

Exploring HTML5 With JavaServer Faces 2.0

Roger Kitain
Oracle Corporation

JAZOON

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY
1-3 JUNE 2010, ZURICH

ORACLE

ORACLE

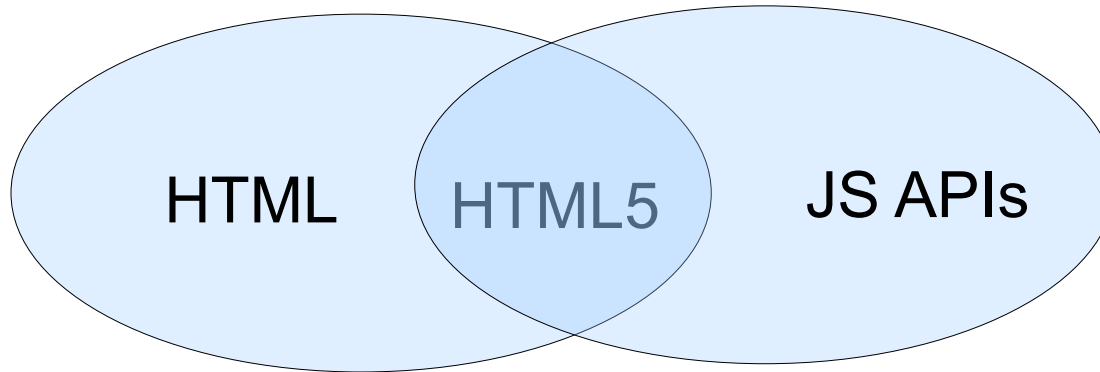
netcetera

AGENDA

- > HTML 5 Overview
 - What is HTML 5?
 - HTML 5 Features
- > JSF 2.0 Component Model And HTML 5
 - Component Model Overview
 - Enhancing Components With HTML 5
- > Demos

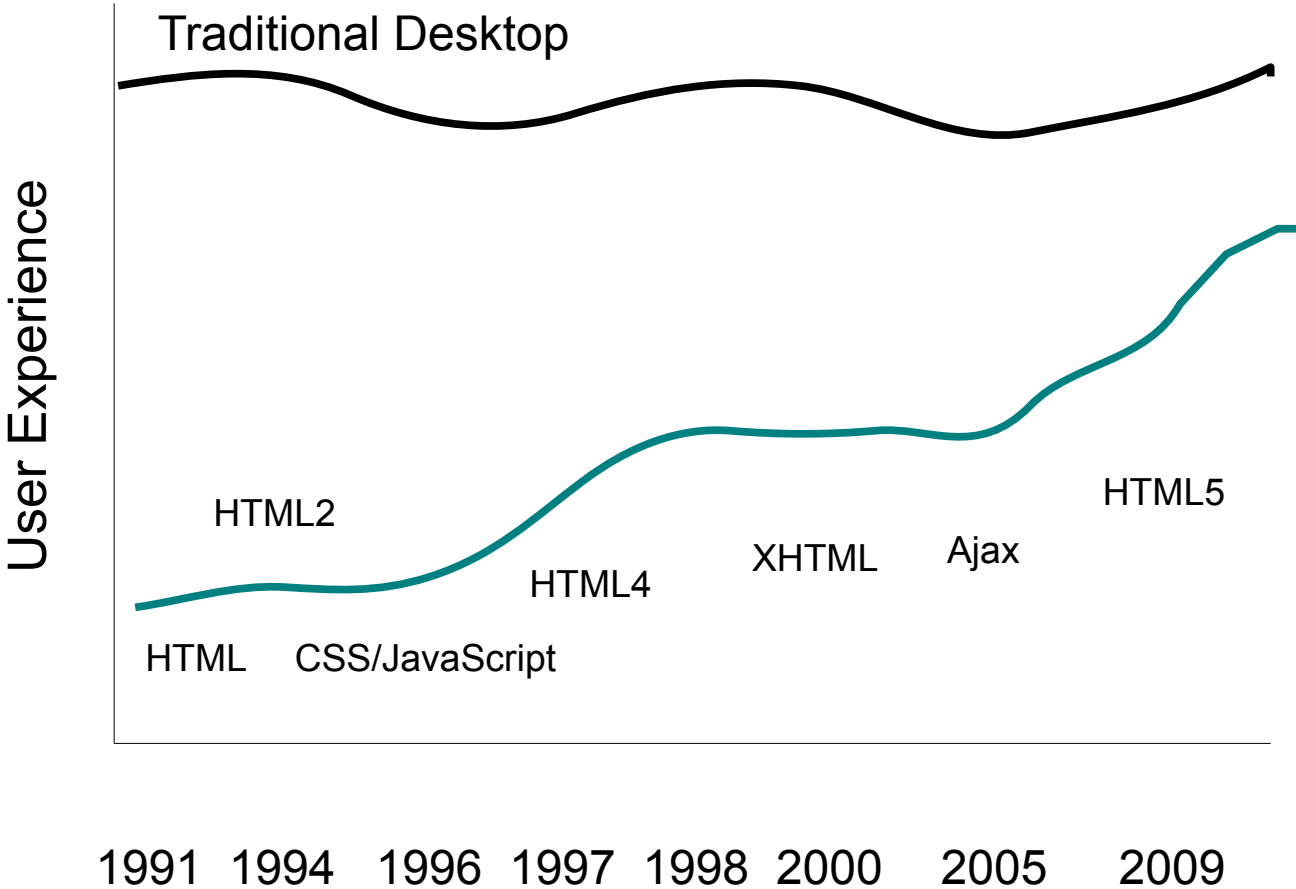
What Is HTML 5?

- > Proposed next standard for HTML 4.0.1, XHTML 1.0 and DOM Level 2 HTML
- > Features promote RIA



- > Pioneered in 2004; First working spec draft: 2008
- > CSS 3 : Working Draft – April 2010

What Is HTML 5?



What is HTML 5? When Will Specification Go Final?



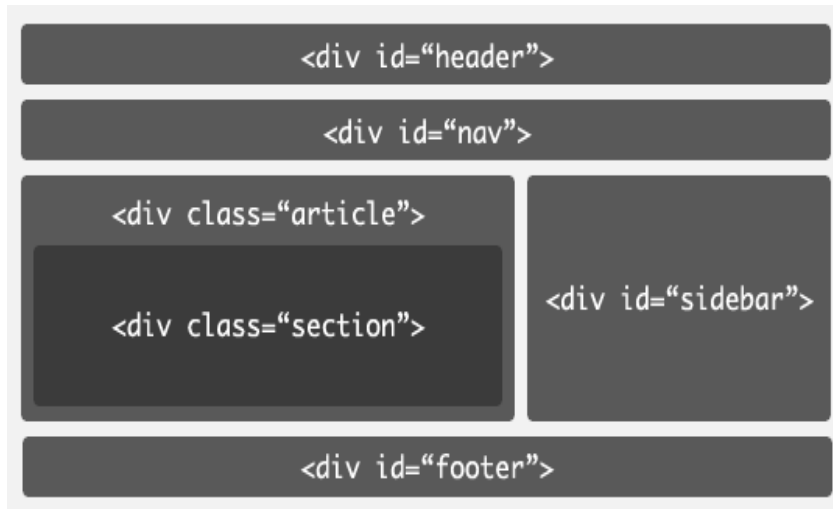
At least that's the rumor.....

HTML 5 Features: HTML Elements

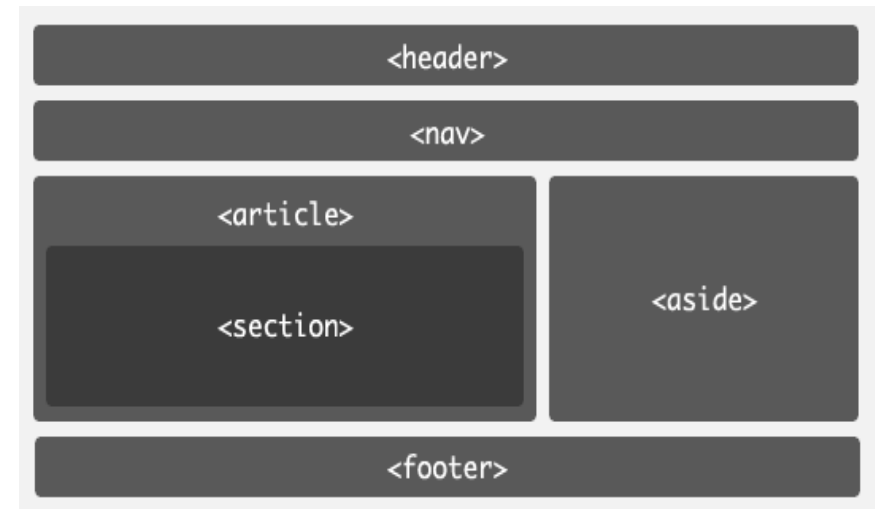
> Semantic elements for structure:

- `<header>`, `<nav>`, `<article>`, `<section>`, `<aside>`, `<footer>`

HTML4



HTML5

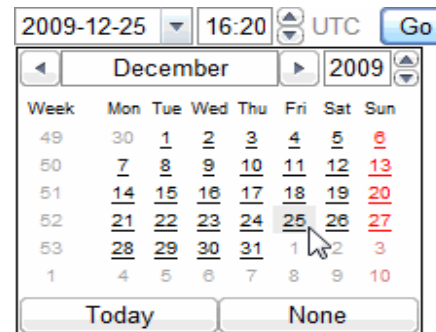


> Some advantages:

- Nested sections with header levels beyond 6 levels in HTML 4
- Cleaner source; easier to author (don't need to go "div crazy")

HTML 5 Features : Html Elements/Attributes

- > Other semantic elements:
 - <figure>, <dialog>, <meter>, <progress>...
 - <progress> can be used with JavaScript to generate “real-time” progress bar
- > Form elements / attributes:
 - <input name=”q” placeholder=”Search Here”>
 - <input name=”q” autofocus>
 - Attribute values for <input type=
 - ➔ **email** (Great for mobile devices – iphone!). **number**, **range**, **date**, **datetime**, **month**, **week**, **time**, **search**, **color**
- For date / time Opera would render:



HTML 5 Features : Media Elements

- > Audio
 - Most audio played through flash plugin
 - Not all browsers have same plugins
- > <audio> element:
 - Standard way to include audio: sound files or audio stream
 - 3 supported formats: Ogg Vorbis, MP3, Wav (browsers may support subset)
- > Usage example:

```
<audio controls="controls">  
  <source src="song.ogg" type="audio/ogg">  
  <source src="song.mp3" type="audio/mpeg">  
  Your browser does not support the audio element.  
</audio>
```

- > Specify multiple audio file formats: browser will use first recognized format
- > “controls” attribute: adds “play”, “pause”, and “volume” controls

HTML 5 Features : Media Elements

- > Video
 - Most video played through flash plugin
 - Not all browsers have same plugins
- > <video> element:
 - Standard way to include video
 - 2 supported formats: Ogg Vorbis, MPEG4 (browsers may support subset)
- > Usage example:

```
<video width="320" height="240" controls="controls">  
  <source src="movie.ogg" type="video/ogg">  
  <source src="movie.mp4" type="video/mpeg">  
  Your browser does not support the video element.  
</video>
```
- > Specify multiple video file formats: browser will use first recognized format
- > “controls” attribute: adds “play”, “pause”, and “volume” controls

HTML 5 Features : Graphic Elements

> Canvas

- A container for graphics – use JavaScript to paint the graphics
- Use *height* and *width* attributes (pixels) for canvas dimensions
- Example:

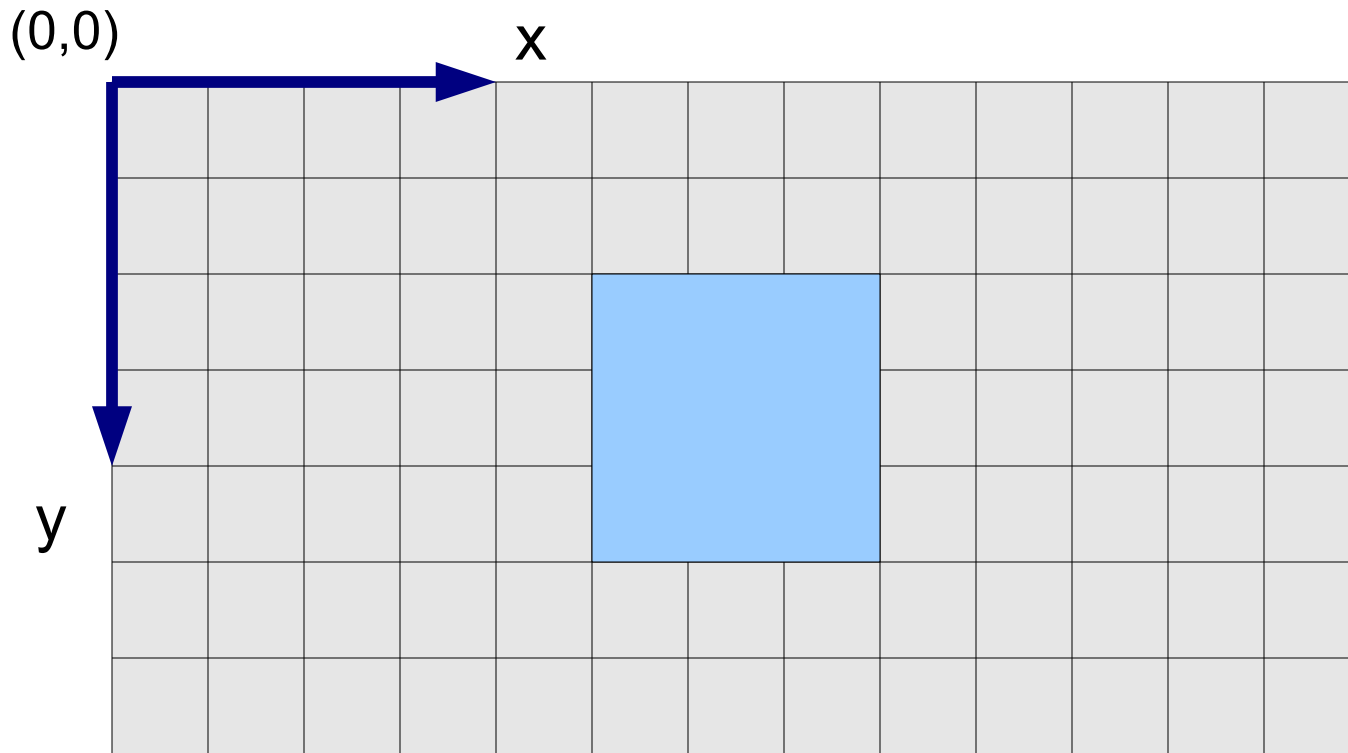
```
<canvas id="aCanvas" height="80" width="100"> </canvas>
```

```
<script type="text/javascript">  
  var canvas=document.getElementById('aCanvas');  
  var context=canvas.getContext('2d');  
  context.fillStyle='#FF0000';  
  context.fillRect(0,0,80,100);  
</script>
```

HTML 5 Features : Graphic Elements: Canvas

- > Standard graphics coordinate system

```
ctx.fillRect(5,2,3,3);
```



HTML 5 Features : Event Attributes

- > Attach JavaScript to new event types:
 - Mouse events:
 - ➔ ondrag, ondragend, ondragenter, ondragleave, ondragover, ondragstart, ondrop, onmousewheel, onscroll
 - Window events:
 - ➔ onafterprint, onbeforeprint, onbeforeunload, onerror, onhaschanged, onmessage, onoffline, ononline, ...
 - Form events:
 - ➔ onformchange, onforminput, oninput, oninvalid, ...
 - Media events:
 - ➔ Apply to media elements such as <audio>, <video>

HTML 5 Features : JavaScript API

- > Web Workers:
 - Separate JS processes running in separate threads
 - Execute concurrently; don't block UI
 - Message passing for coordination
 - High start-up performance cost; high memory cost

```
var worker = new Worker('worker.js');  
worker.onmessage = function(event) {alert(event.data);};  
  
worker.js:  
  postMessage(data);
```

- Delegation:
 - ➔ Split expensive tasks among multiple workers

HTML 5 Features : JavaScript API

> GeoLocation

- JavaScript access to the browser's location
- New property on global *navigator* object:: navigator.geolocation

```
function get_location() {  
  If (Modernizr.geolocation) {  
    navigator.geolocation.getCurrentPosition(show_map);  
  } else // no support...  
  ..  
}  
  
function show_map(position) {  
  var latitude = position.coords.latitude;  
  var longitude = position.coords.longitude;  
  // do something interesting – show map for example  
}
```

HTML 5 Features : JavaScript API

- > Audio/Video manipulation:
 - Dynamically create <audio>, <video>
 - Add custom controls to <audio>, <video>
 - Control <audio>, <video> attributes

```
var video = document.createElement('video');  
video.src = 'video.ogv';  
video.controls = true;  
document.body.appendChild(video);
```

HTML 5 Features : JavaScript API

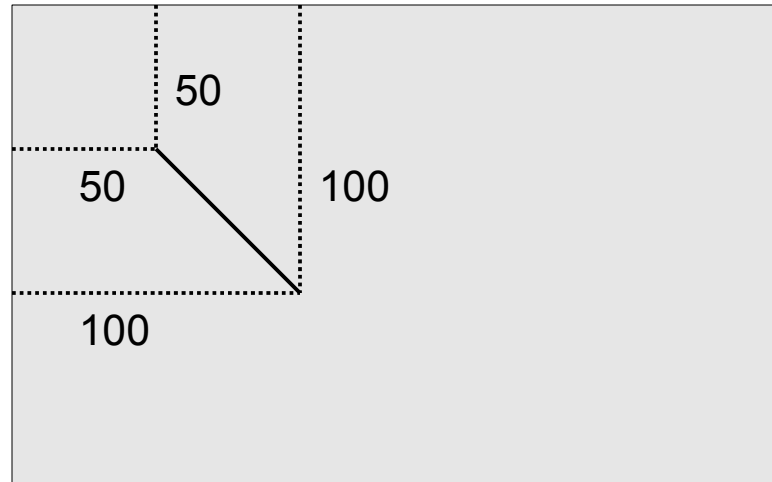
- > Canvas:
 - JavaScript to enable drawing/animation in the browser

```
<canvas id="example" width="200" height="200">
    ...
</canvas>
    ...
var example = document.getElementById('example');
var context = example.getContext('2d');
context.fillStyle = "rgb(255,0,0)";
context.fillRect(30, 30, 50, 50);
```

HTML 5 Features : JavaScript API

- > Canvas:
 - Functions for simple shapes:
 - ➔ `fillRect(x,y,w,h)` Draws rectangle
 - ➔ `strokeRect(x,y,w,h)` Draws outline of rectangle
 - ➔ `clearRect(x,y,w,h)` Clears pixels within given rectangle
 - Functions for complex shapes, paths

```
ctx.strokeStyle = "rgb(65, 60, 50)";  
ctx.beginPath();  
ctx.moveTo(50, 50);  
ctx.lineTo(100,100);  
ctx.stroke();
```



HTML 5 Features : JavaScript API

> Web Sockets:

- Provide bi-directional communication channel in the browser
- send() : Send data from browser to server
- onmessage event handler: process data from server
- Separate specification (from HTML 5)
- Many server implementations: Grizzly, GlassFish 3.1, jWebSocket, Kaazing,...

```
var ws = new WebSocket("ws://www.websocket.org");
ws.onopen = function(evt) { alert("Connection open ..."); };
ws.send(data);
ws.onmessage = function(evt) { alert( "Msg: " + evt.data); };
ws.onclose = function(evt) { alert("Connection closed."); };
ws.disconnect();
```

HTML 5 Features : What's Available .. And Where?

- > <http://html5test.com/>
 - Will tell you what HTML5 features are available for the current browser.
 - <http://weblogs.java.net/blog/rogerk/archive/2010/05/25/testing-html5-feature-availability-browsers>

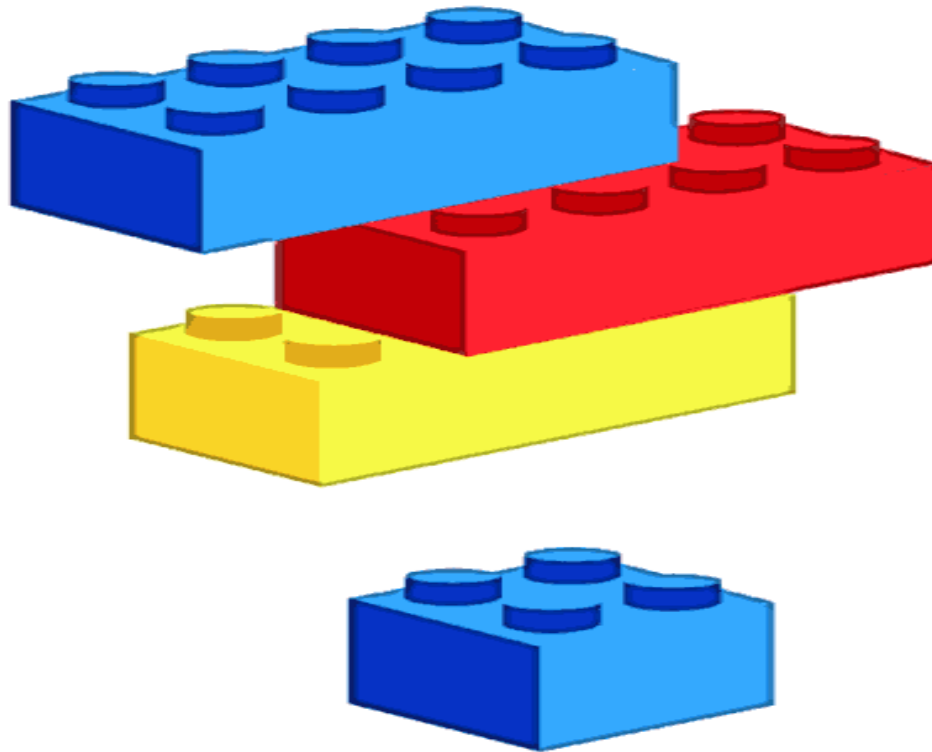
JSF 2.0 Component Model



JSF 2.0 Component Model

- > Facelets is the foundation
 - Optimized for JSF
 - XHTML and tags
 - Eliminates translation/compilation
 - Templating
- > Powerful tools:
 - Templating
 - Composite Components

JSF 2.0 Composite Components



JSF 2.0 Composite Components

- > True abstraction:
 - Reusable component
- > Turns page markup into a JSF UI component with attached validators, converters, listeners

Using Page (XHTML)

```
<html ...  
xmlns:my="http....">  
  <my:comp  
    value="yes" />  
</html>
```



Component
(XHTML)

JSF 2.0 Composite Components

```
<html xmlns="http://www.w3.org/1999/xhtml"  
xmlns:h="http://java.sun.com/jsf/html"  
xmlns:f="http://java.sun.com/jsf/core"  
xmlns:my="http://java.sun.com/jsf/composite/comp">
```

`<my:out value="yes"/>`

On disk:

`<context root>/resources/comp/out.xhtml`

JSF 2.0 Composite Components

What's Inside The Black Box?



- > Interface
 - The usage contract
 - Everything page author needs to know to use component
- > Implementation
 - Markup used to create component
 - How the component is implemented

JSF 2.0 Composite Components

```
<context-root>resources/ezcomp/LoginPanel.xhtml
```

```
<html... xmlns:ui="http://java.sun.com/jsf/facelets"
  xmlns:cc="http://java.sun.com/jsf/composite">
...
<h:body>
  <cc:interface>
    <cc:attribute name="userVal" required="true" />
    <cc:attribute name="passValue" required="true" />
    <cc:actionSource name="loginAction" targets="loginButton" />
  </cc:interface>

  <cc:implementation>
    <div> Username:<h:inputText id="userId" value="#{cc.attrs.userVal}"/> </div>
    <div> Password:<h:inputSecret id="passId" value="#{cc.attrs.passVal}"/></div>
    <div> <h:commandButton value="Login" id="loginButton" /> </div>
  </cc:implementation>
...
</h:body>
```

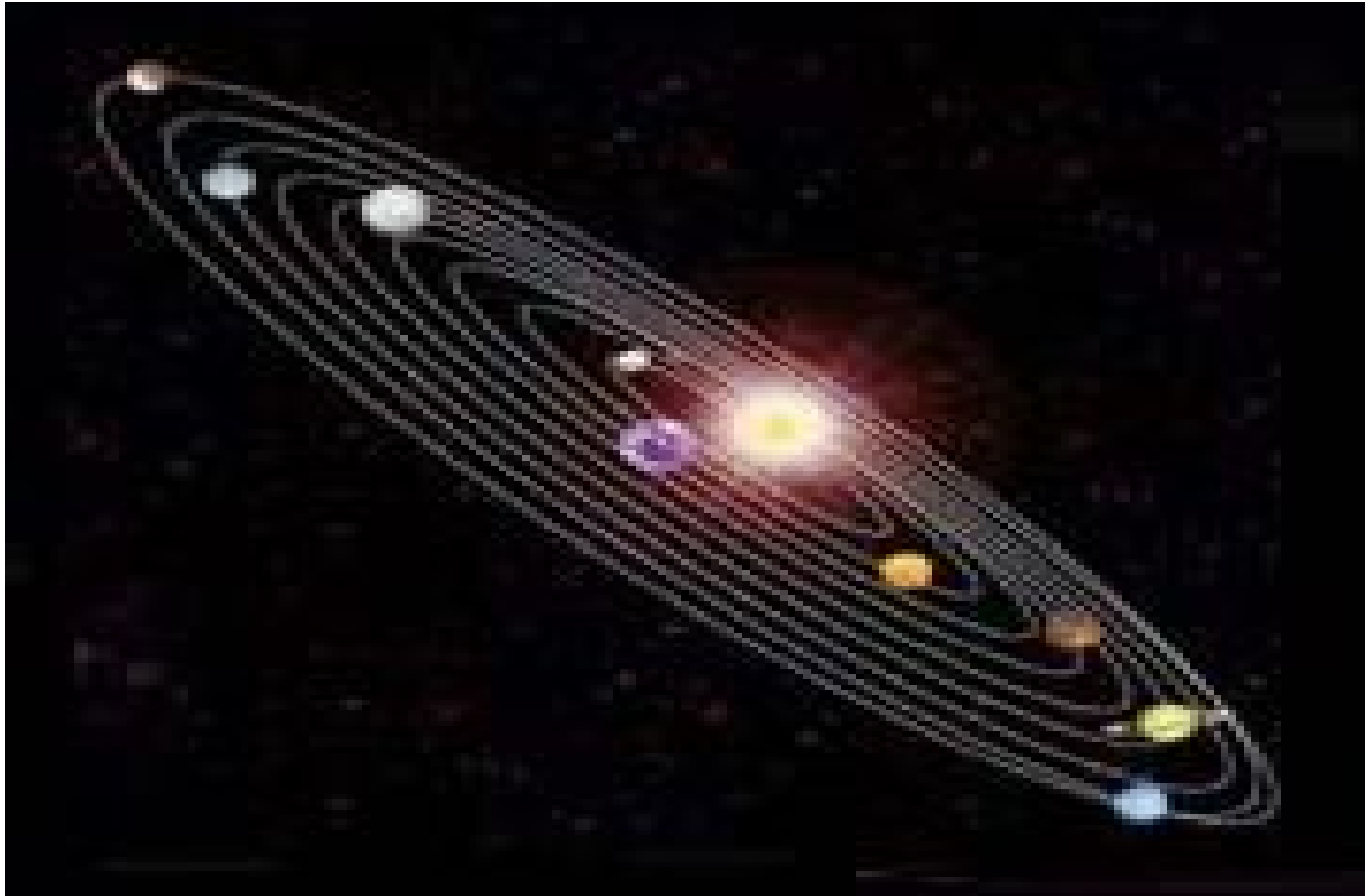
JSF 2.0 Composite Components

“Using” Page

```
<html...xmlns:ui="http://java.sun.com/jsf/facelets"
  xmlns:ez="http://java.sun.com/jsf/composite/ezcomp">
...
<h:form>
  <div id="compositeComponent" class="grayBox"
    style="border: 1px solid #090;">
    <ez:loginPanel >
      <f:actionListener for="loginAction" binding="#{bean.action}" />
    </ez:loginPanel>
  </div>
  <p><h:commandButton value="reload" /></p>
</h:form>
```

27

JSF 2.0 Composite Components With HTML 5



28

JAZOON

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY
1-3 JUNE 2010, ZÜRICH

ORACLE®

ORACLE® netcetera

Enhancing JSF 2.0 Components With HTML 5

- > JSF 2.0 specification introduced JavaScript to promote Ajax
- > Composite components work well with JavaScript
- > Composite components can leverage the HTML 5 JavaScript API

Enhancing JSF 2.0 Components With HTML 5

```
<html... xmlns:ui="http://java.sun.com/jsf/facelets" xmlns:h="http://java.sun.com/jsf/html"
  xmlns:f="http://java.sun.com/jsf/core" xmlns:cc="http://java.sun.com/jsf/composite">
<h:head> ... </h:head> ...
<h:body>
  <cc:interface>
    <cc:attribute name="src" required="true" />
    <cc:attribute name="controls" required="false" />
  </cc:interface>
  <cc:implementation>
    <h:outputScript library="js" name="audio.js" target="head"/>
    <audio src="#{cc.attrs.src}" controls="#{cc.attrs.controls}"></audio>
    <input type="button" value="Play" onclick="play()"/>
    <input type="button" value="Pause" onclick="pause()"/>
  </cc:implementation>
</h:body>
```

Enhancing JSF 2.0 Components With HTML 5

audio.js:

```
function play() {  
    var audio = document.getElementsByTagName("audio")[0];  
    audio.play();  
    var display = document.getElementsByTagName("input")[0];  
    display.value = audio.src;  
}  
  
function pause() {  
    var audio = document.getElementsByTagName("audio")[0];  
    audio.pause();  
}
```

Enhancing JSF 2.0 Components With HTML 5

“Using” Page

```
<html... xmlns:ui="http://java.sun.com/jsf/facelets"  
  xmlns:h="http://java.sun.com/jsf/html"  
  xmlns:f="http://java.sun.com/jsf/core"  
  xmlns:h5="http://java.sun.com/jsf/composite/media">  
<h:head>  
...  
</h:head>  
...  
<h:body>  
  ...  
  <h5:audiobox src="resources/media/Lightson.ogg" controls="controls"/>  
</h:body>  
....
```

DEMOS

And let's look at some code...



What's Next?

- > With respect to JSF:
 - JSF 2.0 Rev A (Minor Maintenance Release)
 - JSF 2.1 (Major Maintenance Release)
- > We would like to hear from you!

Summary

- > HTML 5
 - Really about markup **and** JavaScript API
- > HTML 5 Features
 - Promote Rich User Interfaces
 - Graphics
 - Media
 - Multiprocessing
 - Communication
- > JSF 2.0 Components work well with JavaScript
 - Leverage HTML 5 JavaScript APIs
- > Future Directions

Resources

- > <http://glassfish.dev.java.net>
- > <http://javaserverfaces.dev.java.net>
- > <http://dev.w3.org/html5/spec/Overview.html>
- > <http://dev.w3.org/html5/websockets>
- > <http://grizzly.dev.java.net>

Roger Kitain
Oracle Corporation

<http://twitter.com/rogerk09>
<http://www.java.net/blogs/rogerk>
roger.kitain@oracle.com

JAZOON

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY
1-3 JUNE 2010, ZURICH

ORACLE®

ORACLE®

netcetera