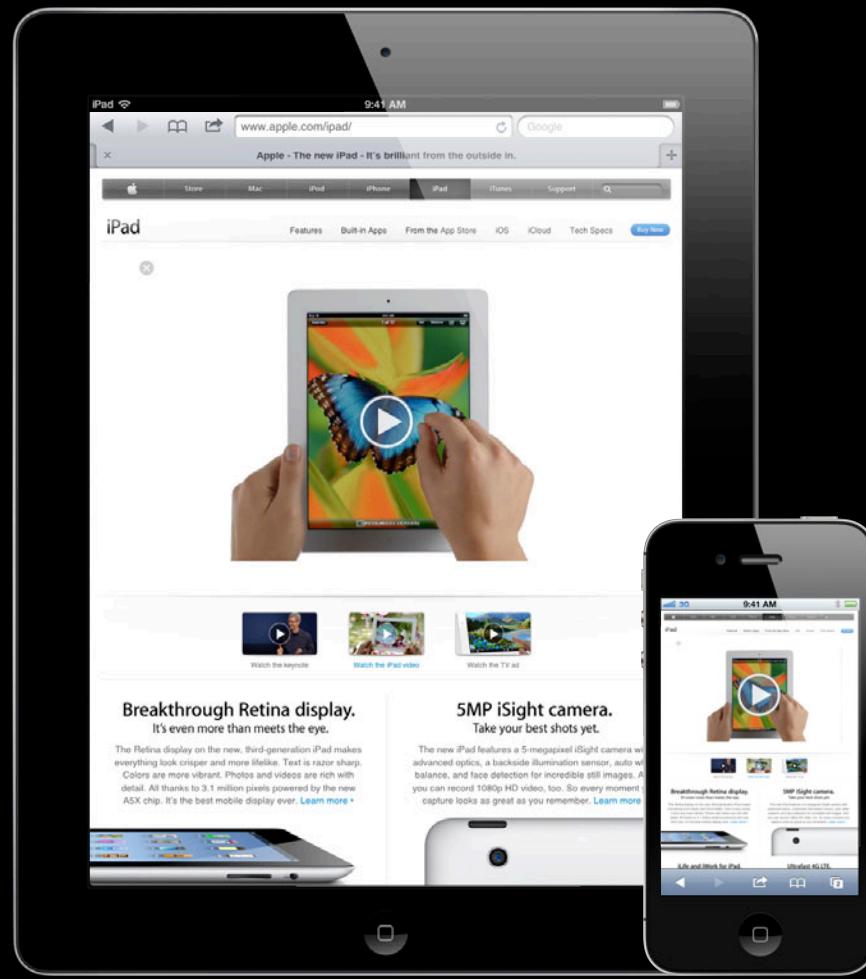


<audio> <video>









HTML5



@!#%?!





New Ways to Integrate Media with HTML5

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

Augment your interface with audio

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

Augment your interface with audio

When Perfect Sync Is Required

Sophisticated decoration

Untitled

<http://sharksareso cool.com>

Sharks

Sharks are a group of fishes characterized by a cartilaginous skeleton, five to seven gill slits on the sides of the head, and pectoral fins that are not fused to the head. Modern sharks are classified within the clade Selachimorpha (or Selachii), and are the sister group to the rays. However, the term "shark" has also been used for extinct members of the suborder Elasmobranchii outside the Selachimorpha, such as Cladoselache and Xenacanthus. Under this broader definition, the earliest known sharks date from more than 420 million years ago.

Since that time, sharks have diversified into over 400 species. They range in size from the small dwarf lanternshark (*Etmopterus perryi*), a deep sea species of only 17 centimetres (6.7 in) in length, to the whale shark (*Rhincodon typus*), the largest fish in the world, which reaches approximately 12 metres (39 ft). Despite its size, the whale shark feeds only on plankton, squid, and small fish by filter feeding. Sharks are



When Perfect Sync Is Required

Adding a sign language track



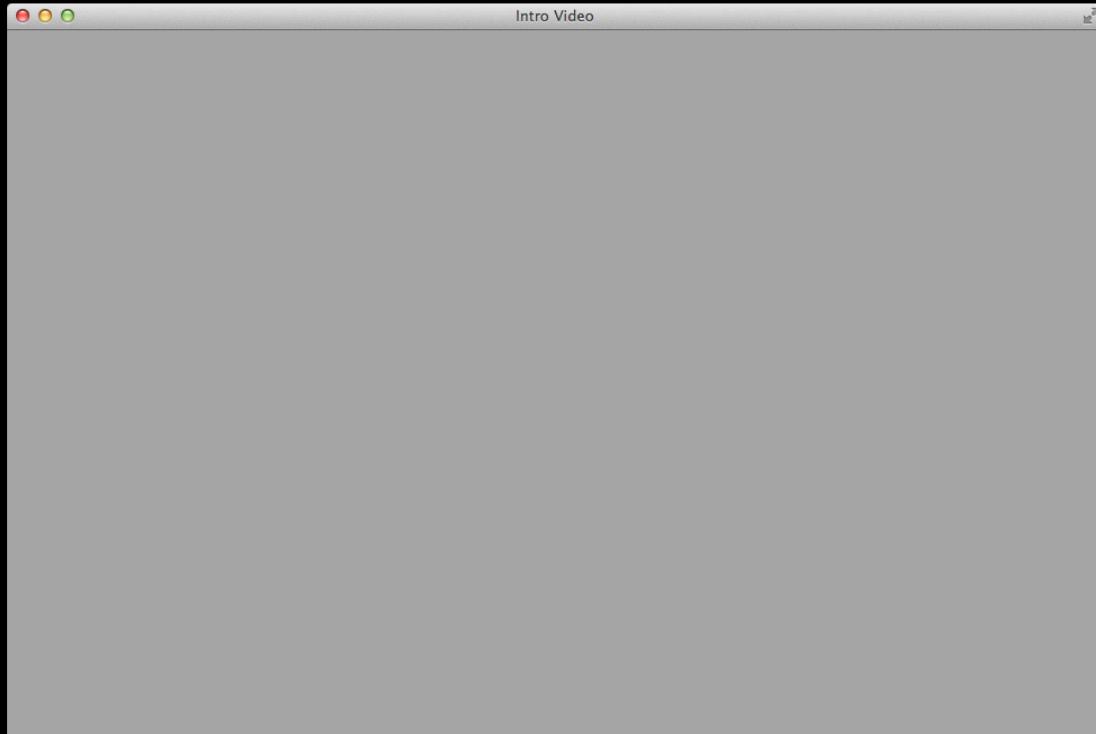
When Perfect Sync Is Required

Comparing two similar events



When Perfect Sync Is Required

Substituting an alternate audio track



When Perfect Sync Is Required

Substituting an alternate audio track



Keeping Playback in Perfect Sync

Maybe I'll try...

- Starting the video

```
// user has tapped PLAY
var myVideos = document.getElementsByTagName("video");
for (int i = 0; i < myVideos.length; i++) {
    myVideos[i].play();
}
```

Keeping Playback in Perfect Sync

Maybe I'll try...

- Starting the video

```
// user has tapped PLAY
var myVideos = document.getElementsByTagName("video");
for (int i = 0; i < myVideos.length; i++) {
    myVideos[i].play();
}
```

- Script execution takes a finite, unpredictable amount of time
- Media playback is not blocked by script execution

Keeping Playback in Perfect Sync

Other challenges

Keeping Playback in Perfect Sync

Other challenges

- Determining when all videos have enough data buffered to play

Keeping Playback in Perfect Sync

Other challenges

- Determining when all videos have enough data buffered to play
- Changing `playbackRate`, for slow motion or fast-forwarding

Keeping Playback in Perfect Sync

Other challenges

- Determining when all videos have enough data buffered to play
- Changing `playbackRate`, for slow motion or fast-forwarding
- Dealing with unexpected stalls on a single video

Media Controllers

New in HTML5 and Safari 6.0



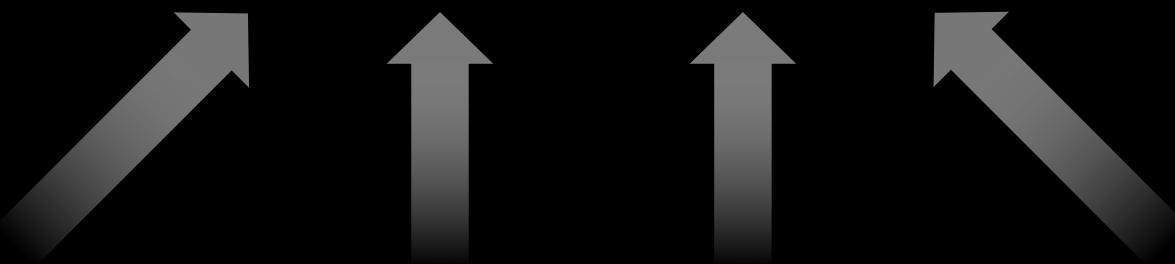
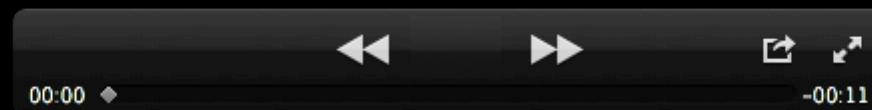
Media Controllers

New in HTML5 and Safari 6.0



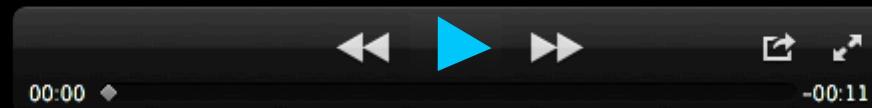
Media Controllers

New in HTML5 and Safari 6.0



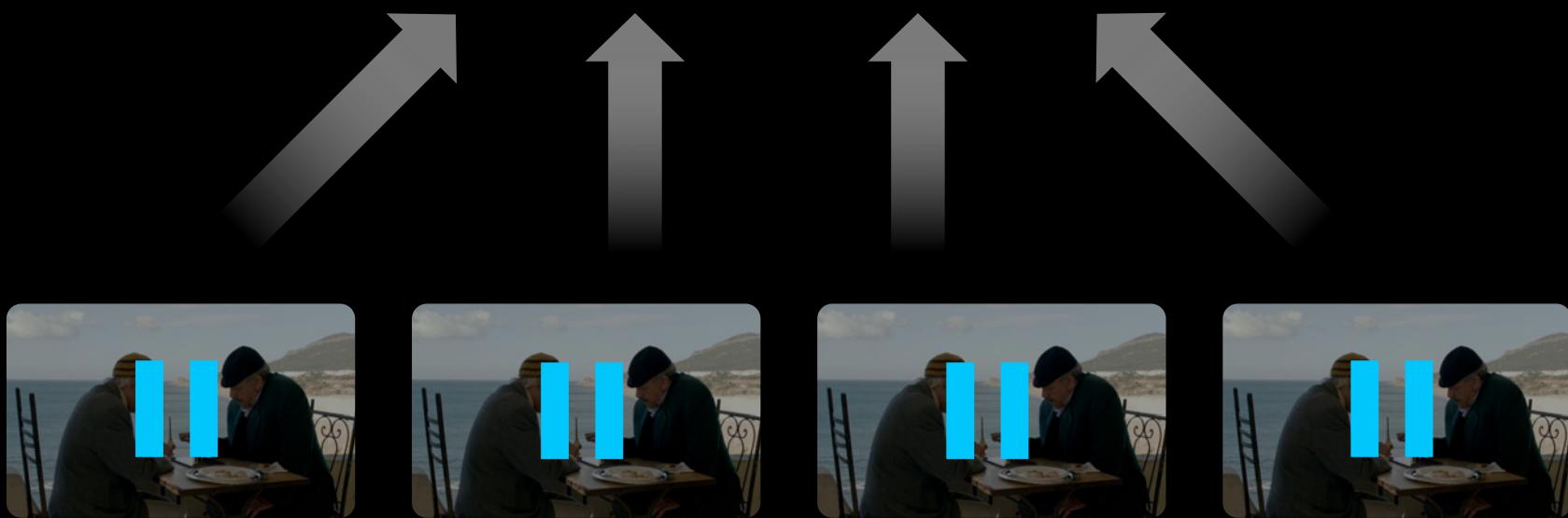
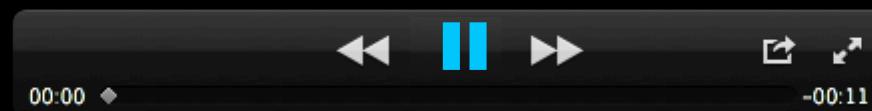
Media Controllers

New in HTML5 and Safari 6.0



Media Controllers

New in HTML5 and Safari 6.0



Media Controllers

Use familiar <audio> and <video> API

Media Controllers

Use familiar <audio> and <video> API

Functions

```
play()  
pause()  
canPlayType()
```

Media Controllers

Use familiar <audio> and <video> API

Functions

play()
pause()
canPlayType()

Attributes

currentTime
duration
ended
muted
paused
playbackRate
played
volume

Media Controllers

Use familiar <audio> and <video> API

Functions

play()
pause()
canPlayType()

Attributes

currentTime
duration
ended
muted
paused
playbackRate
played
volume

Events

abort
canplay
canplaythrough
durationchange
emptied
ended
error
loadeddata
loadedmetadata
loadstart
...

Media Controllers

Use familiar <audio> and <video> API

Functions

play()
pause()
canPlayType()

Attributes

currentTime
duration
ended
muted
paused
playbackRate
played
volume
loop
autoplay

Events

abort
canplay
canplaythrough
durationchange
emptied
ended
error
loadeddata
loadedmetadata
loadstart
...

Media Controllers

Use familiar <audio> and <video> API

Functions

play()
pause()
canPlayType()

Attributes

currentTime
duration
ended
muted
paused
playbackRate
played
volume
~~loop~~
~~autoplay~~

Events

abort
canplay
canplaythrough
durationchange
emptied
ended
error
loadeddata
loadedmetadata
loadstart
...

Setting up a Media Controller

In HTML

Setting up a Media Controller

In HTML



Setting up a Media Controller

In HTML

- Set the `mediagroup` attribute on all slaved elements



Setting up a Media Controller

In HTML

- Set the **mediagroup** attribute on all slaved elements

```
<video src="bohemian-rhapsody.mp4" width="1280" height="720">  
  Sorry, your browser can't display the HTML5 video element.  
</video>
```

```
<video src="sign-language.mp4" width="300" height="200">  
</video>
```

Setting up a Media Controller

In HTML

- Set the `mediagroup` attribute on all slaved elements

```
<video src="bohemian-rhapsody.mp4" width="1280" height="720"  
mediagroup="videoController" >  
Sorry, your browser can't display the HTML5 video element.  
</video>
```

```
<video src="sign-language.mp4" width="300" height="200"  
mediagroup="videoController" >  
</video>
```

Setting up a Media Controller

In HTML

- Set the `mediagroup` attribute on all slaved elements

```
<video src="bohemian-rhapsody.mp4" width="1280" height="720"  
mediagroup="videoController" >  
    Sorry, your browser can't display the HTML5 video element.  
</video>
```

```
<video src="sign-language.mp4" width="300" height="200"  
mediagroup="videoController" >  
</video>
```

- Access the controller on any slaved element to control them all

```
var myVideo = document.getElementsByTagName("video")[0];  
  
var myController = myVideo.controller;  
  
myController.play();  
myController.pause();
```

Setting up a Media Controller

In JavaScript

- Create a media controller object

```
var myController = new MediaController();
```

Setting up a Media Controller

In JavaScript

- Create a media controller object

```
var myController = new MediaController();
```

- Set the controller on any media element

```
document.querySelector("video").controller = myController;  
document.getElementsByTagName("video")[1].controller = myController;
```

Setting up a Media Controller

In JavaScript

- Create a media controller object

```
var myController = new MediaController();
```

- Set the controller on any media element

```
document.querySelector("video").controller = myController;  
document.getElementsByTagName("video")[1].controller = myController;
```

- Control all media with a single controller

```
myController.play();  
myController.pause();
```

Demo

Synchronizing video playback with a media controller

Beth Dakin

Safari and WebKit Engineer

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

Augment your interface with audio

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

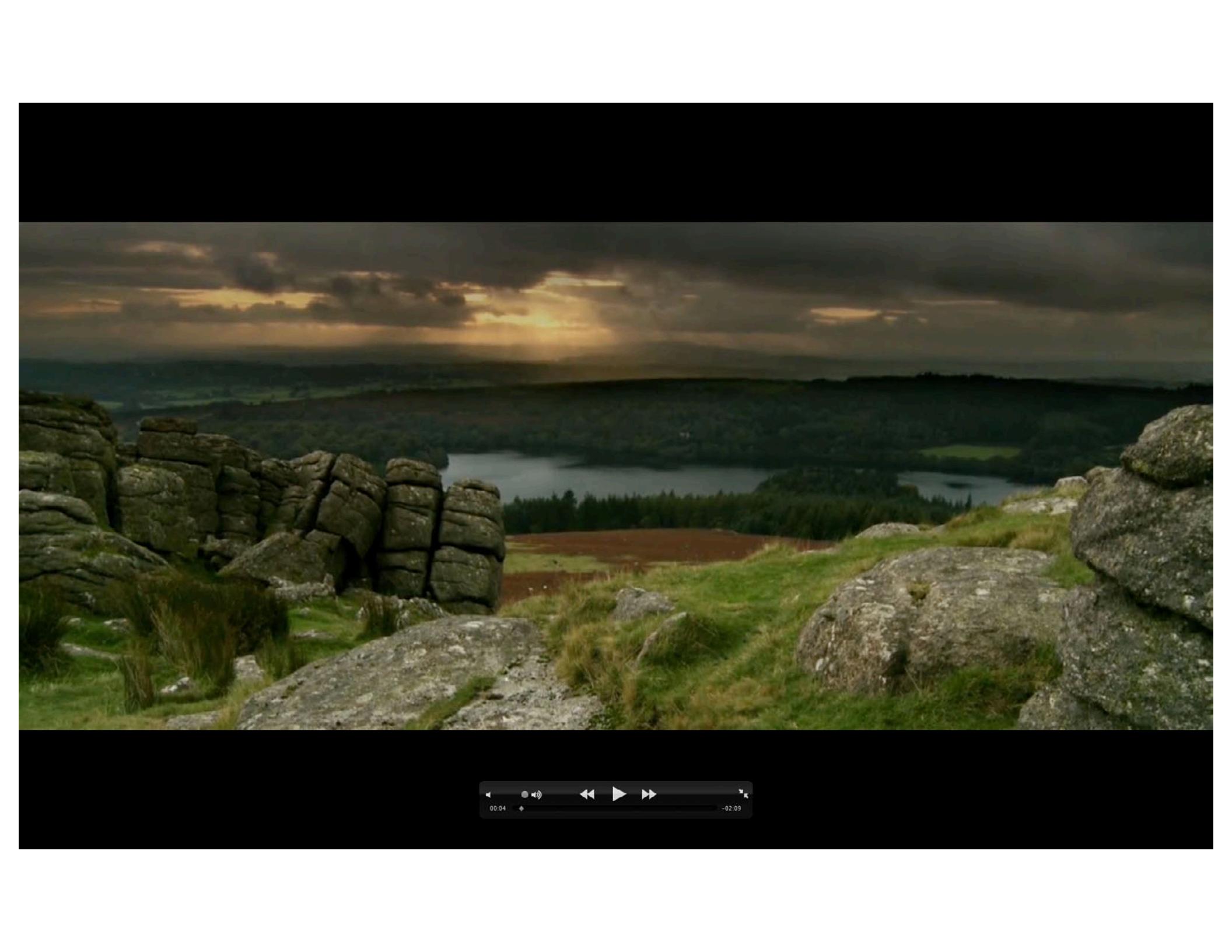
Implement fullscreen interfaces

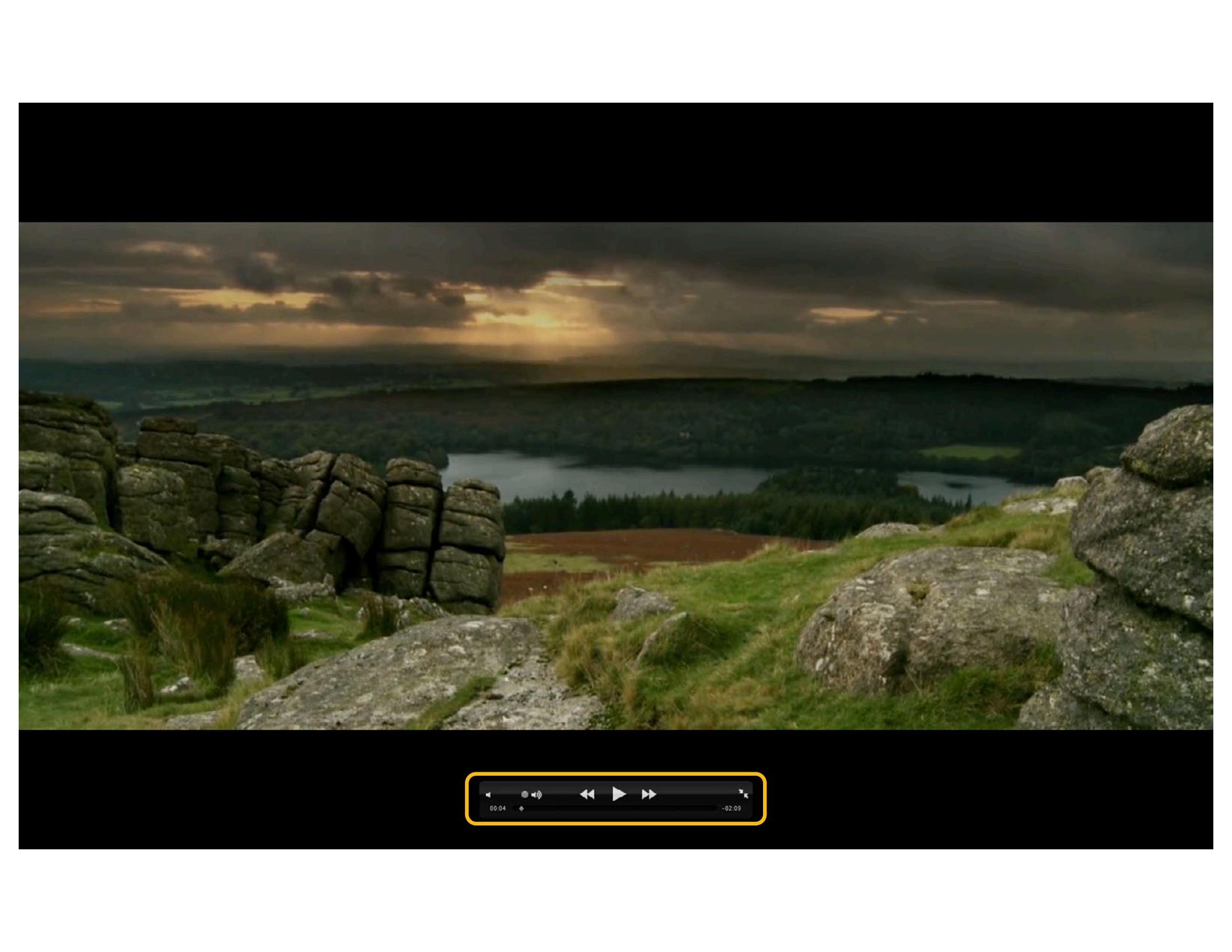
Augment your interface with audio



<http://www.james-watson.net/>







Implementing Fullscreen Interfaces

Learn how to...

Implementing Fullscreen Interfaces

Learn how to...

- Enter and exit fullscreen mode

Implementing Fullscreen Interfaces

Learn how to...

- Enter and exit fullscreen mode
- Define a custom appearance or behavior for the fullscreen element

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

```
document.getElementById("main").webkitRequestFullscreen();
```

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

```
document.getElementById("main").webkitRequestFullscreen();
```

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

```
document.getElementById("main").webkitRequestFullscreen();
```

<https://dvcs.w3.org/hg/fullscreen/raw-file/tip/Overview.html>

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

```
document.getElementById("main").webkitRequestFullscreen();
```

- Takes over the screen from which the request initiated

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

```
document.getElementById("main").webkitRequestFullscreen();
```

- Takes over the screen from which the request initiated
- Must be triggered by an explicit user action
 - A `mousemove` event does not qualify

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

```
document.getElementById("main").webkitRequestFullscreen();
```

- Takes over the screen from which the request initiated
- Must be triggered by an explicit user action
 - A `mousemove` event does not qualify
- Case matters!

Entering and Exiting Fullscreen Mode

Taking any webpage element fullscreen

- Fetch the element and request fullscreen for it

```
document.getElementById("main").webkitRequestFullscreen();  
                                                                    ↑
```

- Takes over the screen from which the request initiated
- Must be triggered by an explicit user action
 - A `mousemove` event does not qualify
- Case matters!

Entering and Exiting Fullscreen Mode

Going back to the page view

Entering and Exiting Fullscreen Mode

Going back to the page view

```
document.webkitExitFullscreen();
```

Entering and Exiting Fullscreen Mode

Going back to the page view

- Also requires an explicit user action

```
document.webkitExitFullscreen();
```

Entering and Exiting Fullscreen Mode

Going back to the page view

- Also requires an explicit user action
 - The initial click to enter fullscreen mode qualifies

```
document.webkitExitFullscreen();
```

Entering and Exiting Fullscreen Mode

Going back to the page view

- Also requires an explicit user action
 - The initial click to enter fullscreen mode qualifies

```
document.webkitExitFullscreen();
```

- Case matters!

Entering and Exiting Fullscreen Mode

Going back to the page view

- Also requires an explicit user action
 - The initial click to enter fullscreen mode qualifies

```
document.webkitExitFullscreen();  
                                     ↑
```

- Case matters!

Entering and Exiting Fullscreen Mode

Detecting fullscreen changes

- A single event listener for exiting or entering fullscreen mode

Entering and Exiting Fullscreen Mode

Detecting fullscreen changes

- A single event listener for exiting or entering fullscreen mode

`webkitfullscreenchange`

Entering and Exiting Fullscreen Mode

Detecting fullscreen changes

- A single event listener for exiting or entering fullscreen mode

```
var main = document.getElementById("main");
main.addEventListener('webkitfullscreenchange', addKeyListeners, false);
```

Entering and Exiting Fullscreen Mode

Detecting fullscreen changes

- A single event listener for exiting or entering fullscreen mode

```
var main = document.getElementById("main");
main.addEventListener('webkitfullscreenchange', addKeyListeners, false);
```

- Check for a fullscreen element to determine enter vs. exit

Entering and Exiting Fullscreen Mode

Detecting fullscreen changes

- A single event listener for exiting or entering fullscreen mode

```
var main = document.getElementById("main");
main.addEventListener('webkitfullscreenchange', addKeyListeners, false);
```

- Check for a fullscreen element to determine enter vs. exit

```
function addKeyListeners() {
  if (document.webkitFullscreenElement) {
    // entering fullscreen mode
    document.addEventListener('keydown', handleKeyPress, false);
  }
  else {
    // exiting fullscreen mode
    document.removeEventListener('keydown', handleKeyPress, false);
  }
}
```

Apple – iPhone 4S – The best of everything. Built right in.

www.apple.com/iphone/built-in-apps/

Reader

iPhone

Features Built-in Apps From the App Store iOS iCloud Tech Specs Buy iPhone

[Browse All Built-in Apps](#)

It does everything. Better.

Shoot 8-megapixel photos and 1080p HD video. See your favorite people with FaceTime. Browse the web. Send messages, messages, and more messages. Check your email and calendar. Listen to music. Watch movies. Oh, and make phone calls. It's the best of everything. Built right in.



 It just might be the best camera ever on a phone.

You can expect a few things from the 8MP iSight camera with all-new optics. Like better shots in low light and better shots in lots of light. More color accuracy, reduced motion blur, better exposure, and a lot more focus on faces with face detection. Expect to take it everywhere, pull it out anywhere, and shoot just about everything you see. And expect beautiful, braggable, framable photos.

[Learn more about Camera >](#)



 Video recording.
Film your life in 1080p HD.

Now you can shoot and edit 1080p HD video. From your iPhone. Film your favorite people, your big adventures, and the small stuff you see every day. And it will all be even clearer, bigger, brighter, and more entertaining than you remember.

[Learn more about HD video recording >](#)

Apple – iPhone 4S – The best of everything. Built right in.

www.apple.com/iphone/built-in-apps/

Reader

iPhone

Features Built-in Apps From the App Store iOS iCloud Tech Specs Buy iPhone

[Browse All Built-in Apps](#)

It does everything. Better.

Shoot 8-megapixel photos and 1080p HD video. See your favorite people with FaceTime. Browse the web. Send messages, messages, and more messages. Check your email and calendar. Listen to music. Watch movies. Oh, and make phone calls. It's the best of everything. Built right in.



It just might be the best camera ever on a phone.

You can expect a few things from the 8MP iSight camera with all-new optics. Like better shots in low light and better shots in lots of light. More color accuracy, reduced motion blur, better exposure, and a lot more focus on faces with face detection. Expect to take it everywhere, pull it out anywhere, and shoot just about everything you see. And expect beautiful, braggable, framable photos.

[Learn more about Camera >](#)

Video recording.
Film your life in 1080p HD.

Now you can shoot and edit 1080p HD video. From your iPhone. Film your favorite people, your big adventures, and the small stuff you see every day. And it will all be even clearer, bigger, brighter, and more entertaining than you remember.

[Learn more about HD video recording >](#)



Apple – iPhone 4S – The best of everything. Built right in.

www.apple.com/iphone/built-in-apps/

Reader

iPhone

Features Built-in Apps From the App Store iOS iCloud Tech Specs Buy iPhone

[Browse All Built-in Apps](#)

It does everything. Better.

Shoot 8-megapixel photos and 1080p HD video. See your favorite people with FaceTime. Browse the web. Send messages, messages, and more messages. Check your email and calendar. Listen to music. Watch movies. Oh, and make phone calls. It's the best of everything. Built right in.



It just might be the best camera ever on a phone.

You can expect a few things from the 8MP iSight camera with all-new optics. Like better shots in low light and better shots in lots of light. More color accuracy, reduced motion blur, better exposure, and a lot more focus on faces with face detection. Expect to take it everywhere, pull it out anywhere, and shoot just about everything you see. And expect beautiful, braggable, framable photos.

[Learn more about Camera >](#)

Video recording.
Film your life in 1080p HD.

Now you can shoot and edit 1080p HD video. From your iPhone. Film your favorite people, your big adventures, and the small stuff you see every day. And it will all be even clearer, bigger, brighter, and more entertaining than you remember.

[Learn more about HD video recording >](#)



Squeee News, Video and Gossip

Reader SIGN IN

LATEST STORIES WEDNESDAY, JUN 6, 2012

0:14 / 1:15

SQUEE

Teeny Tiny Pig Overcomes His Fear of Stairs to Get to a Bowl of Delicious Oatmeal

BY CASSIE MURDOCH JUN 5, 2012 11:15 PM

Share 36k 124,617 134

Presented by bing

GET OUR TOP STORIES

Like 147,177 people like this.

Teeny Tiny Pig JUN 5, 2012 Overcomes His Fear of Stairs to Get to a Bowl of Delicious Oatmeal

Happy Monday: Here JUN 4, 2012 Are Some Dogs On Skateboards

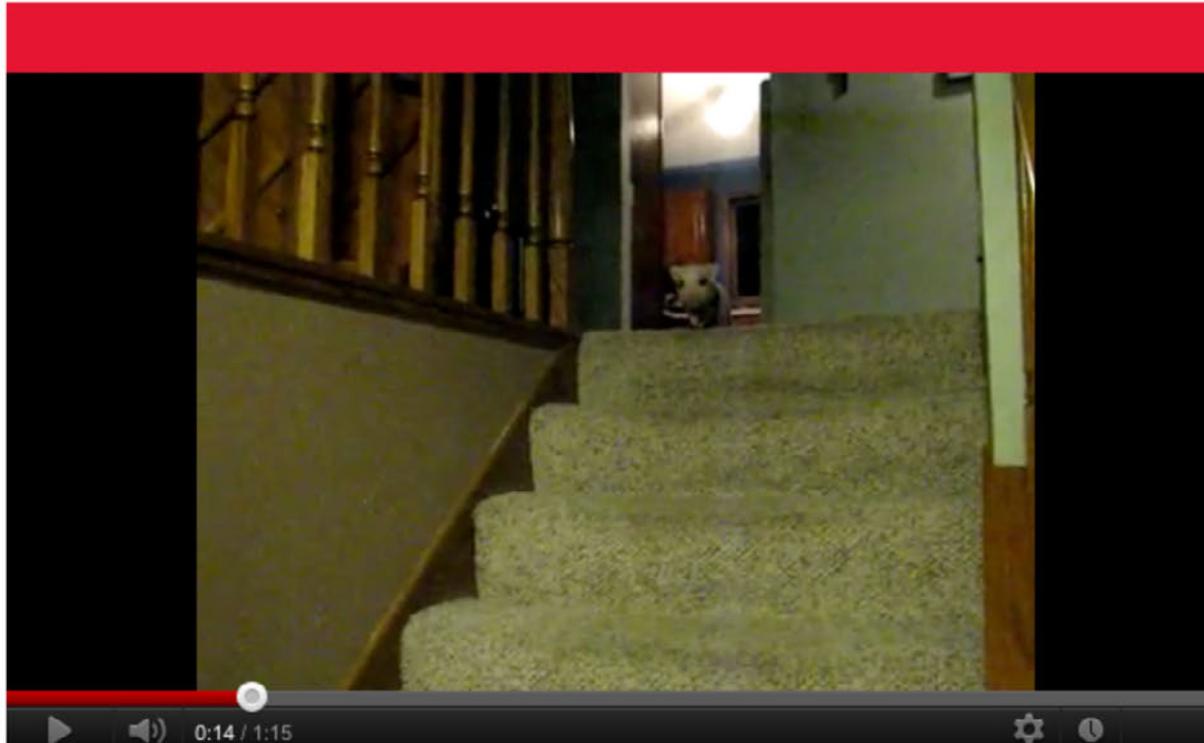
Let These Dog and Pig JUN 3, 2012 Friends Enjoy a Playful Romp All Over Your Heart

Dog Seems a Little JUN 1, 2012 Unnerved by Former Street Cat

Hamster Can Play MAY 30, 2012 Dead Among the Best of Them

Scarf-Wearing Pig MAY 30, 2012 Spotted On Pittsburgh Highway

Squeee News, Video and Gossip



0:14 / 1:15

SQEEE

Teeny Tiny Pig Overcomes His Fear of Stairs to Get to a Bowl of

BY CASSIE MURDOCH JUN 5, 2012 11:15 PM

Share +1 Like 36k 124,617 134

Presented by bing

LATEST STORIES WEDN

Teeny Tiny Pig Overcomes His Fear of Stairs to Get to a Bowl of Delic

Happy Monday: I Are Some Dogs O

Let These Dog an Friends Enjoy a P Over Your Heart

Dog Seems a Littl

Squeee News, Video and Gossip

LATEST STORIES ▾ WEDN

Fullscreen

Teeny Tiny Pig Overcomes His F to a Bowl of Delic

Happy Monday: I Are Some Dogs O

Let These Dog an Friends Enjoy a P Over Your Heart

Dog Seems a Littl

The image shows a screenshot of a web browser window. At the top, there's a dark header bar with the text "Squeee News, Video and Gossip". Below this is a toolbar with standard icons for back, forward, and search. The main content area features a video player with a red progress bar at 0:14 / 1:15. To the right of the video, there's a large yellow arrow pointing down towards a "Fullscreen" button. Below the video, there's a title "SQUEE" and a story summary: "Teeny Tiny Pig Overcomes His Fear of Stairs to Get to a Bowl of". To the right of the story, there's a timestamp "JUN 5, 2012 11:15 PM" and social sharing options: "Share" (with a +1 button), "Like" (with 36k likes), and "124,617 comments". A note says "Presented by bing". On the far right, there are several thumbnail images for other news stories, each with a play button icon.

SQUEE

Teeny Tiny Pig Overcomes His Fear of Stairs to Get to a Bowl of

BY CASSIE MURDOCH JUN 5, 2012 11:15 PM

Share +1 Like 36k 124,617 comments

Presented by bing

Entering and Exiting Fullscreen Mode

You are in control

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

Entering and Exiting Fullscreen Mode

You are in control

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

```
<iframe webkitAllowFullscreen src="http://a-trusted-video-site.com">
```

Entering and Exiting Fullscreen Mode

You are in control

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

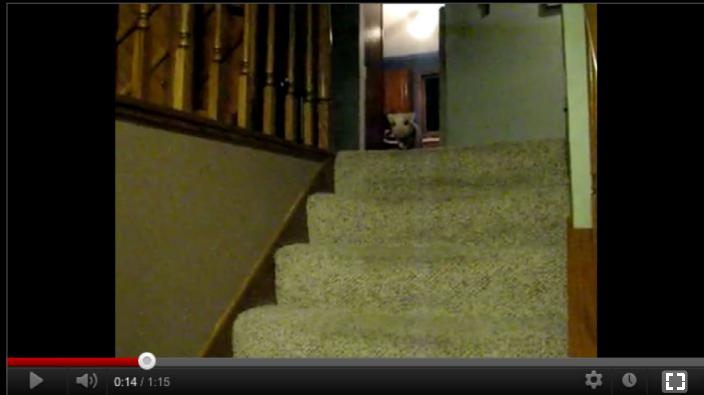
```
<iframe webkitAllowFullscreen src="http://a-trusted-video-site.com">
<iframe src="http://lose-20-pounds-in-a-day.com">
```

Entering and Exiting Fullscreen Mode

Inline content is aware of your decision

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

```
<iframe webkitAllowFullscreen src="http://a-trusted-video-site.com">
```



```
document.webkitFullscreenEnabled
```

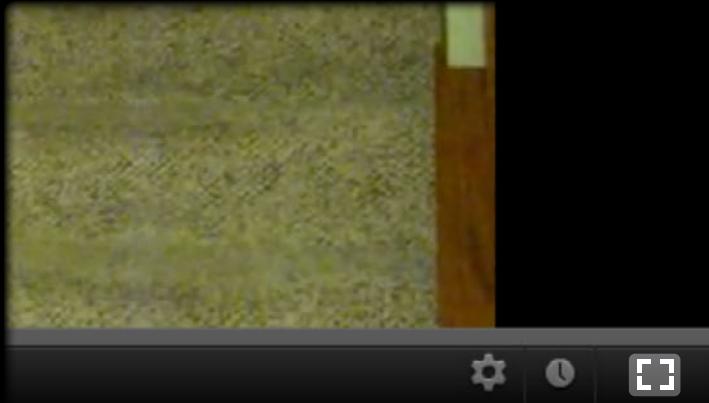
```
</iframe>
```

Entering and Exiting Fullscreen Mode

Inline content is aware of your decision

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

```
<iframe webkitAllowFullscreen src="http://a-trusted-video-site.com">
```



```
document.webkitFullscreenEnabled
```

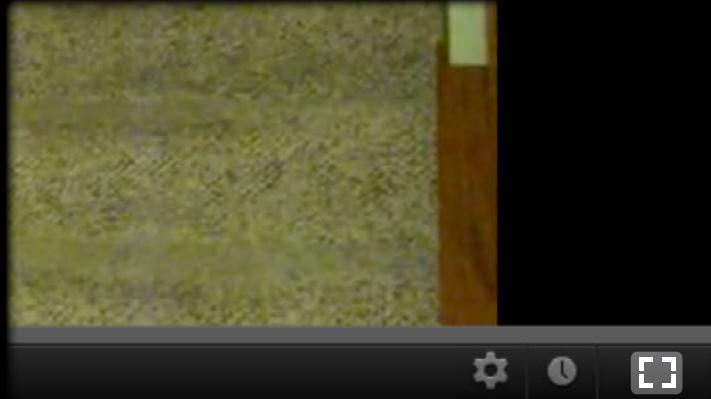
```
</iframe>
```

Entering and Exiting Fullscreen Mode

Inline content is aware of your decision

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

```
<iframe webkitAllowFullscreen src="http://a-trusted-video-site.com">
```



`document.webkitFullscreenEnabled`

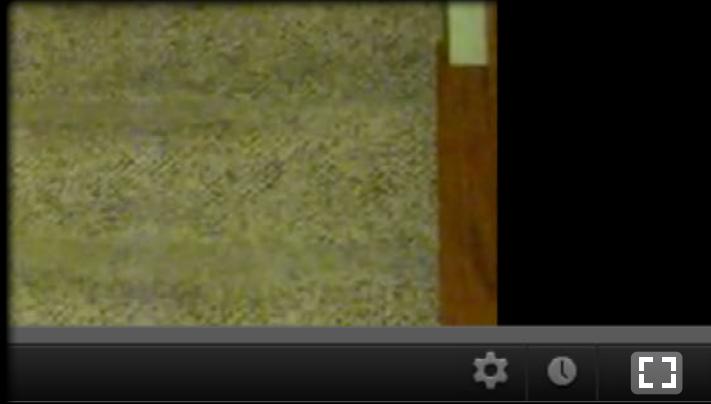
```
</iframe>
```

Entering and Exiting Fullscreen Mode

Inline content is aware of your decision

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

```
<iframe webkitAllowFullscreen src="http://a-trusted-video-site.com">
```



```
if (!document.webkitFullscreenEnabled)  
// hide this button
```

```
</iframe>
```

Entering and Exiting Fullscreen Mode

Inline content is aware of your decision

- iframes cannot take themselves fullscreen by default
- Add the `webkitAllowFullscreen` attribute to allow fullscreen for iframes

```
<iframe webkitAllowFullscreen src="http://a-trusted-video-site.com">
```



```
if (!document.webkitFullscreenEnabled)  
// hide this button
```

```
</iframe>
```

Implementing Fullscreen Interfaces

Learn how to...

- Enter and exit fullscreen mode
- Define a custom appearance or behavior for the fullscreen element

Implementing Fullscreen Interfaces

Learn how to...

- Enter and exit fullscreen mode
- Define a custom appearance or behavior for the fullscreen element

⊗



CSS Filters

Applying filter effects to any element

`-webkit-filter:`



CSS Filters

Applying filter effects to any element

```
-webkit-filter: saturate(0-n);
```



CSS Filters

Applying filter effects to any element



```
-webkit-filter: saturate(0-n);  
              sepia(0-1);  
              invert(0-1);  
              contrast(0-n);  
              grayscale(0-1);  
              opacity(0-1);  
              hue-rotate(0-360deg);  
              blur(0-npx);  
              brightness(0-n);  
              drop-shadow(xOffsetpx yOffsetpx blurpx color);
```

CSS Filters

Applying filter effects to any element



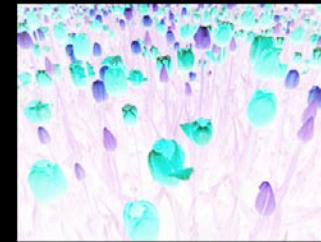
original image



grayscale(1)



sepia(1)



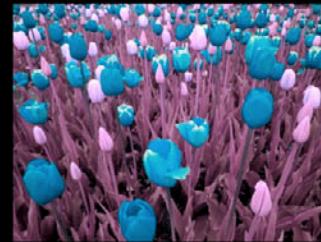
invert(1)



hue-rotate(45deg)



hue-rotate(90deg)



hue-rotate(210deg)



hue-rotate(320deg)



blur(3px)



brightness(0.2)



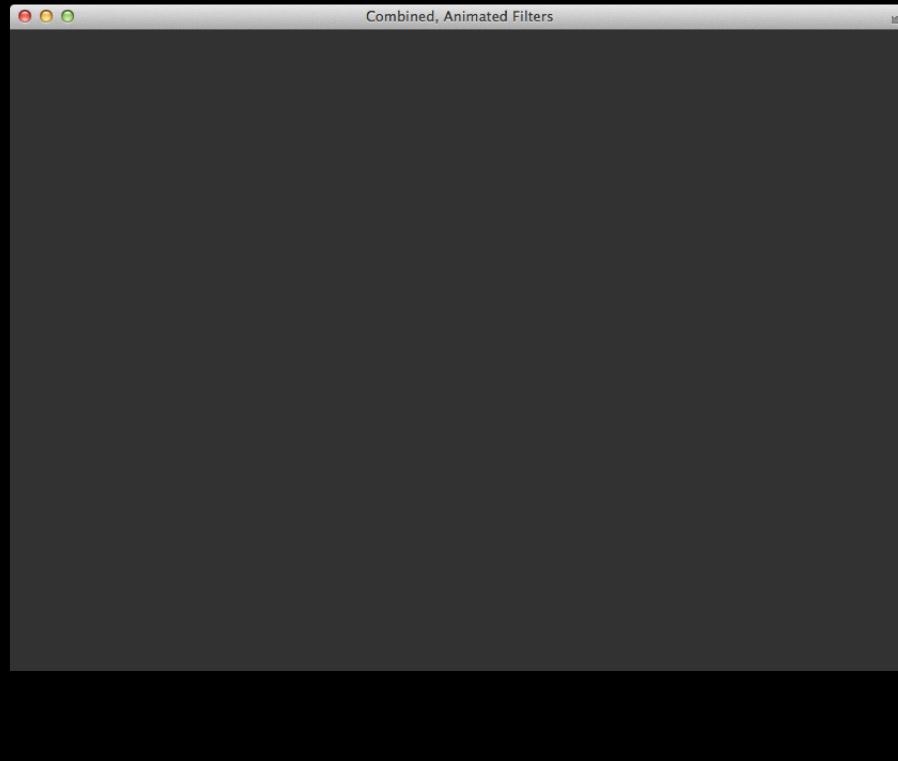
contrast(2)



saturate(3)

CSS Filters

Combining and animating filter functions

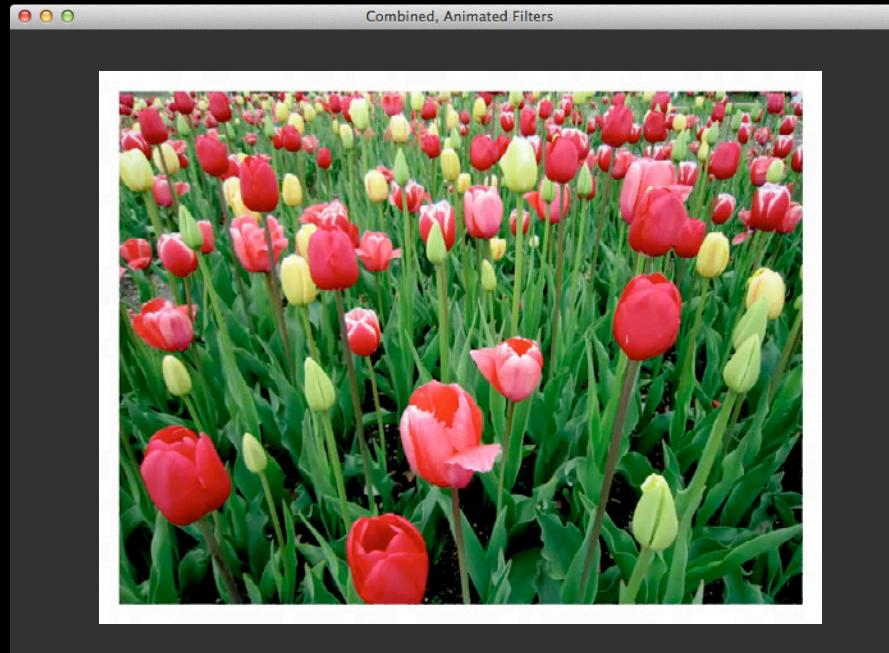


CSS Filters

Combining and animating filter functions



```
-webkit-filter: hue-rotate(320deg) contrast(2);
```



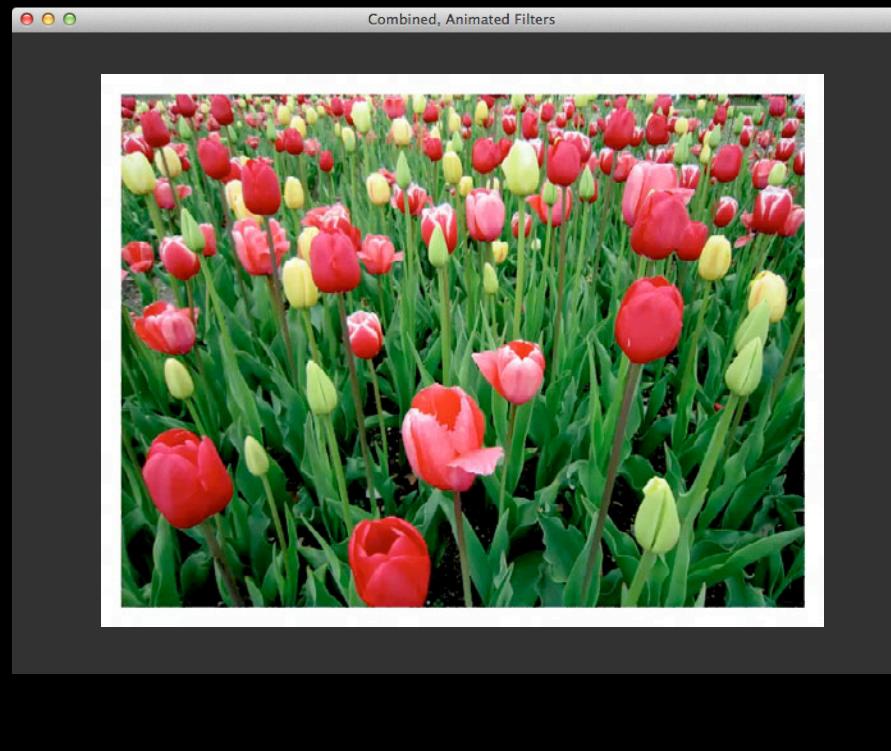
CSS Filters

Combining and animating filter functions



```
-webkit-filter: hue-rotate(320deg) contrast(2);
```

```
-webkit-transition: -webkit-filter 5s linear;
```



CSS Filters

Combining and animating filter functions



```
-webkit-filter: hue-rotate(320deg) contrast(2);
```

```
-webkit-transition: -webkit-filter 5s linear;
```



CSS Filters

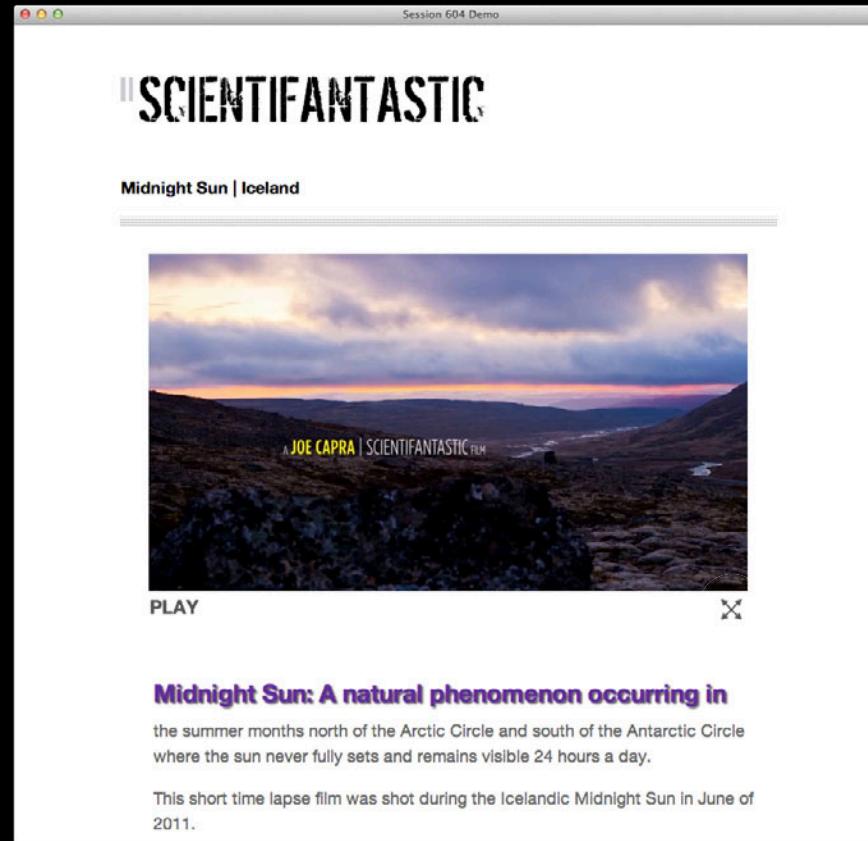
Combining and animating filter functions

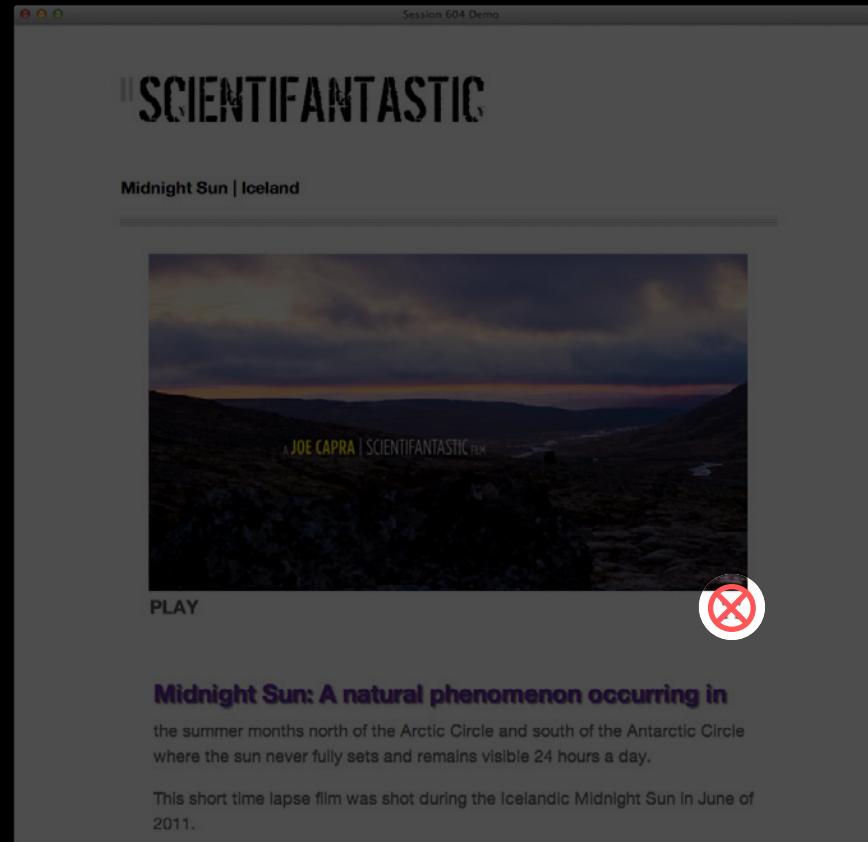


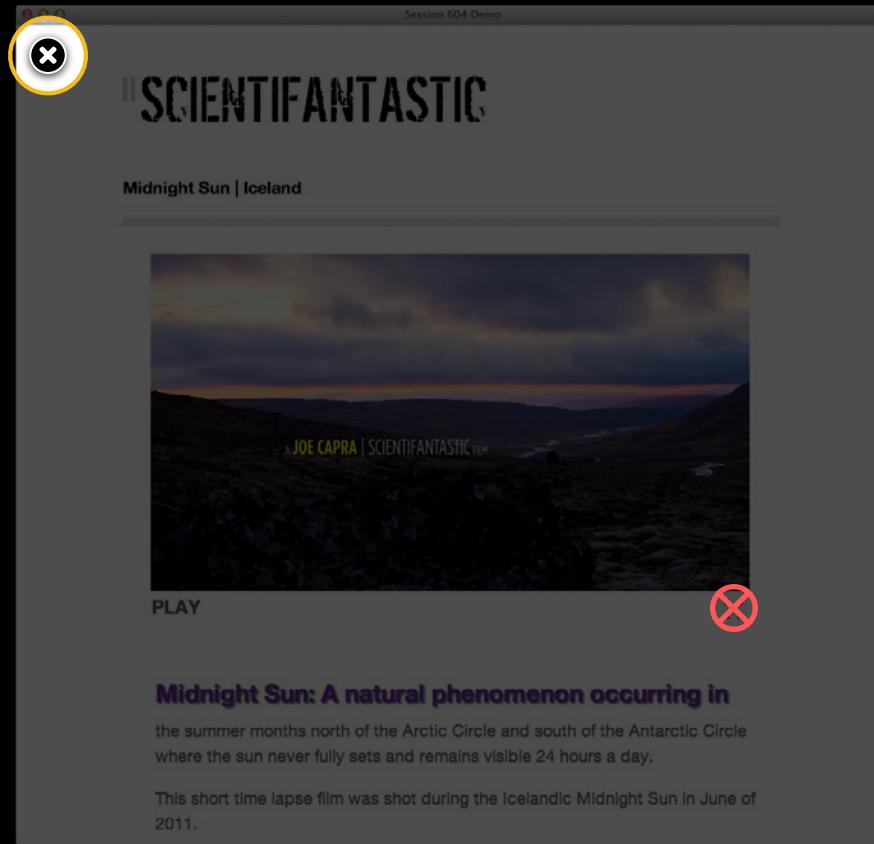
```
-webkit-filter: hue-rotate(320deg) contrast(2);
```

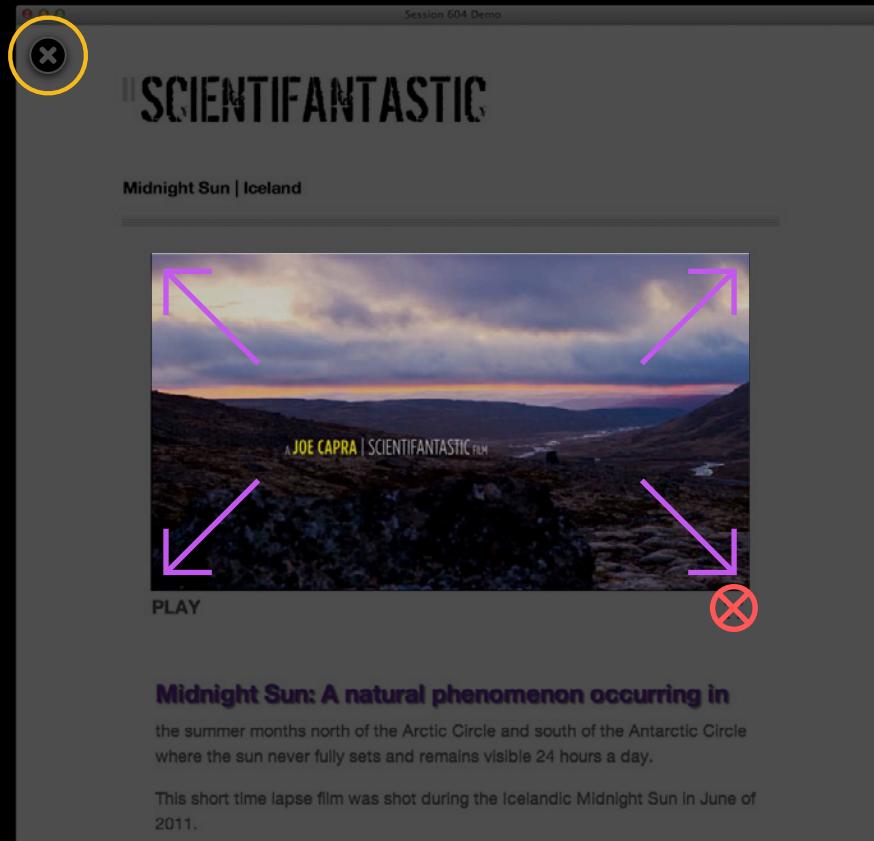
```
-webkit-transition: -webkit-filter 5s linear;
```

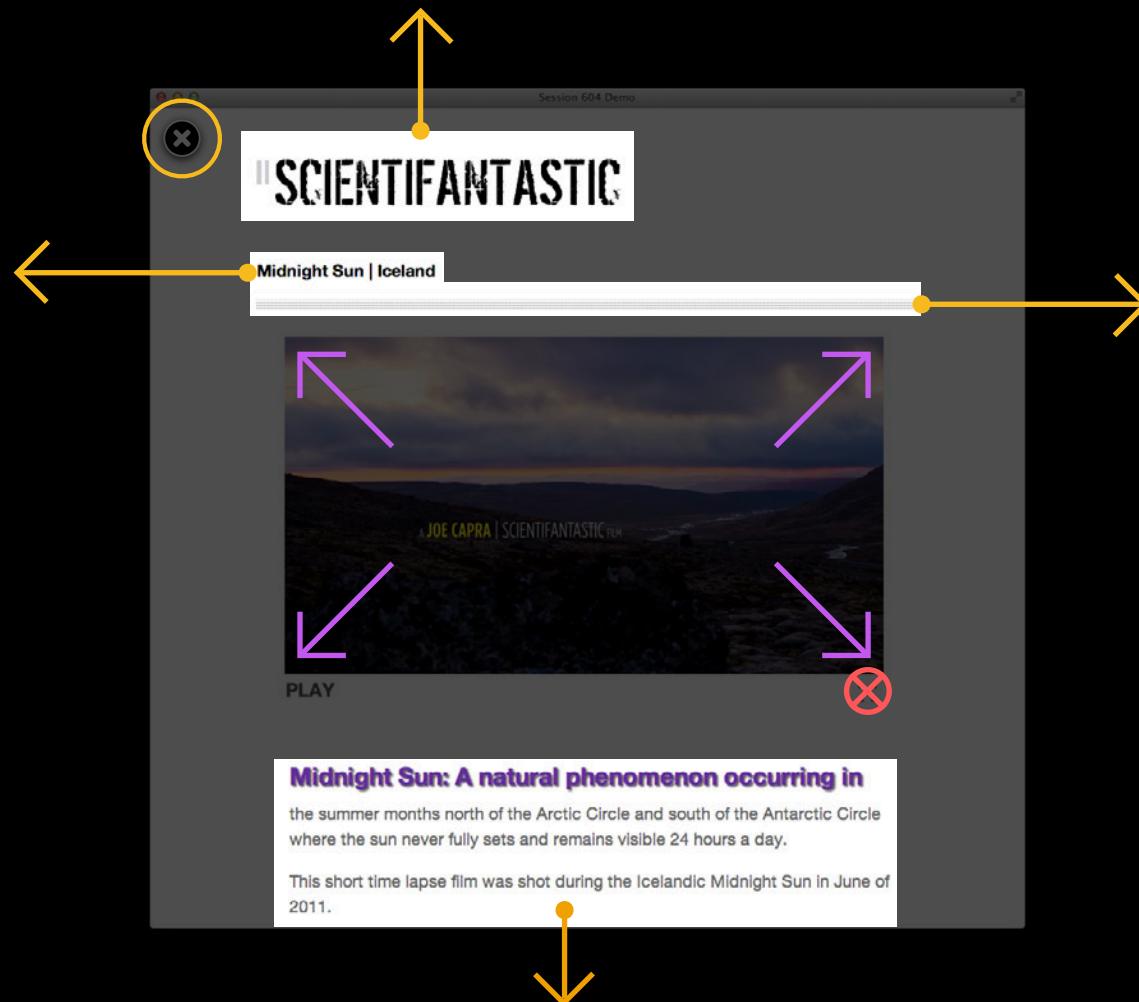


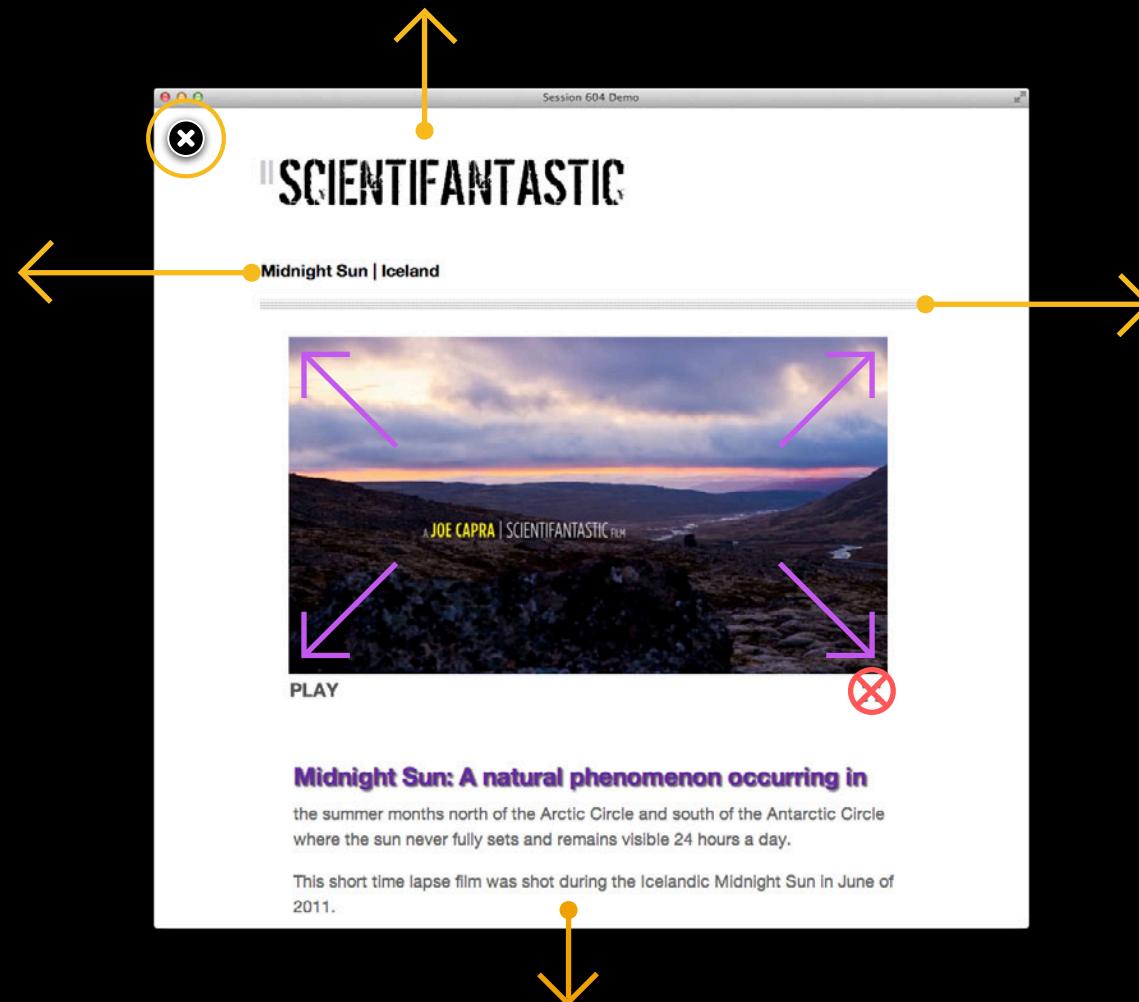












Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
- CSS pseudo-classes are applied by the browser
- You define the appearance

Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
`:--webkit-fullscreen`
- CSS pseudo-classes are applied by the browser
- You define the appearance

Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
`:--webkit-fullscreen`
- CSS pseudo-classes are applied by the browser

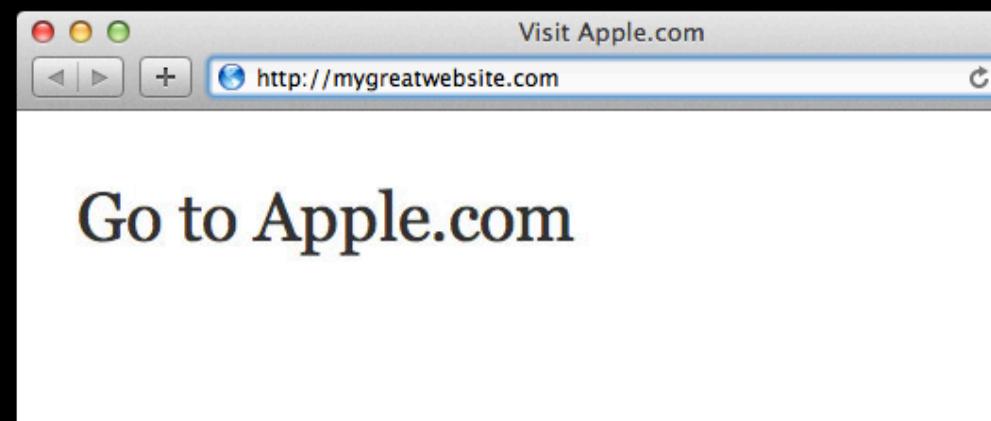
Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
`:--webkit-fullscreen`
- CSS pseudo-classes are applied by the browser
- You define the appearance

Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
 `:webkit-fullscreen`
- CSS pseudo-classes are applied by the browser
- You define the appearance

```
a {  
  color: gray;  
}
```

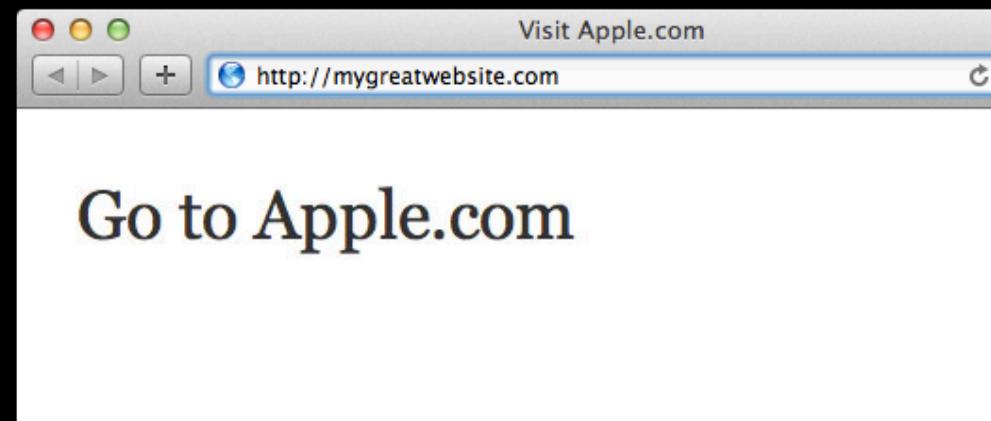


Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
 `:webkit-fullscreen`
- CSS pseudo-classes are applied by the browser
- You define the appearance

```
a {  
  color: gray;  
}
```

```
a:hover {  
  color: white;  
  background-color: darkcyan;  
}
```

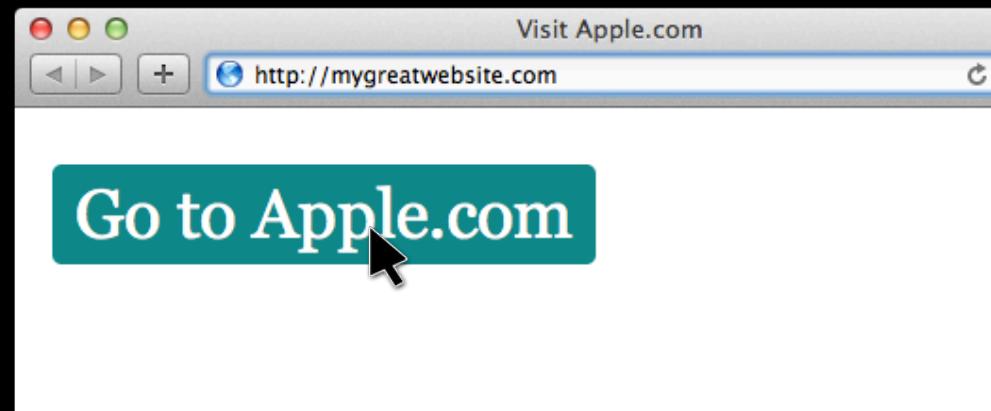


Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
 `:webkit-fullscreen`
- CSS pseudo-classes are applied by the browser
- You define the appearance

```
a {  
  color: gray;  
}
```

```
a:hover {  
  color: white;  
  background-color: darkcyan;  
}
```

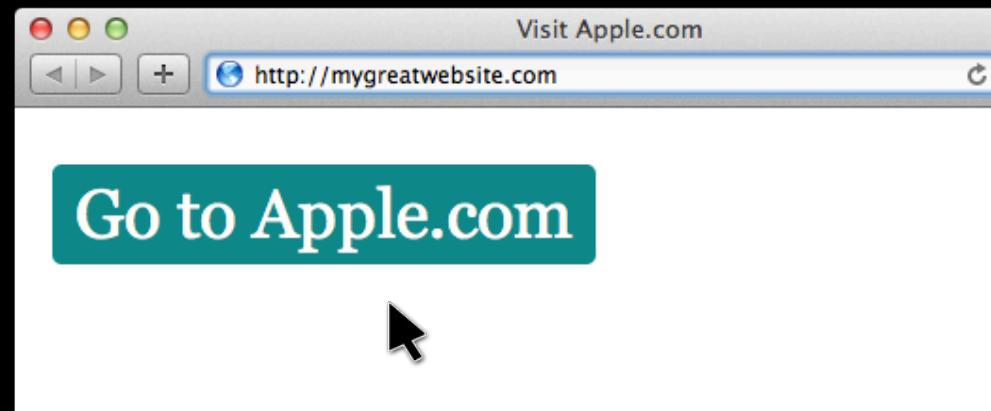


Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode
 `:webkit-fullscreen`
- CSS pseudo-classes are applied by the browser
- You define the appearance

```
a {  
  color: gray;  
}
```

```
a:hover {  
  color: white;  
  background-color: darkcyan;  
}
```



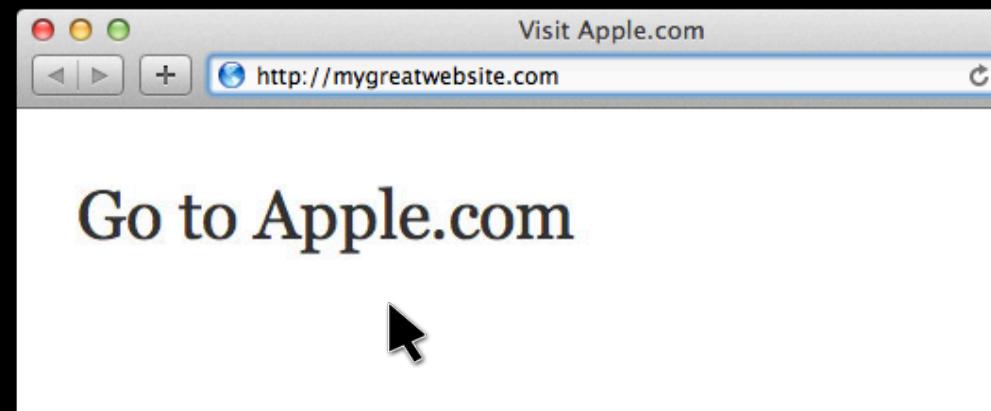
Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode

```
:--webkit-fullscreen
```

- CSS pseudo-classes are applied by the browser
- You define the appearance

```
a {  
    color: gray;  
    -webkit-transition: background-color 2s, color 2s;  
}  
  
a:hover {  
    color: white;  
    background-color: darkcyan;  
}
```



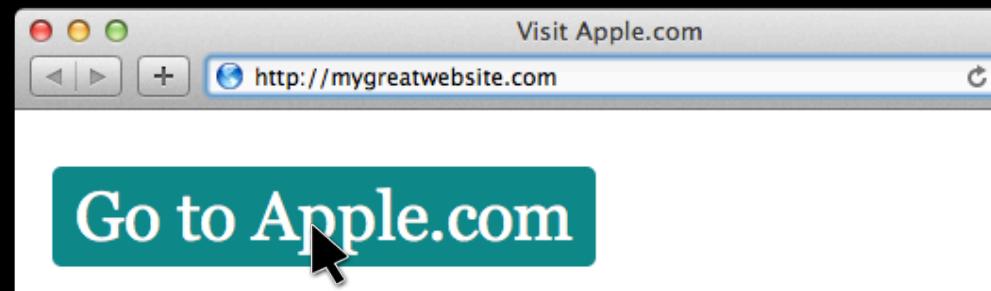
Modifying Element Appearance in Fullscreen

- A CSS pseudo-class for fullscreen mode

```
:--webkit-fullscreen
```

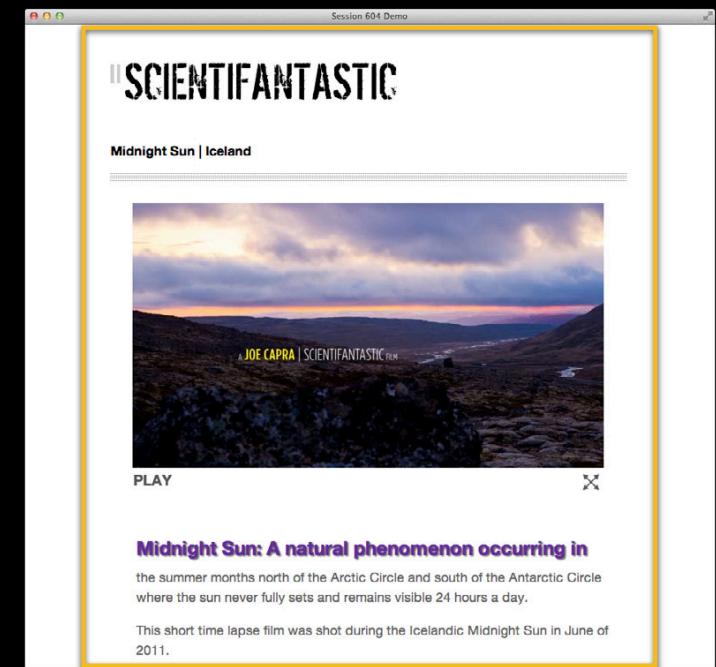
- CSS pseudo-classes are applied by the browser
- You define the appearance

```
a {  
    color: gray;  
    -webkit-transition: background-color 2s, color 2s;  
}  
  
a:hover {  
    color: white;  
    background-color: darkcyan;  
}
```



Modifying Element Appearance in Fullscreen Using the fullscreen pseudo-class

- Changing the background color

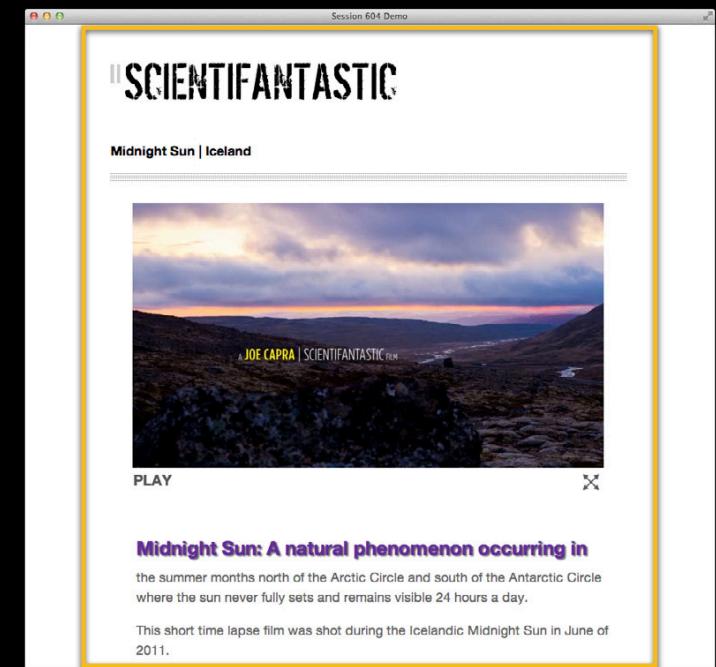


Modifying Element Appearance in Fullscreen Using the fullscreen pseudo-class

- Changing the background color

```
#main {  
  background-color: white;  
}
```

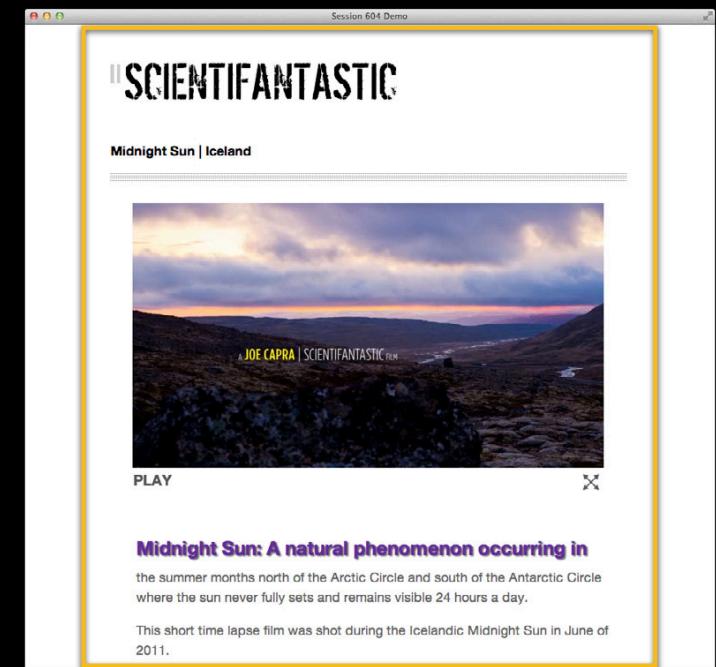
```
// main is the element going fullscreen  
#main:-webkit-fullscreen {  
  background-color: black;  
}
```



Modifying Element Appearance in Fullscreen Using the fullscreen pseudo-class

- Changing the background color

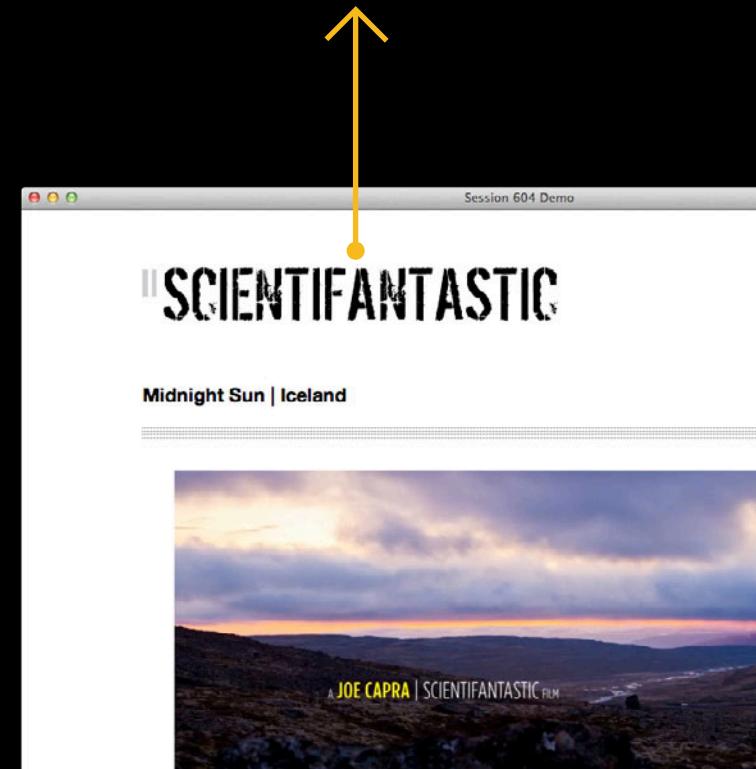
```
#main {  
  background-color: white;  
  -webkit-transition: background-color 2s;  
}  
  
// main is the element going fullscreen  
#main:-webkit-fullscreen {  
  background-color: black;  
}
```



Modifying Element Appearance in Fullscreen Using the fullscreen pseudo-class

- "Sliding" elements off the page

```
<div id="logo"></div>
```



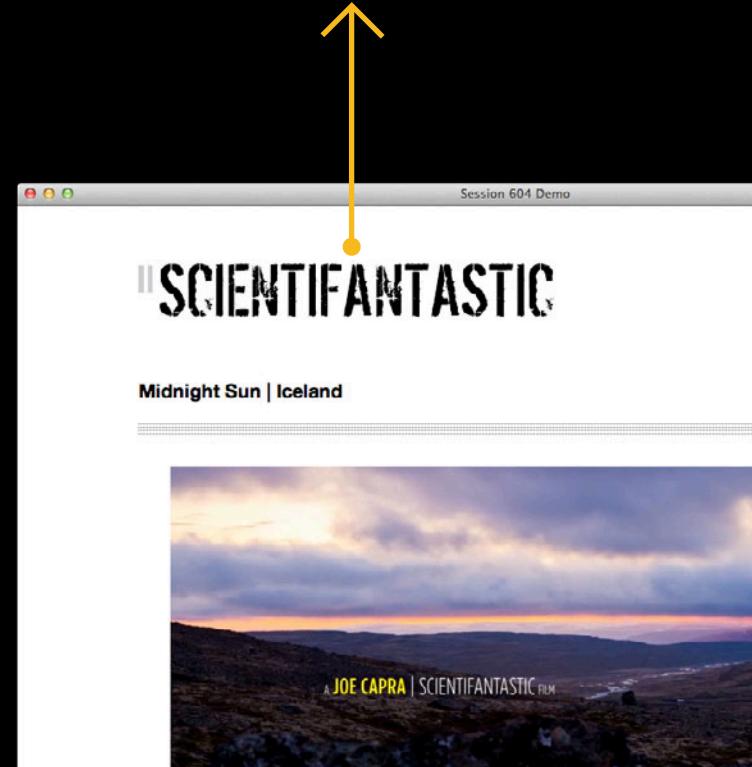
Modifying Element Appearance in Fullscreen Using the fullscreen pseudo-class

- "Sliding" elements off the page

```
<div id="logo"></div>

#logo {
  background-image:url("logo.png");
}

// #logo inside of #main,
// the element going fullscreen
:-webkit-fullscreen #logo {
  -webkit-transform: translateY(-300px);
}
```



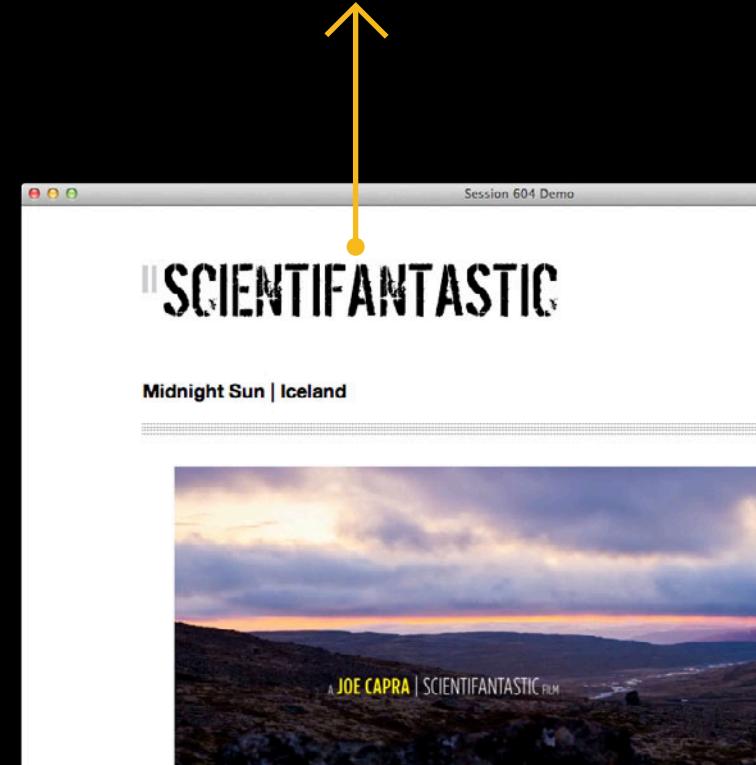
Modifying Element Appearance in Fullscreen Using the fullscreen pseudo-class

- "Sliding" elements off the page

```
<div id="logo"></div>

#logo {
    background-image:url("logo.png");
    -webkit-transition: -webkit-transform 2s;
}

// #logo inside of #main,
// the element going fullscreen
:-webkit-fullscreen #logo {
    -webkit-transform: translateY(-300px);
}
```



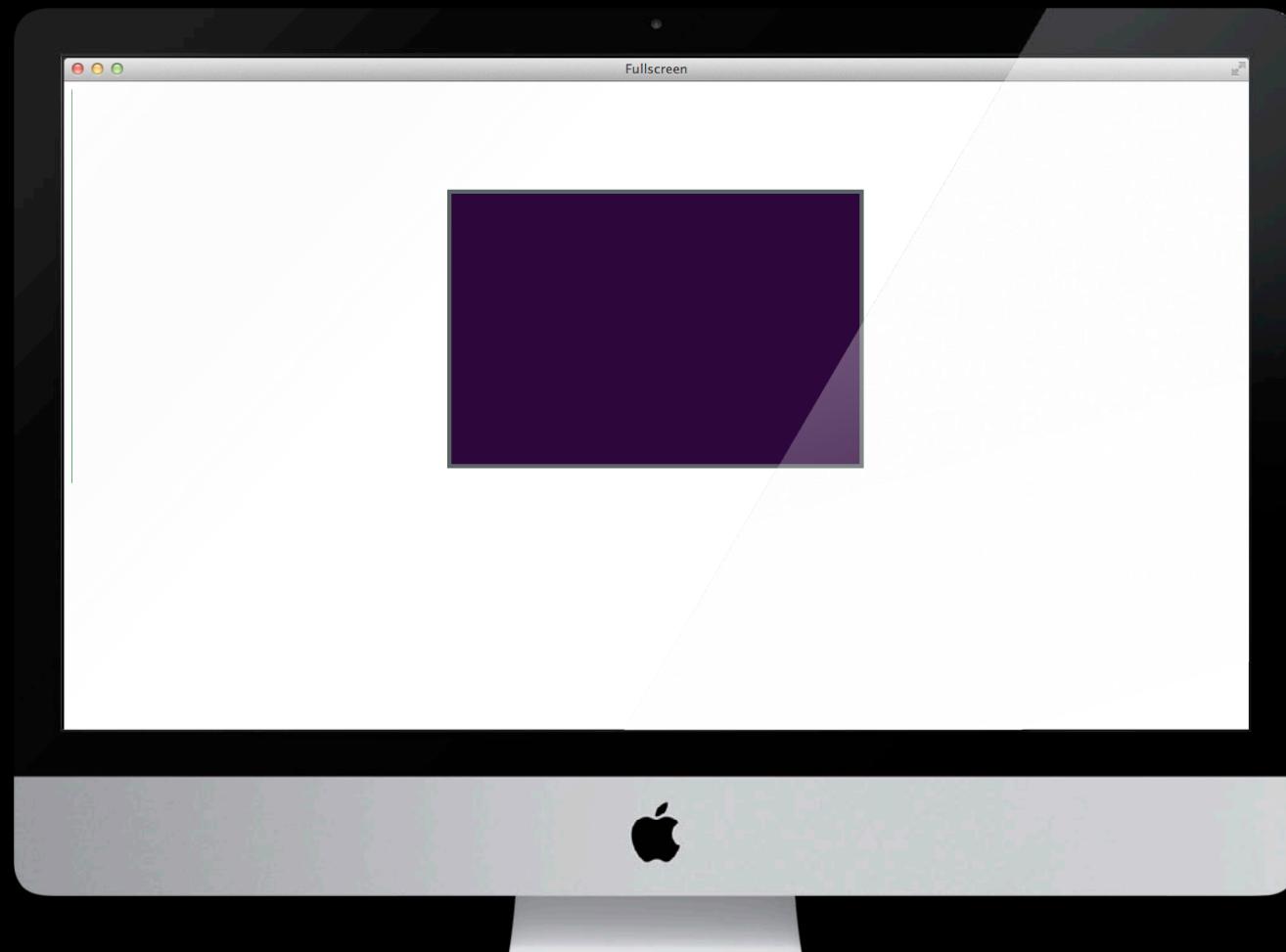
Modifying Element Appearance in Fullscreen

Resizing your element to fill the screen



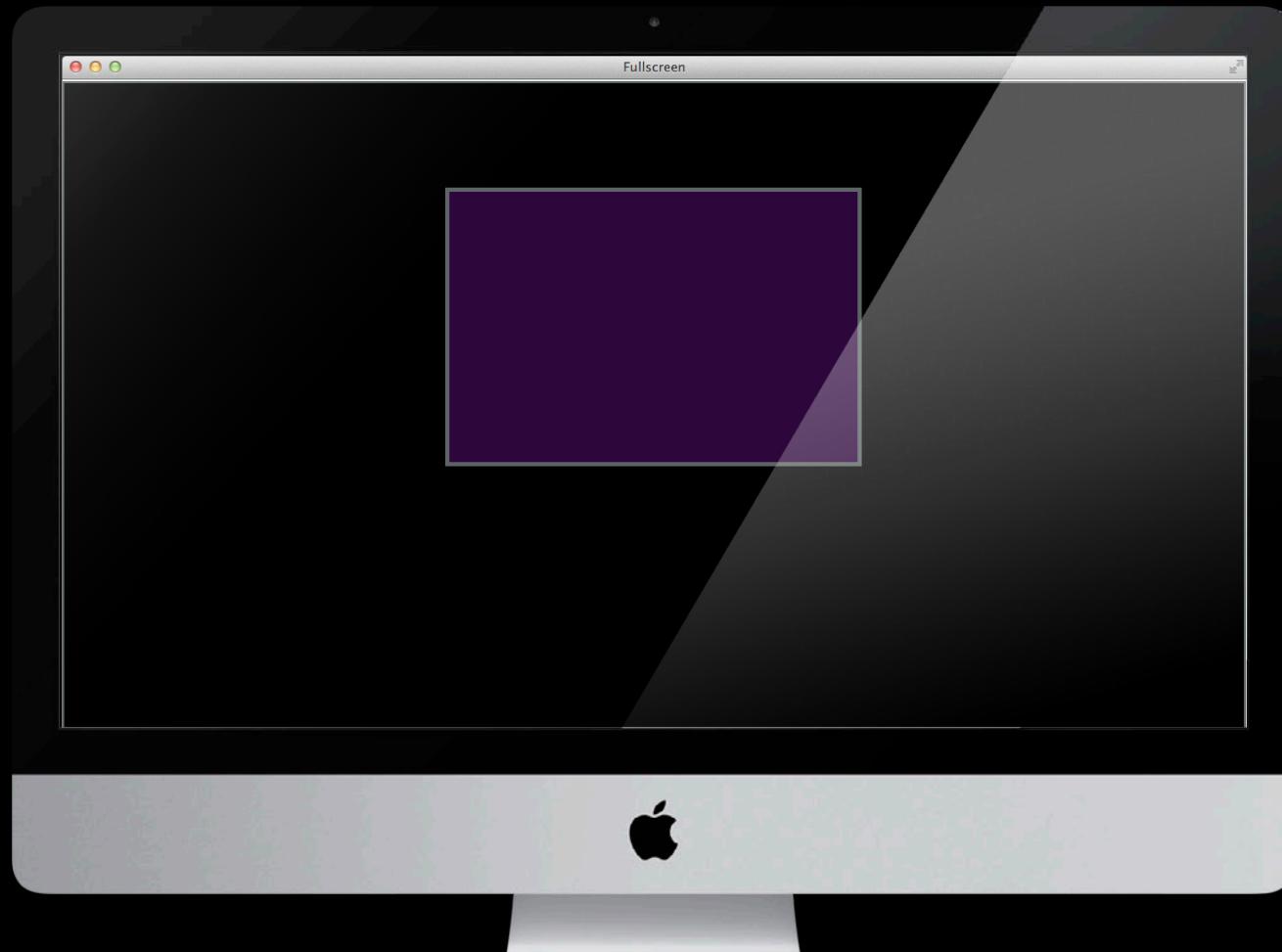
Modifying Element Appearance in Fullscreen

Resizing your element to fill the screen



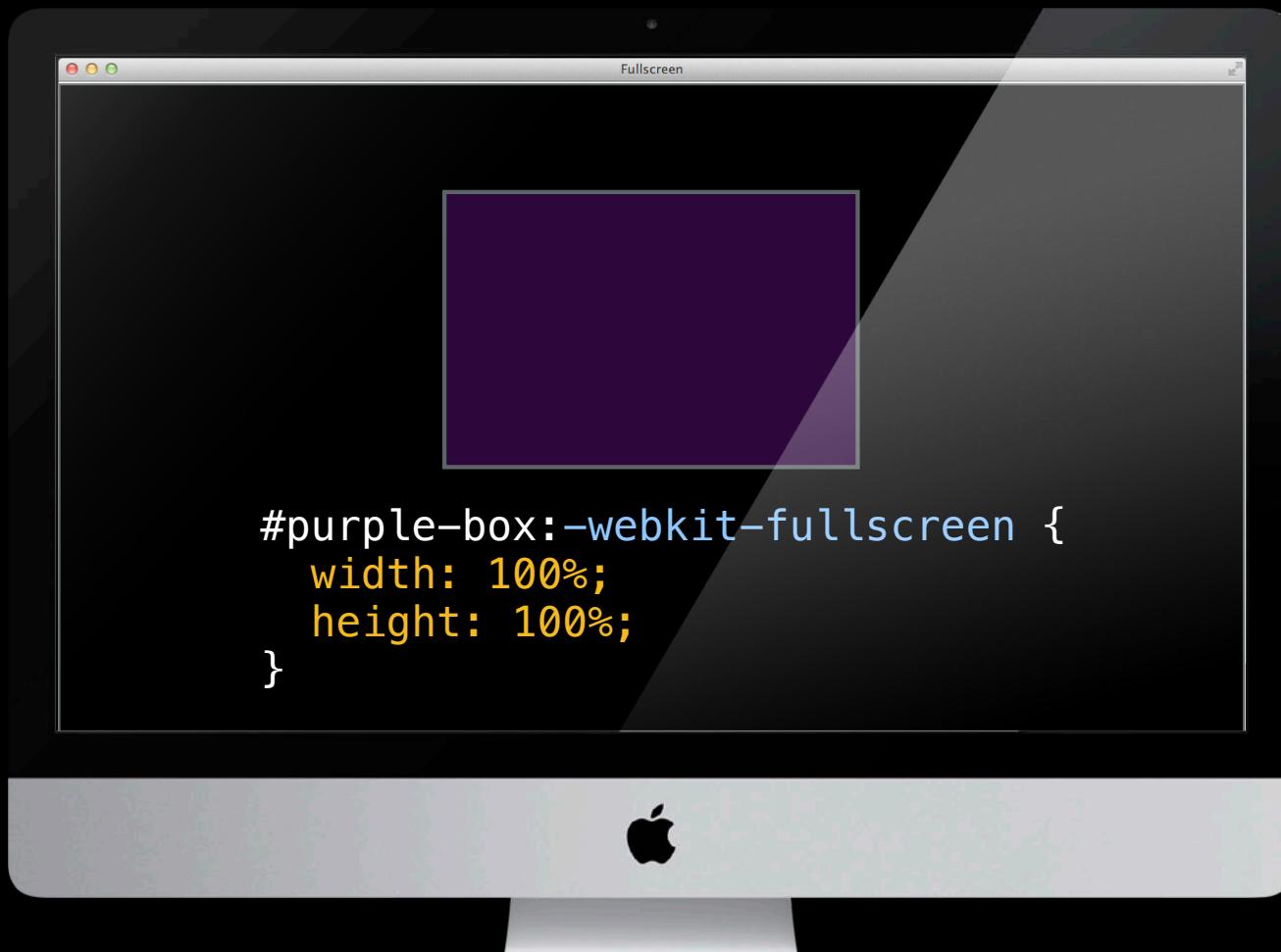
Modifying Element Appearance in Fullscreen

Resizing your element to fill the screen



Modifying Element Appearance in Fullscreen

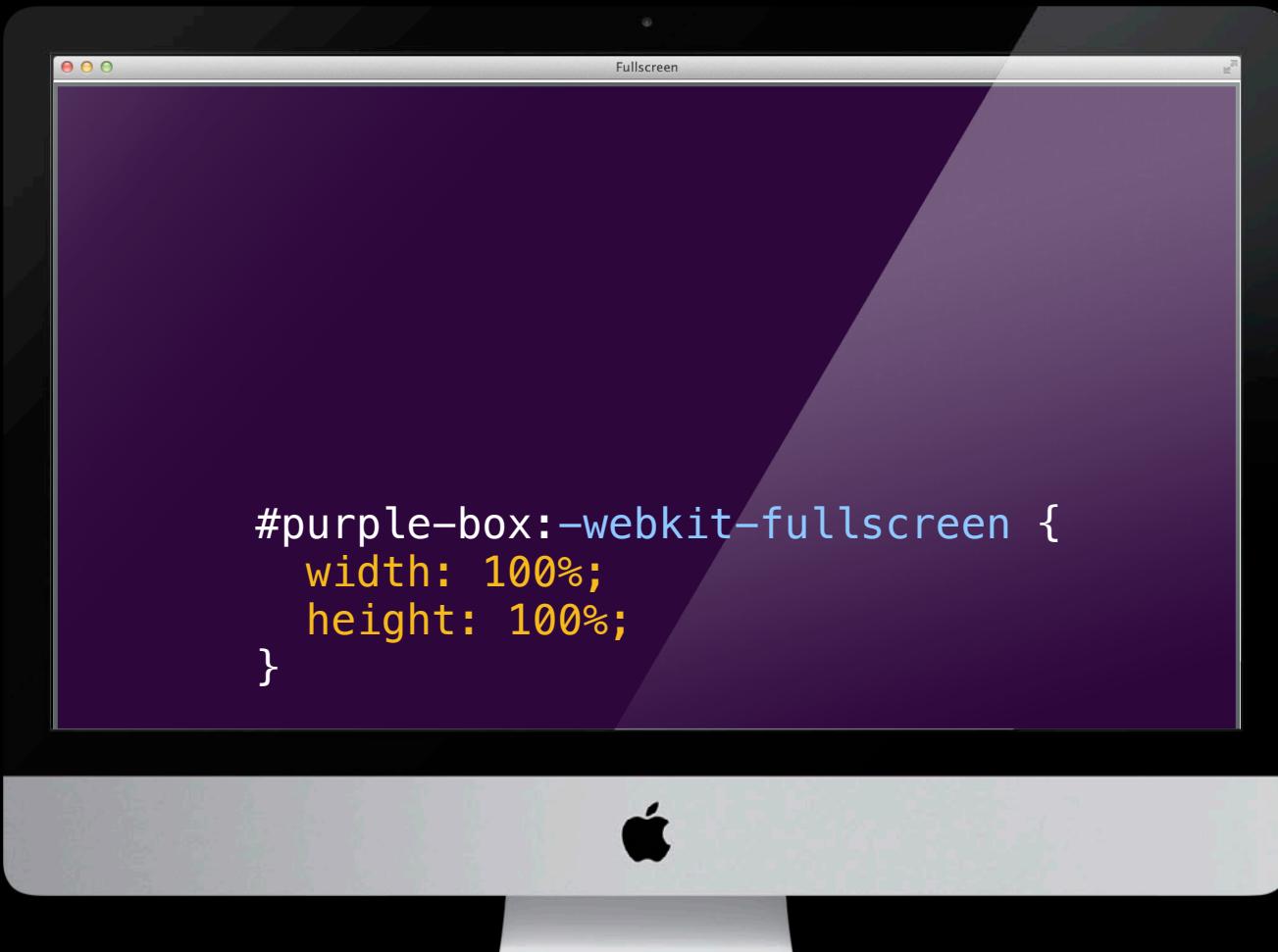
Resizing your element to fill the screen



```
#purple-box:-webkit-fullscreen {  
    width: 100%;  
    height: 100%;  
}
```

Modifying Element Appearance in Fullscreen

Resizing your element to fill the screen



```
#purple-box:-webkit-fullscreen {  
    width: 100%;  
    height: 100%;  
}
```

Implementing Fullscreen Interfaces

You have complete control

Implementing Fullscreen Interfaces

You have complete control

- Easy to trigger and detect fullscreen mode

Implementing Fullscreen Interfaces

You have complete control

- Easy to trigger and detect fullscreen mode
- You decide how iframes behave

Implementing Fullscreen Interfaces

You have complete control

- Easy to trigger and detect fullscreen mode
- You decide how iframes behave
- Custom appearance and behavior
 - `:webkit-fullscreen` pseudo-class
 - `webkitfullscreenchange`
 - `document.webkitFullscreenElement`

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

Augment your interface with audio

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

Augment your interface with audio

Using the <audio> tag

It's easy to...



Using the <audio> tag

It's easy to...

- Stream audio



Using the <audio> tag

It's easy to...

- Stream audio
- Add a background audio track



Using the <audio> tag

It's easy to...

- Stream audio
- Add a background audio track
- Play audio without plug-ins



Using the <audio> Tag

It's much harder to...



Using the <audio> Tag

It's much harder to...

- Play a sound at an exact time



Using the <audio> Tag

It's much harder to...

- Play a sound at an exact time
- Dynamically apply effects, like reverb and echo



Using the <audio> Tag

It's much harder to...

- Play a sound at an exact time
- Dynamically apply effects, like reverb and echo
- Generate sounds



Using the <audio> Tag

It's much harder to...

- Play a sound at an exact time
- Dynamically apply effects, like reverb and echo
- Generate sounds
- Access raw audio data





Web Audio API

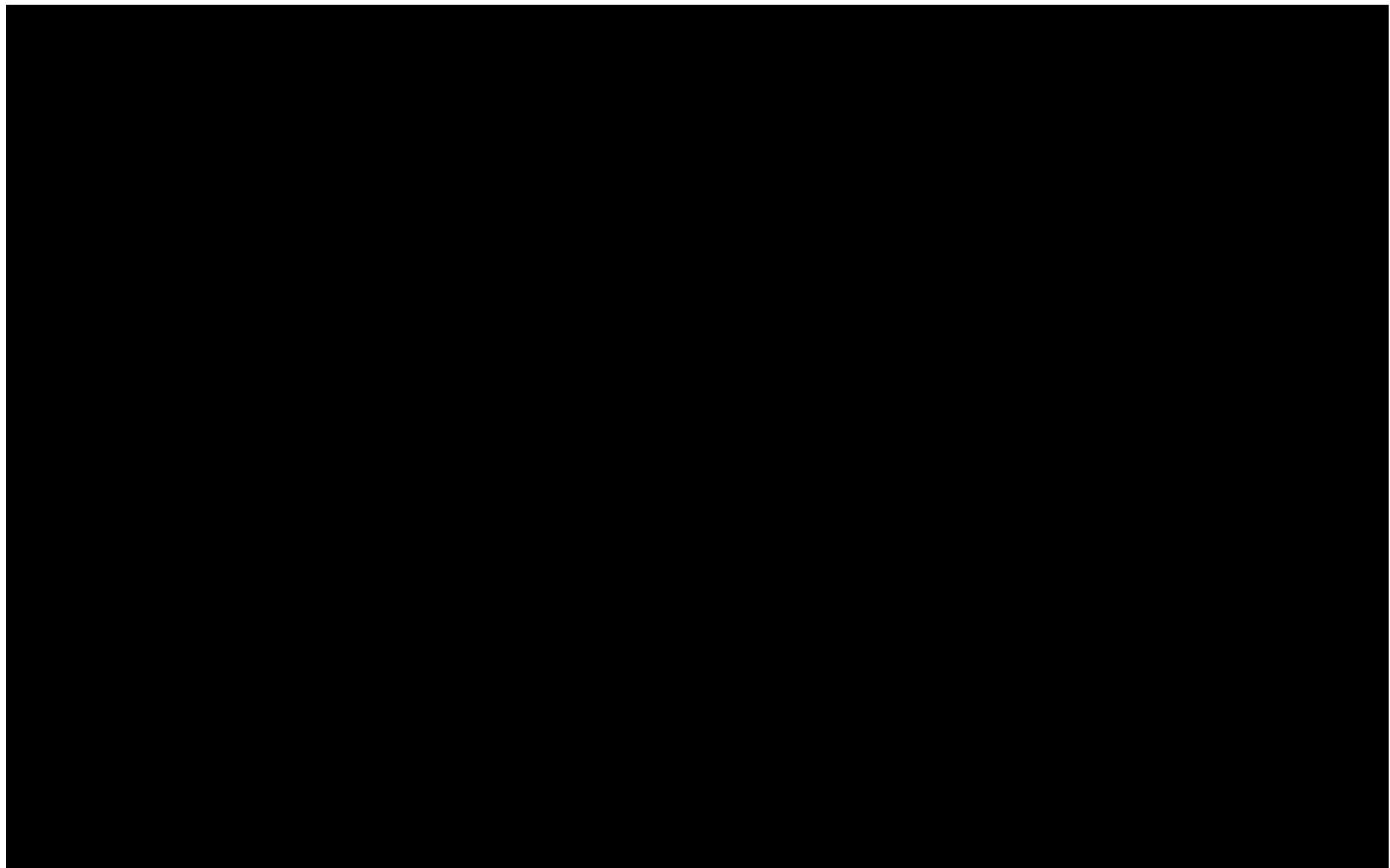


Demo

The power of Web Audio

Web Audio Concepts

- Playing sounds
- Fetching and creating sounds

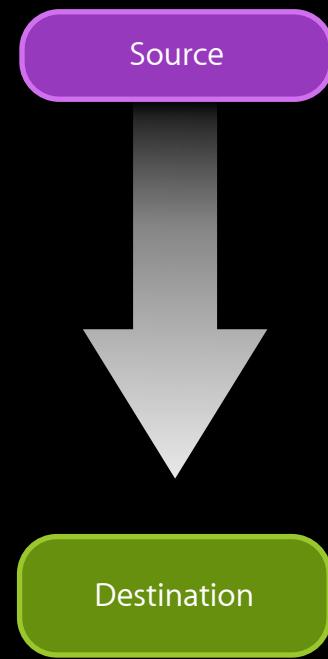


Audio Context

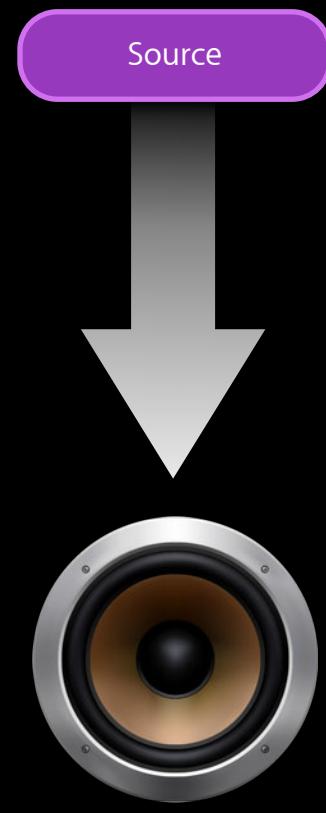
Audio Context

Source

Audio Context



Audio Context



Destination

Audio Sources

Creating an audio source from a remote resource

Audio Sources

Creating an audio source from a remote resource

- Use XMLHttpRequest to get the raw file data

Audio Sources

Creating an audio source from a remote resource

- Use XMLHttpRequest to get the raw file data

```
var request = new XMLHttpRequest();
request.open("GET", "http://sounds.example.org/explosion.aiff", true);
request.responseType = "arraybuffer";
request.addEventListener("load", processBuffer, false);
request.send();
```

Audio Sources

Creating an audio source from a remote resource

- Use XMLHttpRequest to get the raw file data

```
var request = new XMLHttpRequest();
request.open("GET", "http://sounds.example.org/explosion.aiff", true);
request.responseType = "arraybuffer";
request.addEventListener("load", processBuffer, false);
request.send();
```

- Create a buffer from the remote resource

Audio Sources

Creating an audio source from a remote resource

- Use XMLHttpRequest to get the raw file data

```
var request = new XMLHttpRequest();
request.open("GET", "http://sounds.example.org/explosion.aiff", true);
request.responseType = "arraybuffer";
request.addEventListener("load", processBuffer, false);
request.send();
```

- Create a buffer from the remote resource

```
function processBuffer(event) {
  var request = event.target;
  myBuffer = myAudioContext.createBuffer(request.response, false);
}
```

Audio Sources

Creating an audio source from a remote resource

```
var request = new XMLHttpRequest();
request.open("GET", "http://sounds.example.org/explosion.aiff", true);
request.responseType = "arraybuffer";
request.addEventListener("load", processBuffer, false);
request.send();

function processBuffer(event) {
  var request = event.target;
  myBuffer = myAudioContext.createBuffer(request.response, false);
}
```

Audio Sources

Creating an audio source from a remote resource

```
var request = new XMLHttpRequest();
request.open("GET", "http://sounds.example.org/explosion.aiff", true);
request.responseType = "arraybuffer";
request.addEventListener("load", processBuffer, false);
request.send();

function processBuffer(event) {
  var request = event.target;
  myBuffer = myAudioContext.createBuffer(request.response, false);
}
```

- Create an AudioBufferSourceNode to route buffer contents to output

Audio Sources

Creating an audio source from a remote resource

```
var request = new XMLHttpRequest();
request.open("GET", "http://sounds.example.org/explosion.aiff", true);
request.responseType = "arraybuffer";
request.addEventListener("load", processBuffer, false);
request.send();

function processBuffer(event) {
  var request = event.target;
  myBuffer = myAudioContext.createBuffer(request.response, false);
}
```

- Create an AudioBufferSourceNode to route buffer contents to output

```
var mySource = myContext.createBufferSource(); // AudioBufferSourceNode
mySource.buffer = myBuffer;
```

Audio Sources

Playing a sound at exactly the right time



Audio Sources

Playing a sound at exactly the right time



Audio Sources

Playing a sound at exactly the right time

- Use `noteOn(when)` with an `AudioBufferSourceNode`

Audio Sources

Playing a sound at exactly the right time

- Use `noteOn(when)` with an `AudioBufferSourceNode`

```
// where we left off, created an AudioBufferSourceNode  
// and "filled" it with our buffer contents  
var mySource = myContext.createBufferSource(); // AudioBufferSourceNode  
mySource.buffer = myBuffer;
```

Audio Sources

Playing a sound at exactly the right time

- Use `noteOn(when)` with an `AudioBufferSourceNode`

```
// where we left off, created an AudioBufferSourceNode  
// and "filled" it with our buffer contents  
var mySource = myContext.createBufferSource(); // AudioBufferSourceNode  
mySource.buffer = myBuffer;  
  
// connect to the default output  
mySource.connect(myContext.destination);  
  
// play the sound right now  
mySource.noteOn(0);
```

Audio Sources

Other ways to create audio sources

- From an `<audio>` element
 - use `MediaElementAudioSourceNode`



`<audio>`

Audio Sources

Other ways to create audio sources

- From an <audio> element
 - use MediaElementAudioSourceNode
- Synthesize a sound
 - use a built-in AudioSource called Oscillator
 - good for simple tones

Audio Sources

Other ways to create audio sources

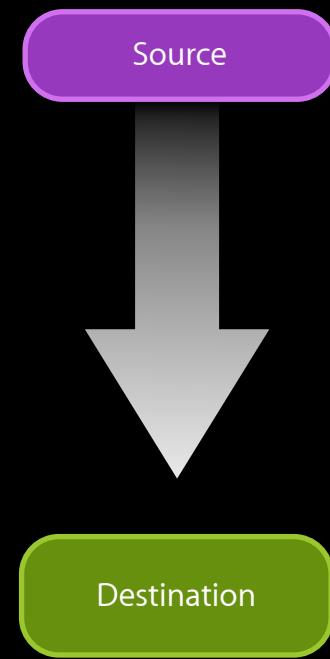
- From an <audio> element
 - use MediaElementAudioSourceNode
- Synthesize a sound
 - use a built-in AudioSource called Oscillator
 - good for simple tones
- Generate a completely custom sound from JavaScript
 - use JavaScriptAudioNode

Audio Context

Source

Destination

Audio Context

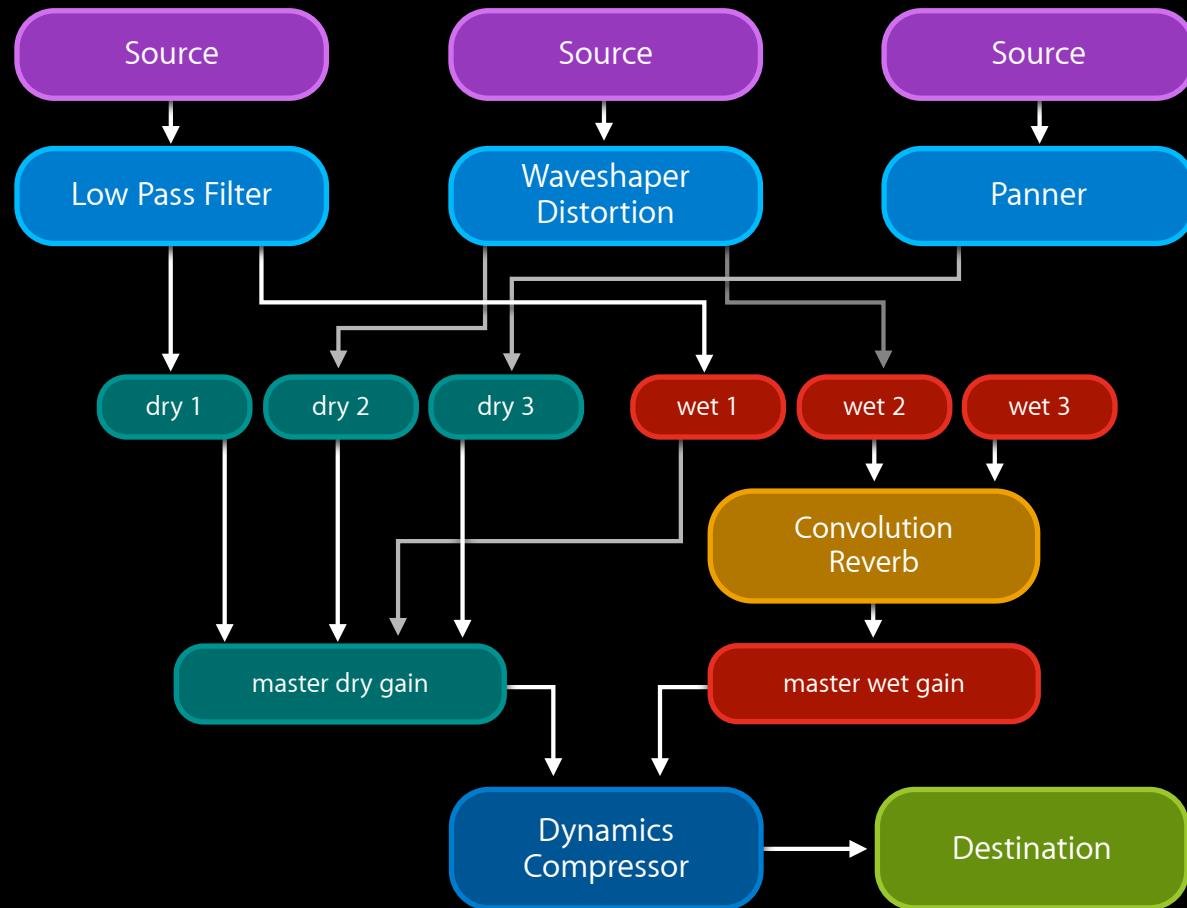


Audio Context

Source

Destination

Audio Context



New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

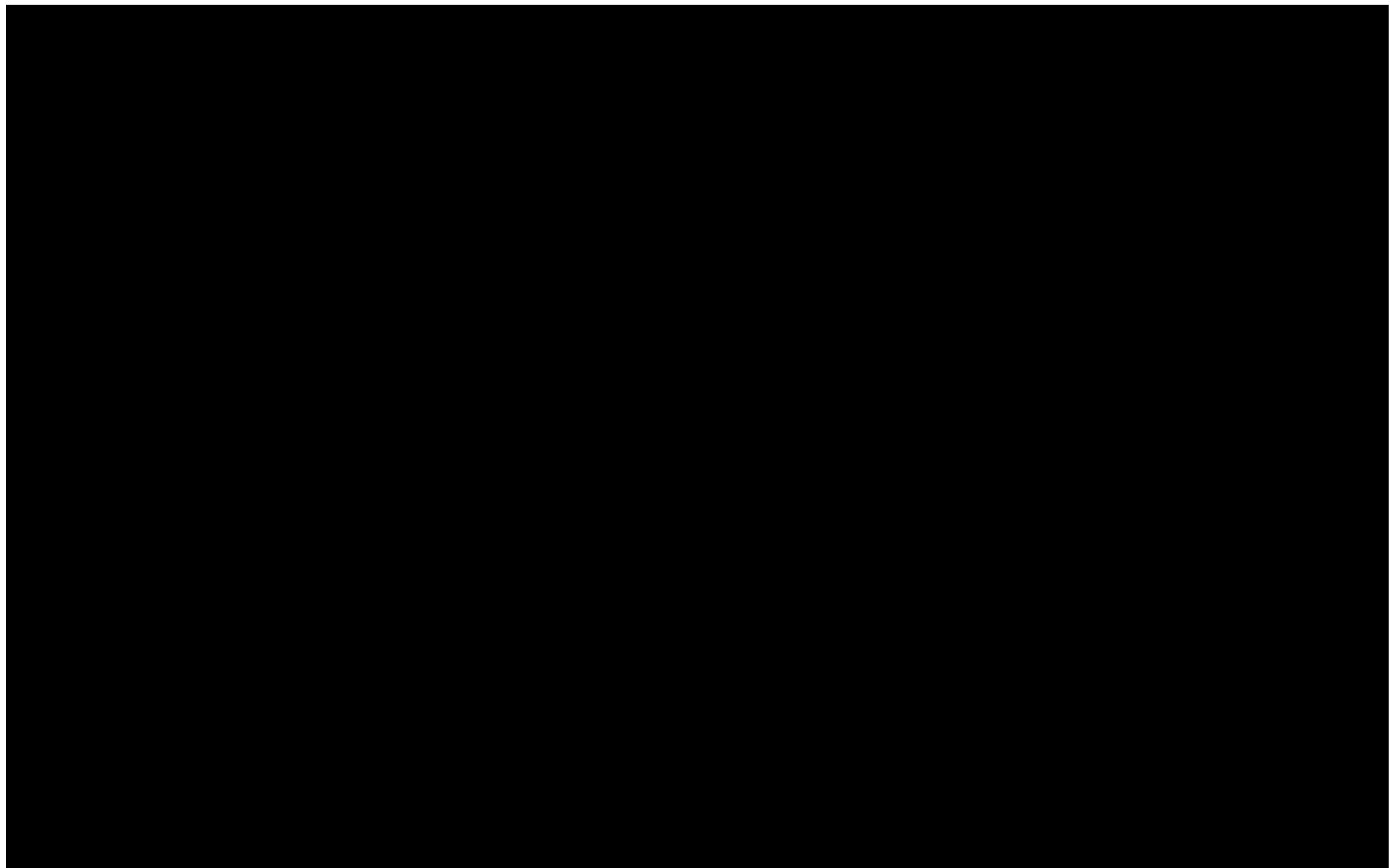
Augment your interface with audio

New Ways to Integrate Media with HTML5

Synchronize multiple media elements

Implement fullscreen interfaces

Augment your interface with audio



Summary

New HTML5 media APIs in Safari 6 and iOS 6

Summary

New HTML5 media APIs in Safari 6 and iOS 6

- Solve common problems and integrate media in new ways

Summary

New HTML5 media APIs in Safari 6 and iOS 6

- Solve common problems and integrate media in new ways



@!#%?!
A large, light gray speech bubble shape containing the text '@!#%?!' in a bold, black, sans-serif font.

Summary

New HTML5 media APIs in Safari 6 and iOS 6

- Solve common problems and integrate media in new ways

@!#%?!



Summary

New HTML5 media APIs in Safari 6 and iOS 6

- Solve common problems and integrate media in new ways
- Media synchronization and fullscreen APIs are simple and powerful

@!#%?!



Summary

New HTML5 media APIs in Safari 6 and iOS 6

- Solve common problems and integrate media in new ways
- Media synchronization and fullscreen APIs are simple and powerful
- Web Audio enables a new class of web applications

@!#%?!



More Information

Vicki Murley

Safari Technologies Evangelist

vicki@apple.com

Documentation

Safari Dev Center

<http://developer.apple.com/safari>

World Wide Web Consortium

<http://www.w3.org>

Apple Developer Forums

<http://devforums.apple.com/community/safari>

Labs

Safari & WebKit Open Lab

Safari & Web Lab
Thursday 3:15PM

Q&A

 WWDC2012