



Adding Touch and Gesture Detection to Web Pages on iPhone OS

Paul Knight

Safari on iOS and WebKit Engineer

Introduction

- Web content for fingers
- Rich web applications

What You'll Learn

- The touch event model
- Build controls with touch events
- Use the DOM and CSS Transforms to build complex interfaces

Prerequisites

- HTML, JavaScript, and CSS
- Document Object Model (DOM) Basics

The Touch Event Model

The Touch Event Model

touchstart

A finger is placed on the screen

touchmove

A finger on the screen moves

touchend

A finger is removed from the screen

touchcancel

Touches have been interrupted

Building a Button



Do Something



Do Something

Building a Button

Do Something



```
<div id="myButtonId" class="tapButton">
```


Building a Button

Do Something

A blue rounded rectangle with the text "JavaScript" in white, positioned over a white document icon with a folded top-right corner.

JavaScript

```
function callback() { ... }
```

Building a Button

Do Something

JavaScript

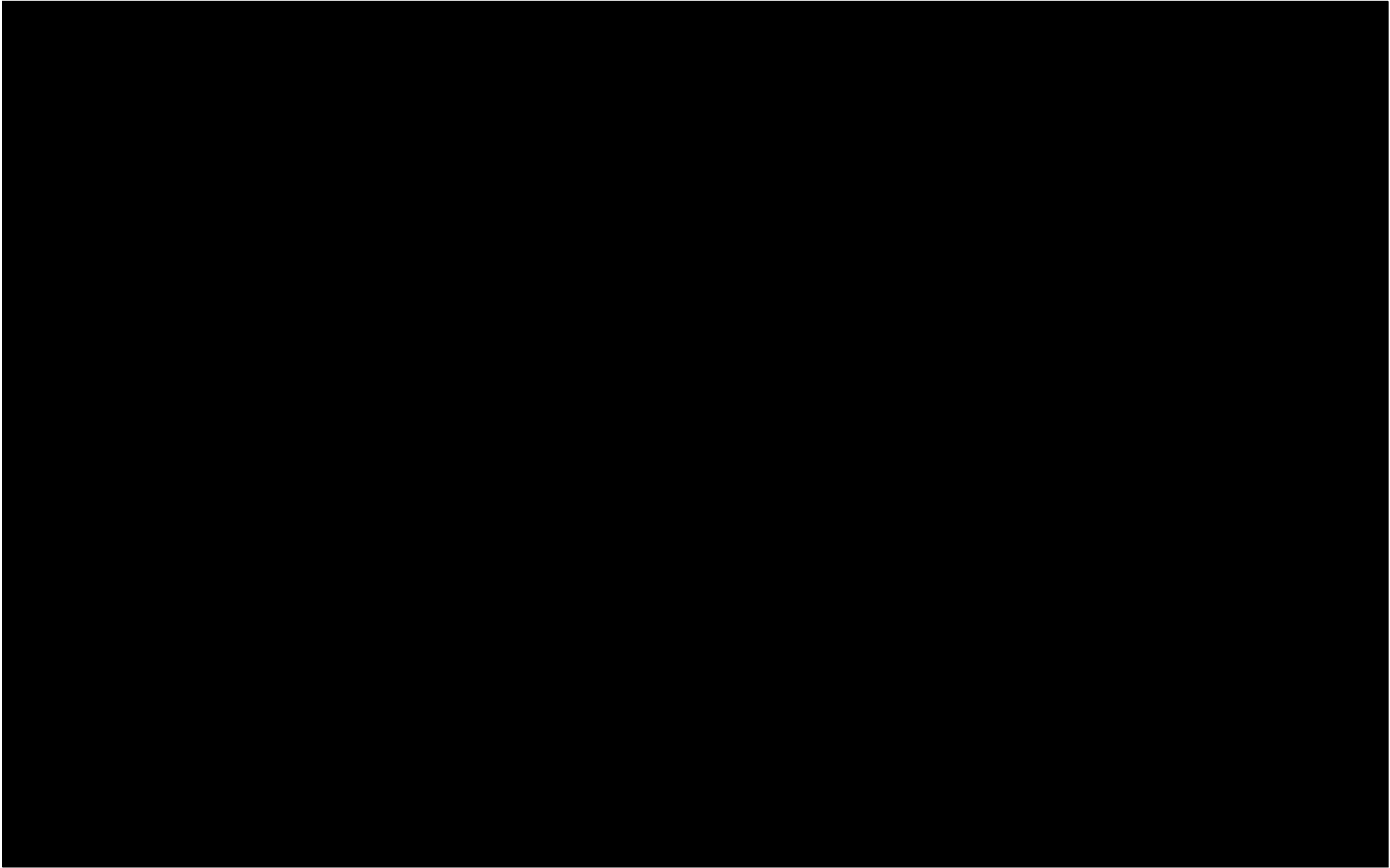
```
function callback() { ... }  
  
element = document.getElementById('myButtonId');
```

Building a Button

Do Something

JavaScript

```
function callback() { ... }  
  
element = document.getElementById('myButtonId');  
element.addEventListener('touchstart', callback, false);
```



Building a Button

Do Something

JavaScript

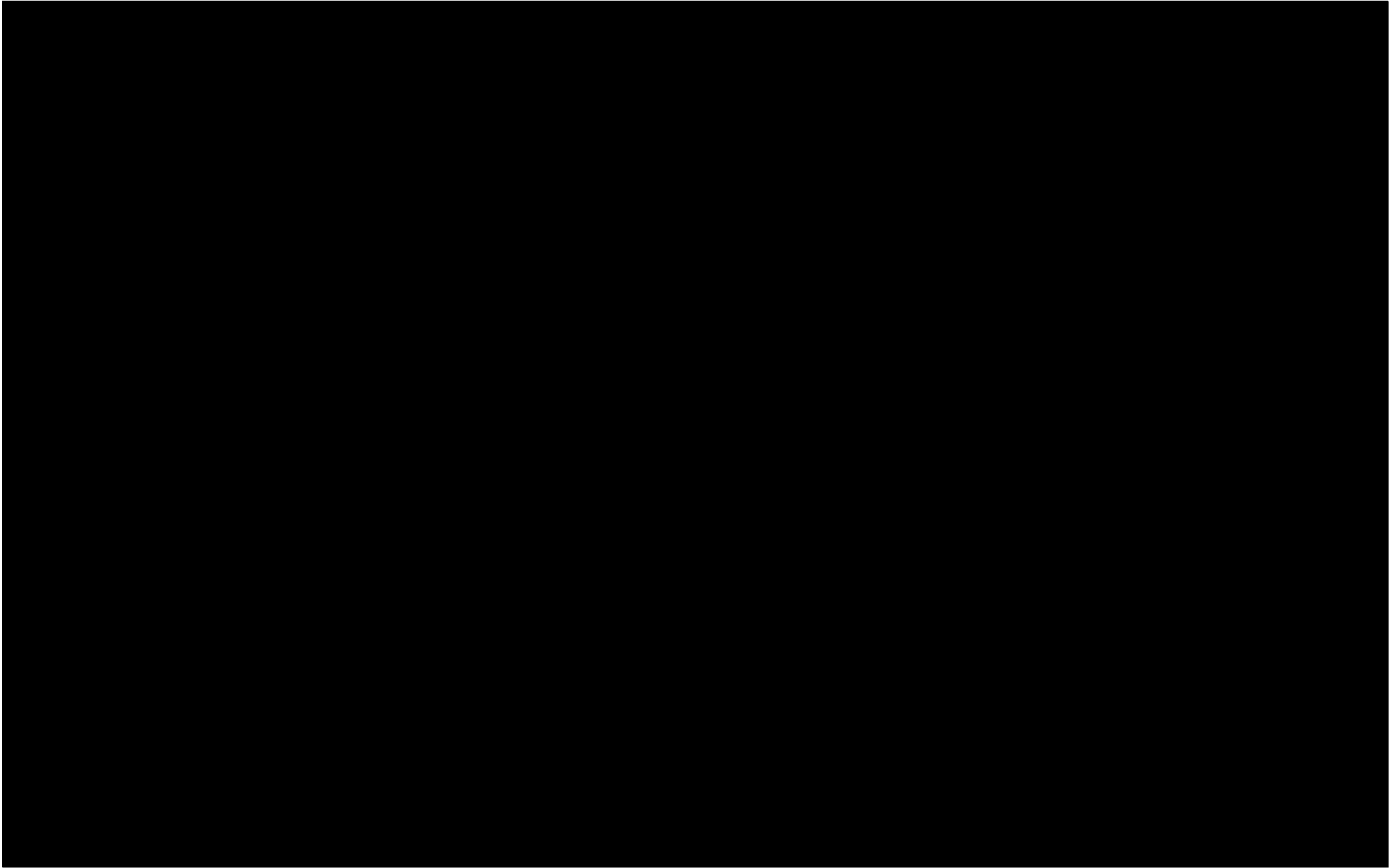
```
function callback() { ... }  
  
element = document.getElementById('myButtonId');  
element.addEventListener('touchstart', callback, false);
```

Building a Button

Do Something

JavaScript

```
function callback() { ... }  
  
element = document.getElementById('myButtonId');  
element.addEventListener('touchend', callback, false);
```



Building a Button

Do Something

A blue rounded rectangle with the text "JavaScript" in white, positioned over a white document icon with a folded corner.

JavaScript

```
function Button(elementID, callback) {  
  
}
```


Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {  
    this.element = document.getElementById(elementID);  
  
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {  
  this.element = document.getElementById(elementID);  
  this.callback = callback;  
  
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {  
  this.element = document.getElementById(elementID);  
  this.callback = callback;  
  this.element.addEventListener('touchstart', this, false);  
  this.element.addEventListener('touchend', this, false);  
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {  
  this.element = document.getElementById(elementID);  
  this.callback = callback;  
  this.element.addEventListener('touchstart', this, false);  
  this.element.addEventListener('touchend', this, false);  
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {
  this.element = document.getElementById(elementID);
  this.callback = callback;
  this.element.addEventListener('touchstart', this, false);
  this.element.addEventListener('touchend', this, false);
}

Button.prototype.handleEvent = function(event) {

}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {
  this.element = document.getElementById(elementID);
  this.callback = callback;
  this.element.addEventListener('touchstart', this, false);
  this.element.addEventListener('touchend', this, false);
}

Button.prototype.handleEvent = function(event) {
  if (typeof(this[event.type]) === "function")
}
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {
  this.element = document.getElementById(elementID);
  this.callback = callback;
  this.element.addEventListener('touchstart', this, false);
  this.element.addEventListener('touchend', this, false);
}

Button.prototype.handleEvent = function(event) {
  if (typeof(this[event.type]) === "function")
}
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {
  this.element = document.getElementById(elementID);
  this.callback = callback;
  this.element.addEventListener('touchstart', this, false);
  this.element.addEventListener('touchend', this, false);
}

Button.prototype.handleEvent = function(event) {
  if (typeof(this[event.type]) === "function")
}
}
```


Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {
  this.element = document.getElementById(elementID);
  this.callback = callback;
  this.element.addEventListener('touchstart', this, false);
  this.element.addEventListener('touchend', this, false);
}

Button.prototype.handleEvent = function(event) {
  if (typeof(this[event.type]) === "function")
    return this[event.type](event);
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {
  this.element = document.getElementById(elementID);
  this.callback = callback;
  this.element.addEventListener('touchstart', this, false);
  this.element.addEventListener('touchend', this, false);
}

Button.prototype.handleEvent = function(event) {
  if (typeof(this[event.type]) === "function")
    return this[event.type](event);
}
```

Building a Button

Do Something

JavaScript

```
Button.prototype.touchstart = function(event) {  
  }  
  
Button.prototype.touchend = function(event) {  
  }  
}
```

Building a Button

Do Something

JavaScript

```
Button.prototype.touchstart = function(event) {  
    this.element.className = 'tapButton highlighted';  
}
```

```
Button.prototype.touchend = function(event) {  
    this.element.className = 'tapButton';  
}
```

Building a Button

Do Something

JavaScript

```
Button.prototype.touchstart = function(event) {  
  this.element.className = 'tapButton highlighted';  
}
```

```
Button.prototype.touchend = function(event) {  
  this.element.className = 'tapButton';  
  this.callback();  
}
```

Building a Button

Do Something

A small icon representing a document or code file, with a teal tab on the top-left corner containing the text "CSS".

CSS

```
.tapButton {  
  height: 18px;  
  -webkit-border-radius: 9px;  
  background: -webkit-gradient(linear, from(#CCC), to(#666));  
}  
  
.tapButton.highlighted {  
  background: -webkit-gradient(linear, from(#666), to(#CCC));  
}
```

Building a Button

Do Something

A small icon representing a document or code file, with a teal tab on the top left corner containing the text "CSS".

CSS

```
.tapButton {  
  height: 18px;  
  -webkit-border-radius: 9px;  
  background: -webkit-gradient(linear, from(#CCC), to(#666));  
}  
  
.tapButton.highlighted {  
  background: -webkit-gradient(linear, from(#666), to(#CCC));  
}
```

Building a Button

Do Something

A small icon representing a document or code file, with a teal tab on the top-left corner containing the text "CSS".

CSS

```
.tapButton {  
  height: 18px;  
  -webkit-border-radius: 9px;  
  background: -webkit-gradient(linear, from(#CCC), to(#666));  
}  
  
.tapButton.highlighted {  
  background: -webkit-gradient(linear, from(#666), to(#CCC));  
}
```


Building a Button

Do Something

A small icon representing a document with a teal tab that says "CSS".

CSS

```
.tapButton {  
  height: 18px;  
  -webkit-border-radius: 9px;  
  background: -webkit-gradient(linear, from(#CCC), to(#666));  
}  
  
.tapButton.highlighted {  
  background: -webkit-gradient(linear, from(#666), to(#CCC));  
}
```



Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {  
    ...  
    this.element.addEventListener('touchcancel', this, false);  
}  
  
Button.prototype.touchcancel = function(event) {  
  
}
```

Building a Button

Do Something

JavaScript

```
function Button(elementID, callback) {  
    ...  
    this.element.addEventListener('touchcancel', this, false);  
}  
  
Button.prototype.touchcancel = function(event) {  
    this.element.className = 'tapButton';  
}
```

Building a Button



Do Something



Do Something

The Touch Event Model

touchstart

A finger is placed on the screen

touchmove

A finger on the screen moves

touchend

A finger is removed from the screen

touchcancel

Touches have been interrupted

The TouchEvent Object

MouseEvent

unsigned short button

long pageX
long pageY

long layerX
long layerY

boolean altKey
boolean ctrlKey
boolean shiftKey

...

TouchEvent

TouchList touches
TouchList changedTouches
TouchList targetTouches

The TouchEvent Object

TouchEvent

TouchList touches
TouchList changedTouches
TouchList targetTouches

Touch

unsigned long identifier

EventTarget target

long screenX
long screenY

long pageX
long pageY

tifier

tifier

The TouchEvent Object

TouchEvent

TouchList **touches**

All touches currently active (down) on the screen

TouchList **changedTouches**

Touches that have changed since the last TouchEvent

TouchList **targetTouches**

Touches that are inside the target element

The TouchEvent Object

TouchEvent

TouchEvent.touches
TouchEvent.changedTouches
TouchEvent.targetTouches

`e.touches.length`

`e.targetTouches[0]`

The TouchEvent Object

TouchEvent

TouchEvent.touches
TouchEvent.changedTouches
TouchEvent.targetTouches

Touch

Touch.identifier

Touch.target

Touch.screenX
Touch.screenY

Touch.pageX
Touch.pageY

Touch.identifier

Touch.identifier

The Touch Object

Touch



The Touch Object

Touch

unsigned long identifier

The Touch Object

Touch

unsigned long identifier

EventTarget target

The Touch Object

Touch

unsigned long identifier

EventTarget target

long screenX
long screenY

long pageX
long pageY



screenX
screenY





pageX
pageY



screenX
screenY

pageX
pageY

Building a Slider



Building a Slider



```
<div id="mySliderID" class="slider">  
  <div class="track"></div>  
  <div class="knob"></div>  
</div>
```


Building a Slider



JavaScript

```
function Slider(elementID, callback) {  
    var element = document.getElementById(elementID);  
  
    this.knob = element.getElementsByClassName('knob')[0];  
    this.knobWidth = this.knob.offsetWidth;  
  
}
```


Building a Slider



JavaScript

```
function Slider(elementID, callback) {  
    var element = document.getElementById(elementID);  
  
    this.knob = element.getElementsByClassName('knob')[0];  
    this.knobWidth = this.knob.offsetWidth;  
  
    this.bar = element.getElementsByClassName('bar')[0];  
    this.barWidth = this.bar.offsetWidth;  
}
```

Building a Slider



JavaScript

```
touchstart : function(event) {  
  
},
```

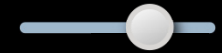
Building a Slider



JavaScript

```
touchstart : function(event) {  
    this.startX = event.targetTouches[0].pageX;  
},
```

Building a Slider



JavaScript

```
touchstart : function(event) {  
    this.startX = event.targetTouches[0].pageX;  
  
},
```

Building a Slider



JavaScript

```
touchstart : function(event) {  
    this.startX = event.targetTouches[0].pageX;  
  
},
```

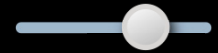
Building a Slider



JavaScript

```
touchstart : function(event) {  
    this.startX = event.targetTouches[0].pageX;  
    this.moveKnobTo(this.startX);  
},
```

Building a Slider



JavaScript

```
touchmove : function(event) {  
  
},
```

Building a Slider



JavaScript

```
touchmove : function(event) {  
    this.moveKnobTo(event.targetTouches[0].pageX);  
},
```


Building a Slider



JavaScript

```
moveKnobTo : function(x) {
```

```
},
```

Building a Slider



JavaScript

```
moveKnobTo : function(x) {  
    var delta = x - this.startX;
```

```
},
```

Building a Slider



JavaScript

```
moveKnobTo : function(x) {  
  var delta = x - this.startX;  
  this.x = Math.min(this.x + delta, this.barWidth - this.knobWidth);  
  this.x = Math.max(this.x, 0);  
  
  },
```

Building a Slider



JavaScript

```
moveKnobTo : function(x) {  
  var delta = x - this.startX;  
  this.x = Math.min(this.x + delta, this.barWidth - this.knobWidth);  
  this.x = Math.max(this.x, 0);  
  
  this.knob.style.left = this.x + 'px';  
  
},
```

Building a Slider



JavaScript

```
moveKnobTo : function(x) {  
    var delta = x - this.startX;  
    this.x = Math.min(this.x + delta, this.barWidth - this.knobWidth);  
    this.x = Math.max(this.x, 0);  
  
    this.knob.style.left = this.x + 'px';  
  
    this.startX = x;  
  
},
```

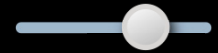
Building a Slider



JavaScript

```
moveKnobTo : function(x) {  
  var delta = x - this.startX;  
  this.x = Math.min(this.x + delta, this.barWidth - this.knobWidth);  
  this.x = Math.max(this.x, 0);  
  
  this.knob.style.left = this.x + 'px';  
  
  this.startX = x;  
  
  var percentage = this.x / (this.barWidth - this.knobWidth);  
  this.callback(percentage);  
},
```

Building a Slider



JavaScript

```
moveKnobTo : function(x) {  
    var delta = x - this.startX;  
    this.x = Math.min(this.x + delta, this.barWidth - this.knobWidth);  
    this.x = Math.max(this.x, 0);  
  
    this.knob.style.left = this.x + 'px';  
  
    this.startX = x;  
  
    var percentage = this.x / (this.barWidth - this.knobWidth);  
    this.callback(percentage);  
},
```

Building a Slider



JavaScript

```
moveKnobTo : function(x) {  
    var delta = x - this.startX;  
    this.x = Math.min(this.x + delta, this.barWidth - this.knobWidth);  
    this.x = Math.max(this.x, 0);  
  
    this.knob.style.webkitTransform =  
        'translate3d(' + this.x + 'px, 0, 0)';  
  
    this.startX = x;  
  
    var percentage = this.x / (this.barWidth - this.knobWidth);  
    this.callback(percentage);  
},
```


Building a Slider



JavaScript

```
moveKnobTo : function(x) {
  var delta = x - this.startX;
  this.x = Math.min(this.x + delta, this.barWidth - this.knobWidth);
  this.x = Math.max(this.x, 0);

  this.knob.style.webkitTransform =
    'translate3d(' + this.x + 'px, 0, 0)';

  this.startX = x;

  var percentage = this.x / (this.barWidth - this.knobWidth);
  this.callback(percentage);
},
```

Building a Slider



Demo

Digital scrapbook

Greg Bolsinga

Safari on iOS and WebKit Engineer

Gesture Events

A second finger appears!

The Gesture Event Model

gesturestart

A second finger has been placed on the screen

gesturechange

Multitouch gesture

gestureend

Second finger removed from screen

The GestureEvent Object

GestureEvent



The GestureEvent Object

GestureEvent

float scale

The GestureEvent Object

GestureEvent

float scale

float rotation

The GestureEvent Object

GestureEvent

float scale
float rotation

long pageX
long pageY

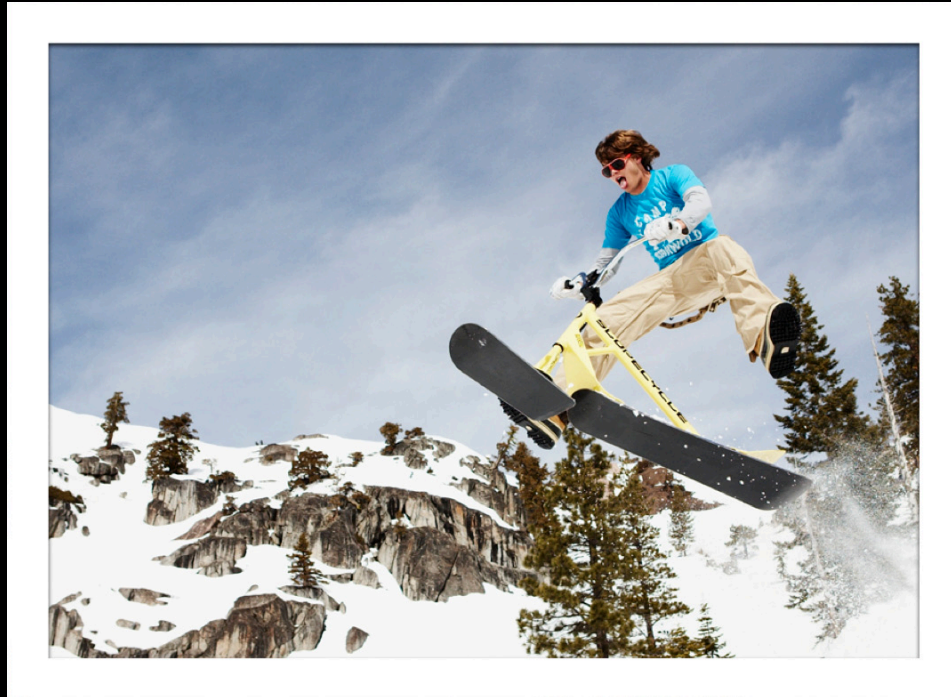
Picture Frame

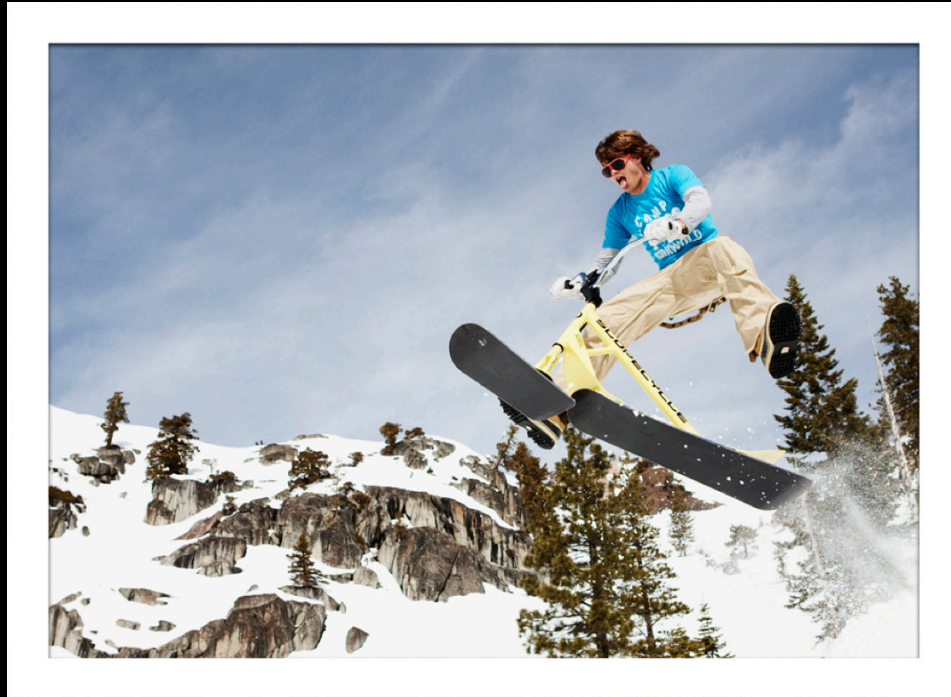


```
<div class="pictureFrame">
```



```
</img>
```





Building a Picture Frame

HTML

```
<div class="pictureFrame">  
    
</div>
```



Building a Picture Frame

JavaScript

```
function PictureFrame(element) {  
    this.scale = 1.0;  
    this.startScale = 1.0;  
    this.rotation = 0;  
    this.startRotation = 0;  
  
}
```



Building a Picture Frame

JavaScript



```
function PictureFrame(element) {  
    this.scale = 1.0;  
    this.startScale = 1.0;  
    this.rotation = 0;  
    this.startRotation = 0;  
  
    this.image = element.getElementsByTagName('img')[0];  
  
}
```

Building a Picture Frame

JavaScript



```
function PictureFrame(element) {  
  this.scale = 1.0;  
  this.startScale = 1.0;  
  this.rotation = 0;  
  this.startRotation = 0;  
  
  this.image = element.getElementsByTagName('img')[0];  
  this.image.addEventListener('gesturechange', this, false);  
}
```

Building a Picture Frame

JavaScript

```
gesturechange : function(e) {
```

```
},
```



Building a Picture Frame

JavaScript

```
gesturechange : function(e) {  
    this.scale = this.startScale * e.scale;  
  
    },
```



Building a Picture Frame

JavaScript

```
gesturechange : function(e) {  
    this.scale = this.startScale * e.scale;  
    this.rotation = this.startRotation + e.rotation;  
  
    },
```



Building a Picture Frame

JavaScript

```
gesturechange : function(e) {  
    this.scale = this.startScale * e.scale;  
    this.rotation = this.startRotation + e.rotation;  
  
    this.transformImage();  
},
```



Building a Picture Frame

JavaScript

```
transformImage : function() {
```

```
},
```



Building a Picture Frame

JavaScript

```
transformImage : function() {  
    this.image.style.webkitTransform =  
  
    },
```



Building a Picture Frame

JavaScript



```
transformImage : function() {  
    this.image.style.webkitTransform =  
        'translate3d(' + this.x + 'px, ' + this.y + 'px, 0) ' +  
  
    },
```

Building a Picture Frame

JavaScript



```
transformImage : function() {  
    this.image.style.webkitTransform =  
        'translate3d(' + this.x + 'px, ' + this.y + 'px, 0) ' +  
        'scale(' + this.scale + ') ' +  
    },
```

Building a Picture Frame

JavaScript



```
transformImage : function() {  
  this.image.style.webkitTransform =  
    'translate3d(' + this.x + 'px, ' + this.y + 'px, 0) ' +  
    'scale(' + this.scale + ') ' +  
    'rotate(' + this.rotation + 'deg)';  
},
```

Building a Picture Frame

JavaScript



```
transformImage : function() {  
  this.image.style.webkitTransform =  
    'translate3d(' + this.x + 'px, ' + this.y + 'px, 0) ' +  
    'scale(' + this.scale + ') ' +  
    'rotate(' + this.rotation + 'deg)';  
},
```

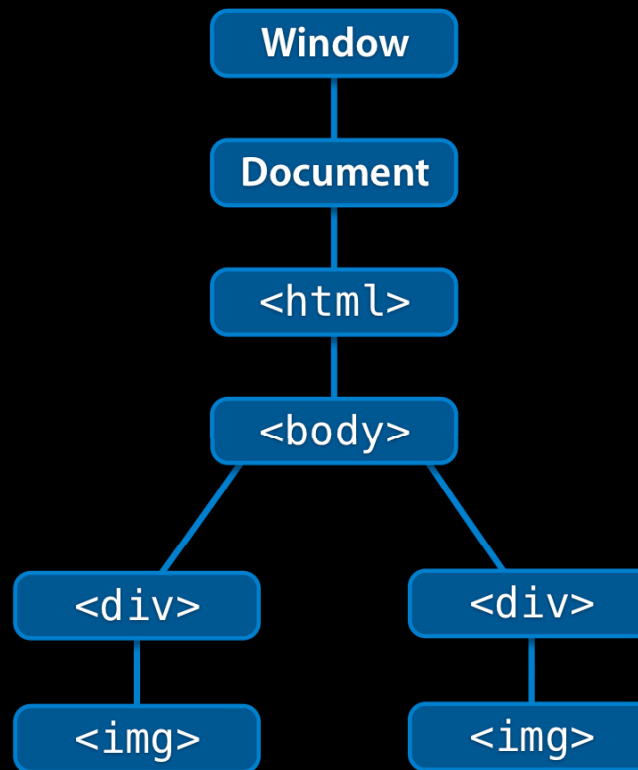
```
"translate3d(-40px, -20px, 0) scale(1.4) rotate(30deg)"
```

Picture Frame

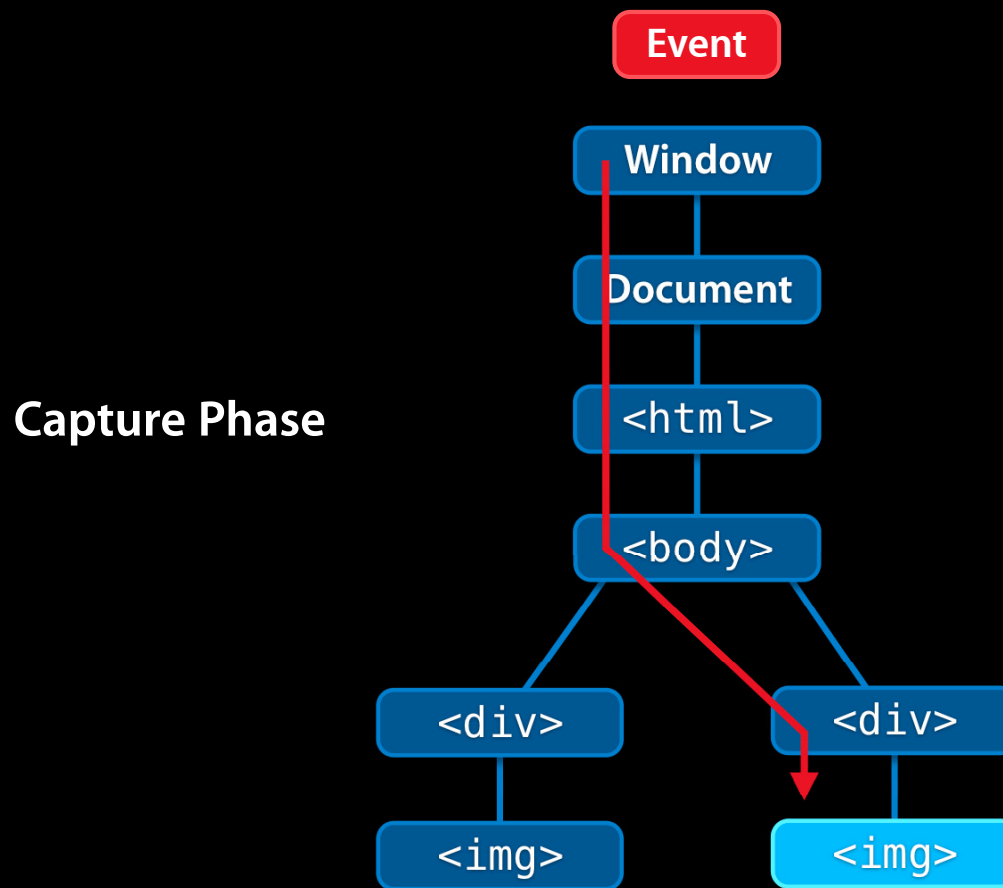


The DOM Event Model

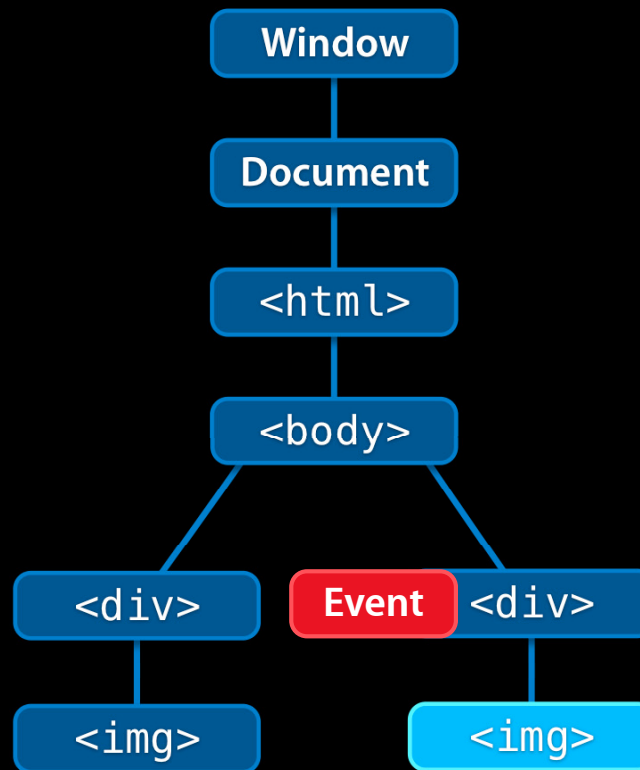
The DOM Event Model



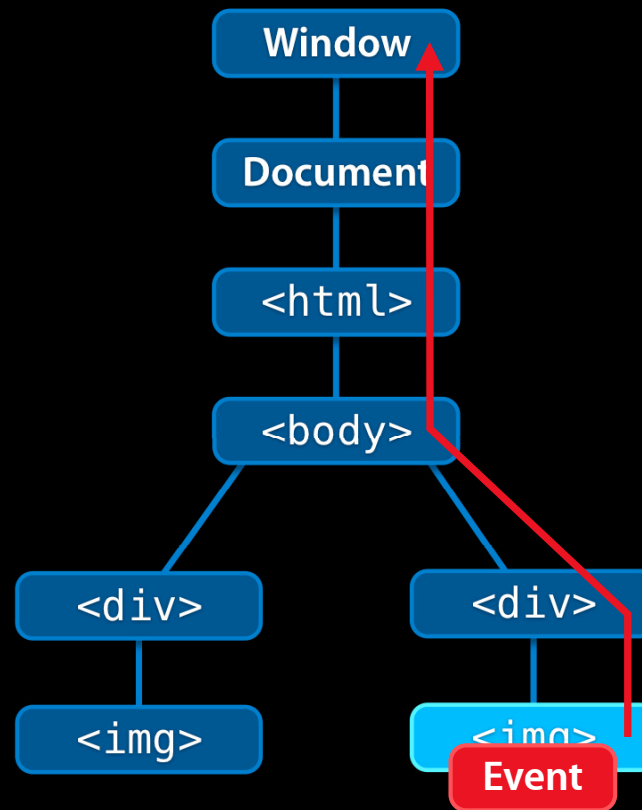
The DOM Event Model



The DOM Event Model

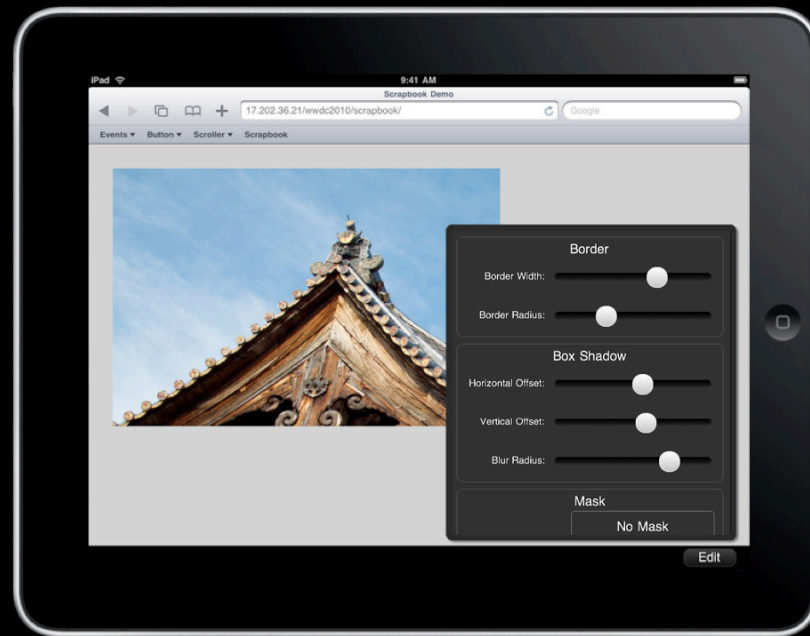


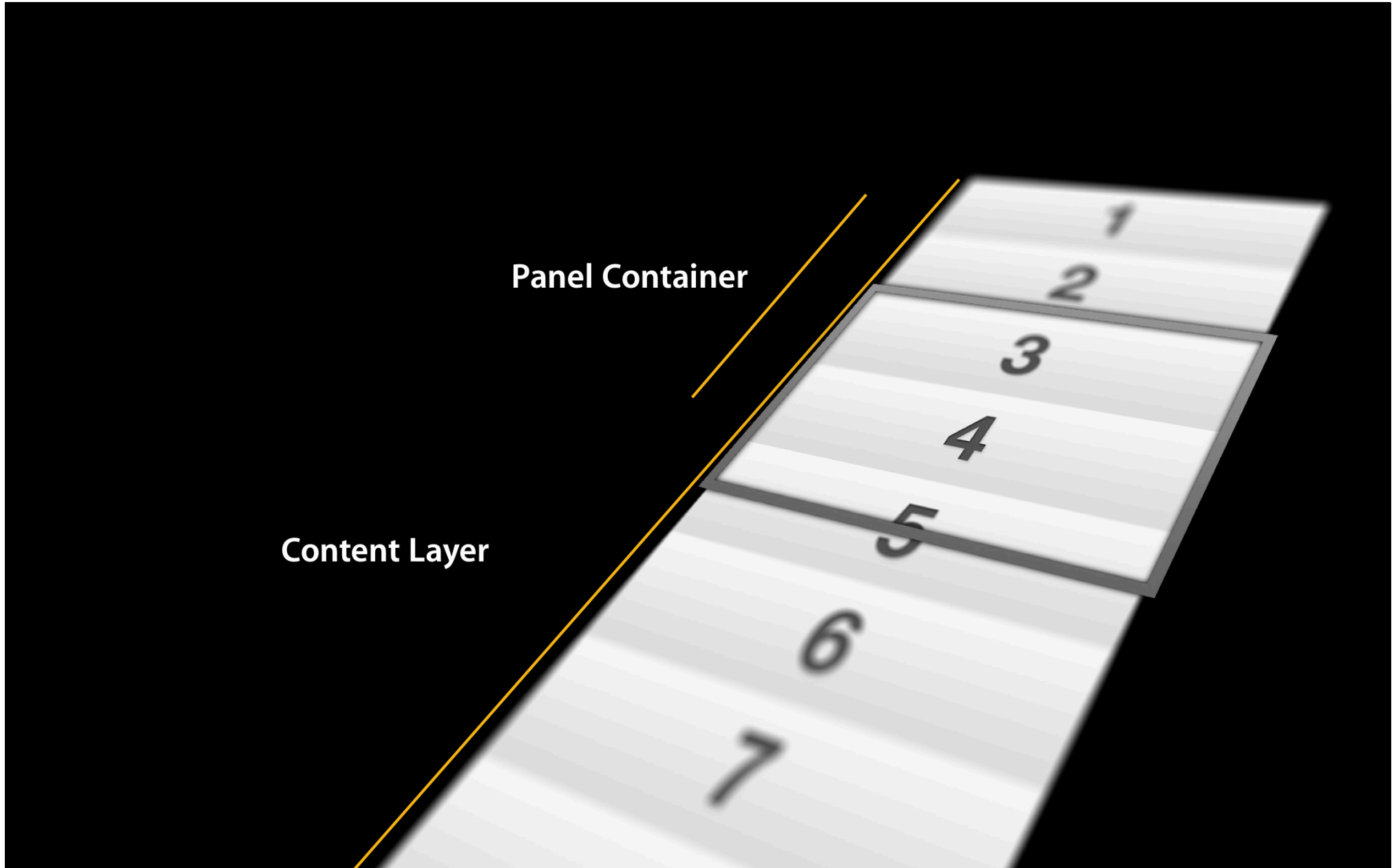
The DOM Event Model



Bubbling Phase

Scrolling Pane

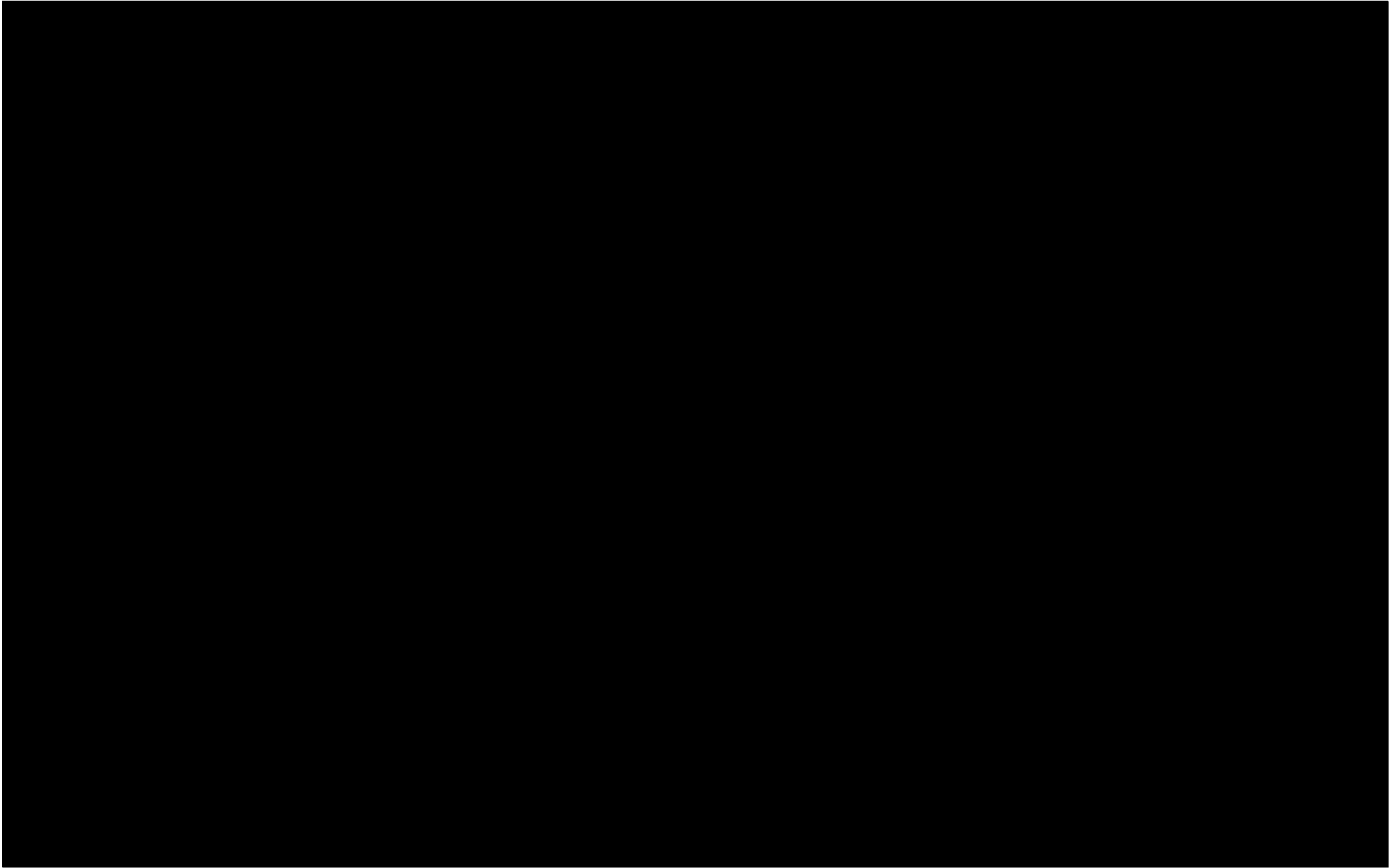




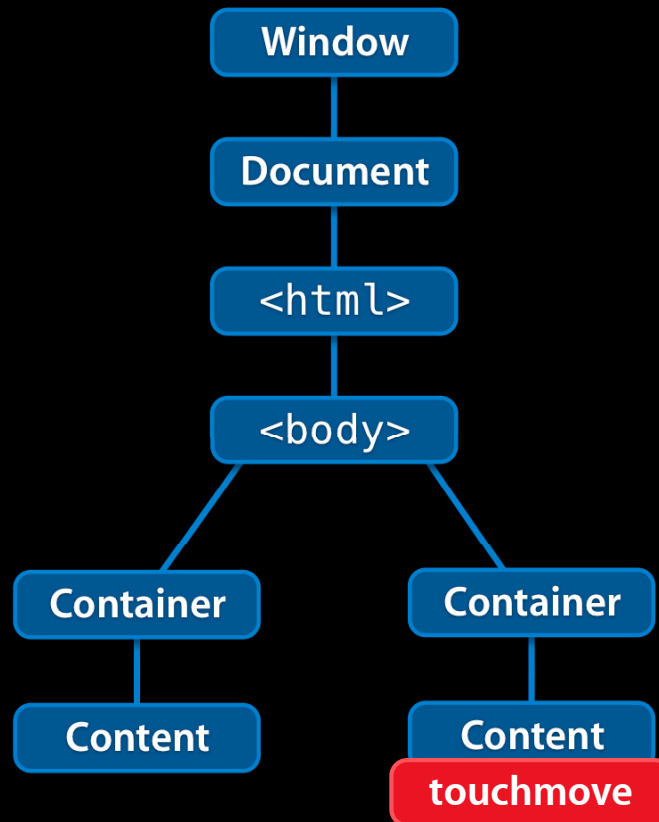
Building a Scrolling Pane



```
<div class="scrollable">  
  <div class="content-layer">  
    . . .  
  </div>  
</div>
```

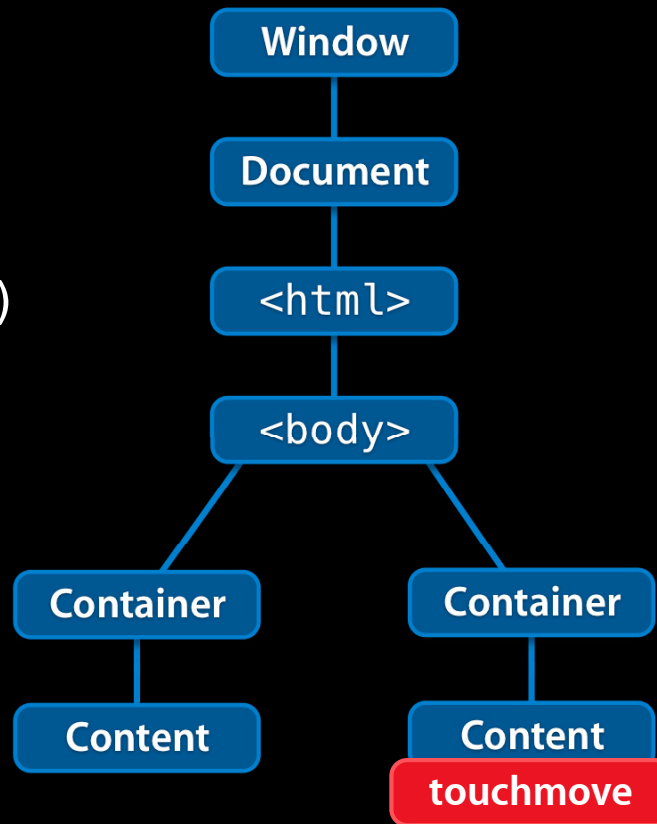


The DOM Event Model



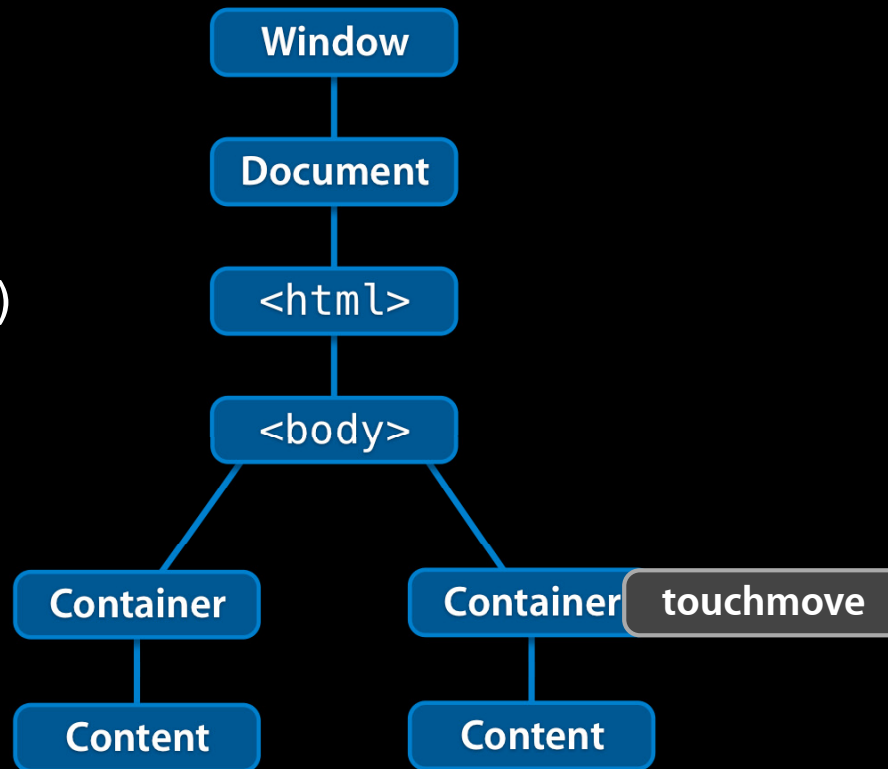
The DOM Event Model

e.preventDefault()



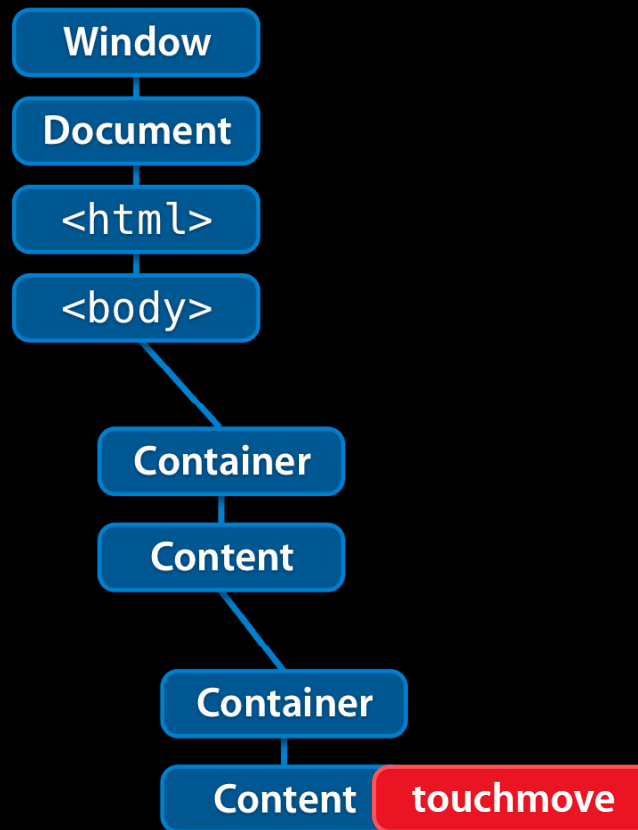
The DOM Event Model

`e.preventDefault()`

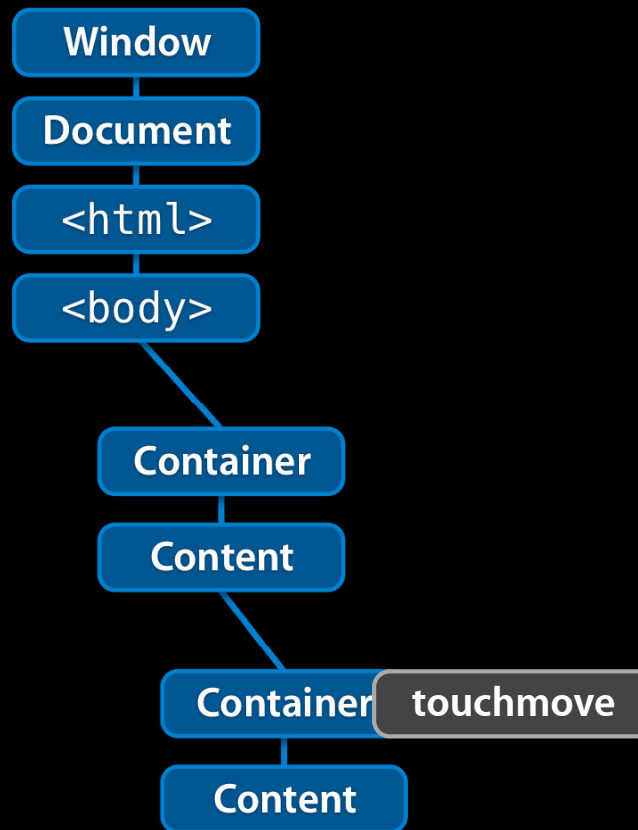




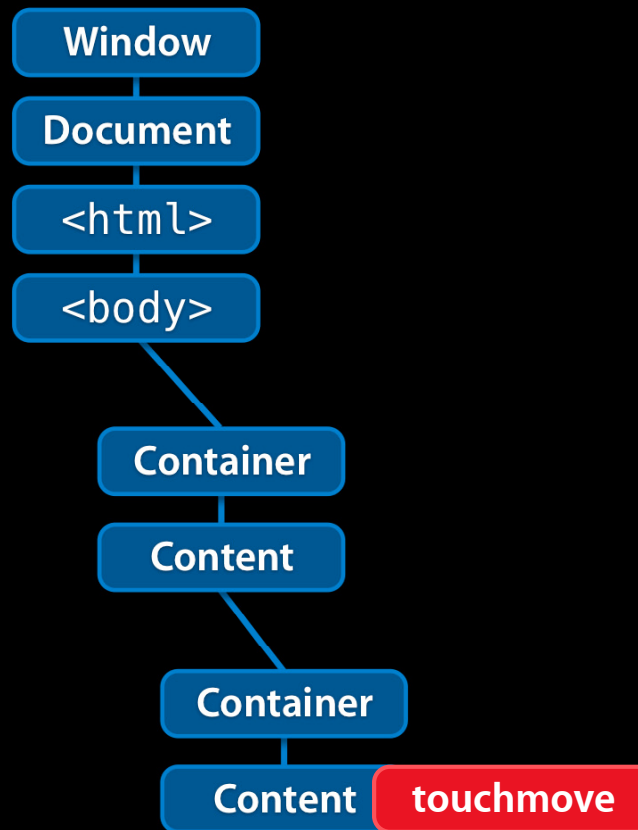
The DOM Event Model



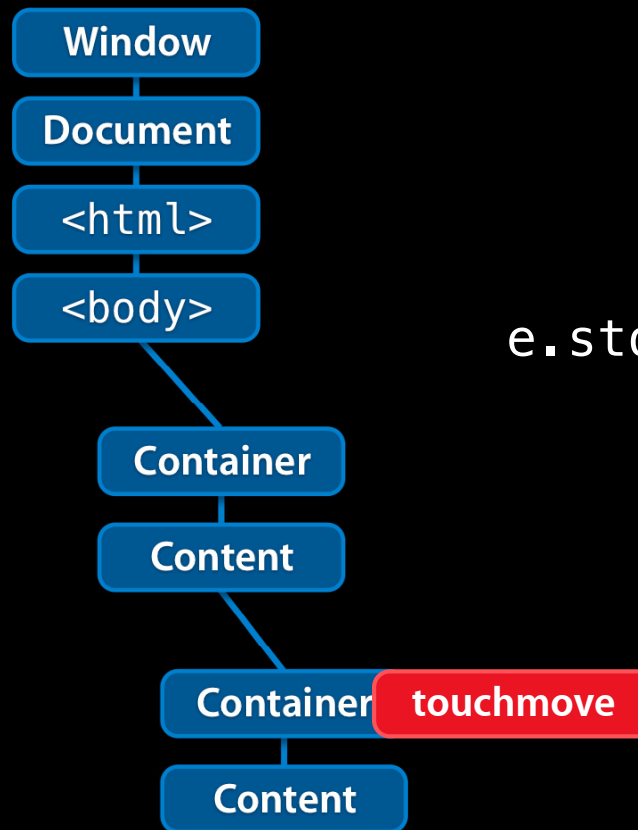
The DOM Event Model



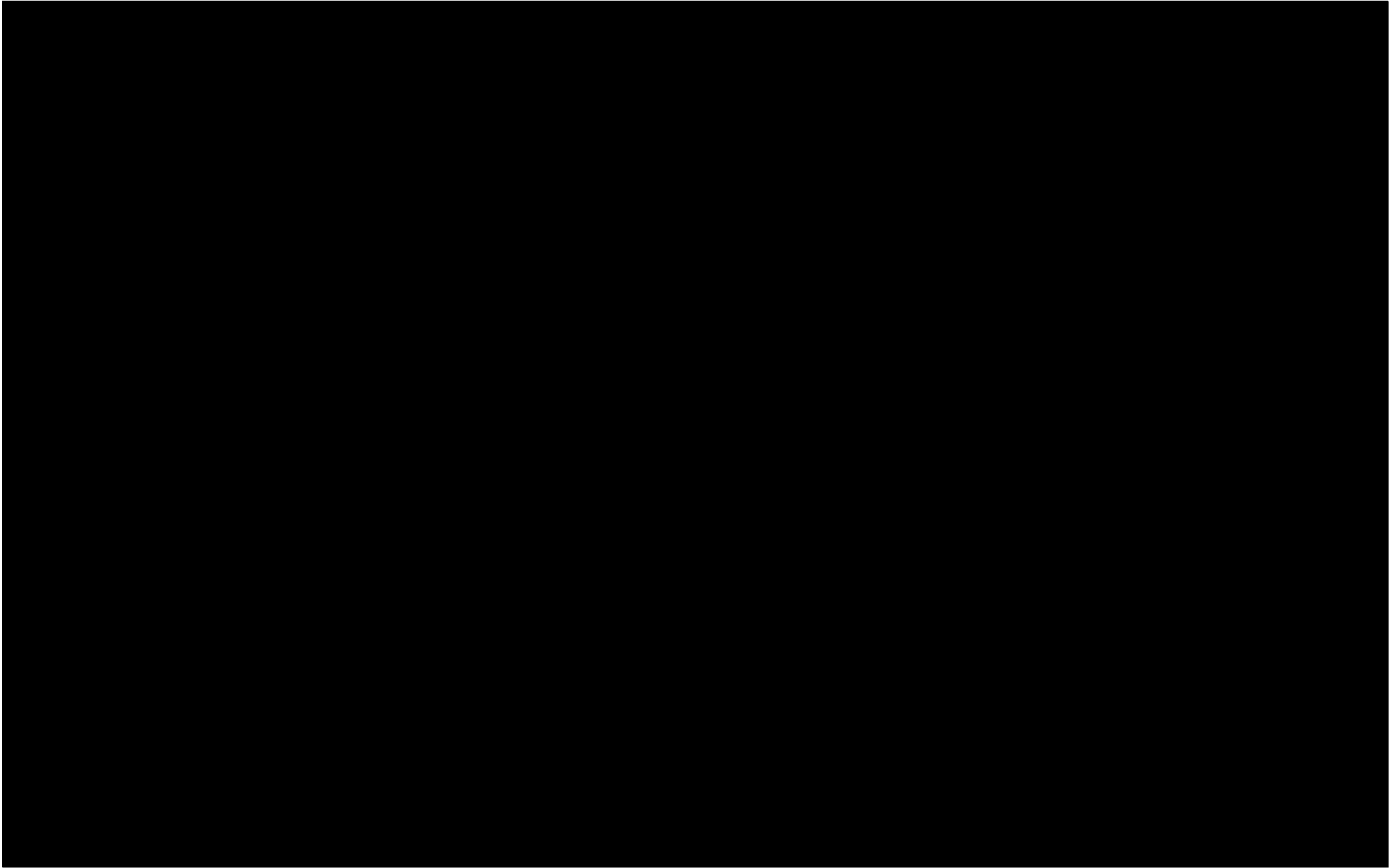
The DOM Event Model



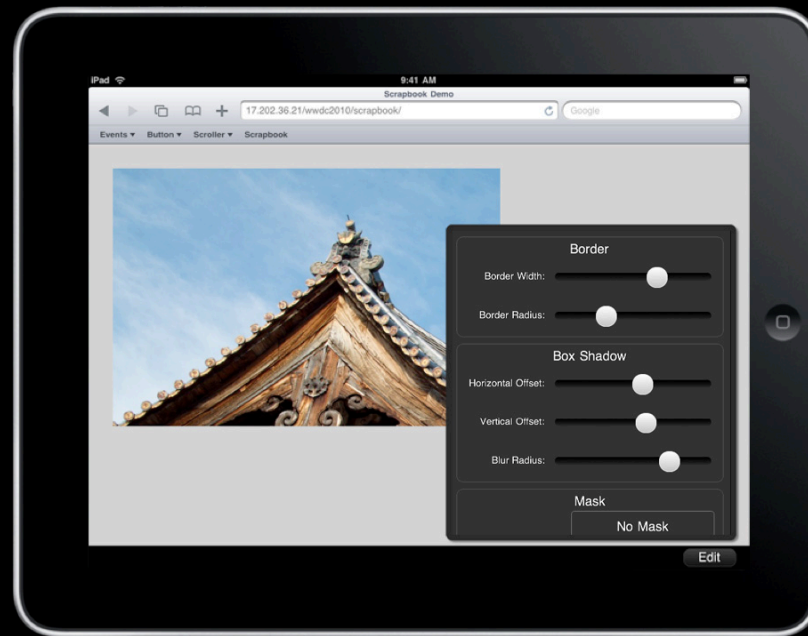
The DOM Event Model



`e.stopPropagation()`



Scrolling Pane



Last-minute Super Awesome Tips

- A TouchEvent object is a list of Touch objects
- Always handle touchcancel
- Test on hardware

Summary

- Use Touch and Gesture Events to build rich web applications
- Strategies for building reusable interface components
- Use CSS Transforms and the DOM Event Model

More Information

Vicki Murley

Safari Technologies Evangelist

vicki@apple.com

Documentation

Safari and WebKit

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

DOM Level 3 Events Specification

<http://www.w3.org/TR/DOM-Level-3-Events/>

Apple Developer Forums

<http://devforums.apple.com>

Labs

Safari on iPhone OS Lab

Internet and Web Lab B
Thursday 9:00AM

Safari on iPhone OS Lab

Internet and Web Lab B
Thursday 2:00PM

Safari Open Lab

Internet and Web Lab A
Friday 9:00AM

Q&A



