



Aplicaciones RIA con ZK

- Sesión 9: Acciones en el cliente y CSS



Índice

- Acciones en el cliente
- Modificación con CSS de componentes individuales
- Clases CSS de ZK
- Modificación de las clases CSS de ZK



Acciones en el cliente

- Acciones CSA para realizar animaciones (ZK 3.6.4)
- Integración con JQuery (ZK 5.0)
- Client computing (ZK 5.0)



Acciones CSA

- CSA: Client Side Actions
- Permiten asociar animaciones y apariciones de componentes con eventos de JavaScript
- Los componentes se localizan con etiquetas distintas de EL

```
<button action="onmouseover: action.show("#{id1}")" />
```

- Posibles acciones:
 - action.show
 - action.hide



Ejemplo

| Acciones en el cliente | | |
|------------------------|----------------------|-------------------------|
| text1: | <input type="text"/> | |
| text2: | <input type="text"/> | This is help for text2. |

```
<row>
  <label value="text1: " />
  <textbox
    action="onfocus: action.show("#{help1}); onblur: action.hide("#{help1})" />
  <label id="help1" visible="false" value="This is help for text1." />
</row>
<row>
  <label value="text2: " />
  <textbox
    action="onfocus: action.show("#{help2}); onblur: action.hide("#{help2})" />
  <label id="help2" visible="false" value="This is help for text2." />
</row>
```



Animaciones

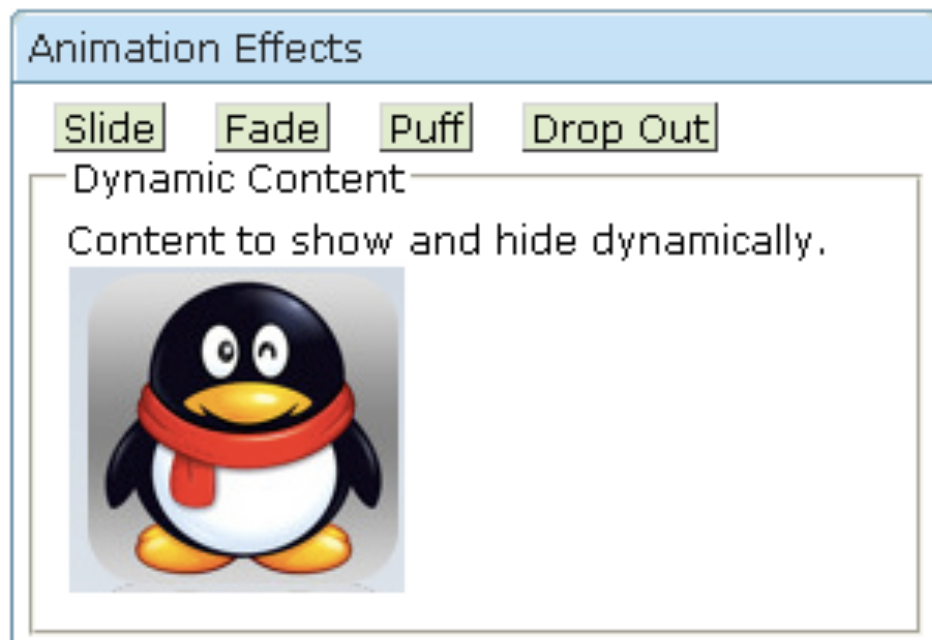
- Otro tipo de acciones: animaciones
- Asociadas a componentes div
- Apariciones
 - `anima.appear()`
 - `anima.slideDown`
- Desapariciones
 - `anima.slideUp()`
 - `anima.fade()`
 - `anima.appear()`
 - `anima.puff()`
 - `anima.dropOut()`



Ejemplo

```
...
<label value="Fade" sclass="ctl"
  action="onmouseover:anima.appear({t});
  onmouseout:anima.fade({t})" />
<label value="Puff" sclass="ctl"
  action="onmouseover:anima.appear({t});
  onmouseout:anima.puff({t})" />
...

<div id="t" visible="false">
  <div>
    <groupbox>
      <caption label="Dynamic Content" />
      Content to show and hide dynamically.
      <image src="imagenes/pinguino.png" />
    </groupbox>
  </div>
</div>
```





CSS en ZK

- ZK define clases CSS para cada componente
- Ejemplo: textbox (comprobarlo en Firebug)

```
.z-textbox {
    background: #FFF url(${c:encodeURIComponent('~./zul/img/grid/text-bg.gif')})
repeat-x 0 0;
    border: 1px solid #7F9DB9;
    font-family: ${fontFamilyC};
    font-size: ${fontSizeM};
    font-weight: normal;
}
.z-textbox-readonly, .z-textbox-text-disd {
    background: #ECEAE4;
}
```


Modificar el CSS de un componente

- Etiqueta style:

Hello, World!

```
<label value="Hello, World!"  
      style="border: 1px outset;background-color:yellow;padding:5px" />
```

- Etiqueta sclass:

OK

```
<window title="CSS Class">  
<style>  
.blue {  
  color: white; background-color: blue;  
}  
</style>  
<button label="OK" sclass="blue" />  
</window>
```

Modificando las clases CSS de ZK

- Hay que reescribir las clases que nos interesen modificar
- Button:

Events:

| CSS\Action | Normal (Open) | Hover | Click, Select, and Drag. | Focus | Focus and Hover | Disable |
|------------|---------------|-------|--------------------------|--------|-----------------|---------|
| Naming: | .z-button | -over | -clk | -focus | | -disd |
| Supported: | V | V | V | V | V | ! |

CSS Specification:

| Class Name | Description | Default Values |
|---|---------------------|--|
| .z-button | Font size and color | font-family: \${fontFamilyT} ; font-size: \${fontSizeM}; color: black; |
| .z-button .z-button-tl, .z-button .z-button-tr, .z-button .z-button-bl, .z-button .z-button-br | Corner graph | background-image:url(\${c:encodeURIComponent('~./zul/img/button/z-btn-trendy-corner.gif')}); |
| .z-button .z-button-tm, .z-button .z-button-bm | Top and bottom edge | background-image:url(\${c:encodeURIComponent('~./zul/img/button/z-btn-trendy-x.gif')}); |
| .z-button .z-button-cl, .z-button .z-button-cr | Right and left edge | background-image:url(\${c:encodeURIComponent('~./zul/img/button/z-btn-trendy-y.gif')}); |
| .z-button .z-button-cm | Center image | background-image:url(\${c:encodeURIComponent('~./zul/img/button/z-btn-trendy-ctr.gif')}); |



Código

```
<zk>
<style>
.z-button {
    color: red;
}
.z-button-over {
    color: blue;
}
.z-button-clk {
    color: black;
}</style>
<button label="click" onClick="" />
</zk>
```