Simple Web Runtime Design Patterns Using jQuery

Simple Web Runtime Design Patterns Using jQuery

Design patterns are “known solutions to common design problems”. The following design patterns make use of the jQuery JavaScript™ library and are intended for designers and developers working with Symbian Web Runtime and the S60 mobile browser.

“jQuery is a JavaScript™ library that simplifies XHTML document traversing, event handling, animating, and Ajax interactions for rapid web development.” – jQuery Website

In recent years, the free and open source jQuery has become extremely popular with visual and interaction designers as a means of easily adding interactivity to designs using techniques they are already familiar with. It allows designers to identify elements from the DOM using CSS selectors, then add unobtrusive JavaScript™ to perform Ajax calls and create simple effects—all with only a few lines of code.

Tip: If you’re new to jQuery, why not spend a few minutes exploring How jQuery works and Getting Started with jQuery.

Use of jQuery on mobile

It is also fortunate that the jQuery library itself has a very small footprint (under 20kb) and performs quite well on mobile devices. This makes it ideal when working with WRT as widgets are essentially one XHTML document containing sub-views, combined with interactivity created through manipulation of the DOM using JavaScript™. Simple, built-in effects such as show/hide, fade and slide all provide a great foundation for creating widgets with a rich user experience.

Note: Support for jQuery will vary amongst 3rd Edition devices.

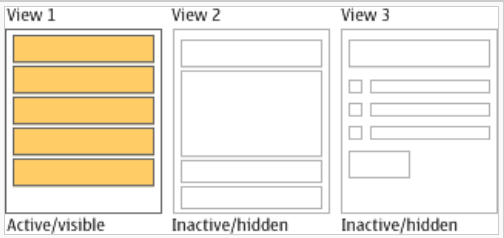
1. Show/Hide design pattern

Problem summary A user must interact with multiple views of an application yet only one view can be visible on-screen at a time.

Use

- To switch views within the widget.
- To show and hide elements within a view.
- Within a tabbed interface; to show and hide tabbed views.

Solution Provide a user or system-initiated trigger to instantly show or hide views or elements within the widget.



This widget has three views yet only one can be active at a time

Use the jQuery `show()`, `hide()` and `toggle()` methods from jQuery Effects.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
  <script src="http://code.jquery.com/jquery-latest.js" type="text/javascript"></script>
  <script type="text/javascript">
  $(function(){
    $("#btn_list").click(function () {
       $("#list").show();
```


(C) Copyright Nokia 2014. All rights reserved.

```
  $("#detail").hide();
    });
 $("#btn_detail").click(function () {
  $("#list").hide();
  $("#detail").show();
 });
 $("#btn_toggle").click(function () {
  $("#list").toggle();
  $("#detail").toggle();
 });
 $("#list").show();
  });
  </script>
  <style type="text/css">
   div {
  height: 5em;
  display: none; }
 div#list {
  background:#f00; }
 div#detail {
  background:#0f0; }
  </style>
</head>
<body>
  <button id="btn_list">List</button>
  <button id="btn_detail">Detail</button>
  <button id="btn_toggle">Toggle</button>
  <div id="list">
   <h1>List View</h1>
  </div>

  <div id="detail">
   <h1>Detail View</h1>
  </div>
</body>
</html>
```

WRT Widget for the example code can be found from File: File:Toggle JQuery.zip

2. Slide transition

Problem summary Abrupt changes in view or state are uncomfortable and often disorienting to the user.

Use

- To gradually hide an item within a view.

Solution Create a connection between the original and final view (or state) through use of an animated transition.

The slide transition is used to gradually close the content area

Use the `slideUp()` and `slideDown()` methods from jQuery Effects.

(C) Copyright Nokia 2014. All rights reserved.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
  <script src="http://code.jquery.com/jquery-latest.js" type="text/javascript"></script>
  <script type="text/javascript">
  $(function(){
    $("#btn_slide").click(function () {
       $("#slide").slideUp();
    });
    $("#btn_reset").click(function () {
  $("#slide").show();
    });
  });
  </script>
  <style type="text/css">
   div {
  height: 5em;
  background: #f00; }
  </style>
</head>
<body>
  <button id="btn_slide">Slide</button>
  <button id="btn_reset">Reset</button>
  <div id="slide"></div>
  <h1>Slide</h1>
</body>
</html>
```

WRT Widget for the example code can be found from File: File:Slide JQuery.zip

Notes: The 'Slide' features of 'jQuery UI' (an additional jQuery library) may be too resource intensive for use on mobile devices. If used, effects will appear quite sluggish—if they appear at all. Instead, use the standard 'SlideUp/SlideDown' from jQuery itself if possible.

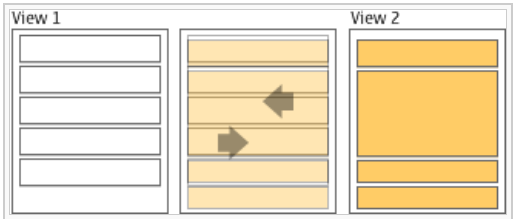
3. Fade in/out

Problem summary Abrupt changes in view are uncomfortable and often disorienting to the user.

Use

- When transitioning between widget views.
- To attract attention to, yet smooth out the loading of new data, images or other media.

Solution Incorporate a transition to gradually fade the visibility of an element into or out of view.



The transition is used to gradually fade out View 1 and fade in View 2

HACK Use WRT's "fade" transition to simulate the `fadeIn()` and `fadeOut()` methods found within jQuery.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
                    "http://www.w3.org/TR/html4/loose.dtd">
```


(C) Copyright Nokia 2014. All rights reserved.

```
<html>
<head>
<title></title>
  <script src="http://code.jquery.com/jquery-latest.js" type="text/javascript"></script>
  <script type="text/javascript">
  $(function(){
    $("#btn_fade").click(function () {
   widget.prepareForTransition("fade");
   $("#list").toggle();
   $("#detail").toggle();
   widget.performTransition();
    });
    $("#list").show();
  });
  </script>
  <style type="text/css">
  div {
   height: 5em;
  display: none; }
 div#list {
  height: 3em;
  background:#f00; }
 div#detail {
  background:#0f0; }
  </style>
</head>
<body>
  <button id="btn_fade">Fade Transition</button>
  <div id="list">
   <h1>List View</h1>
  </div>
  <div id="detail">
   <h1>Detail View</h1>
  </div>
</body>
</html>
```

WRT Widget for the example code can be found from File: File:FadeOut JQuery.zip

Notes: As WRT currently does not support opacity through CSS, the standard `fadeIn()` and `fadeout()` methods in jQuery do not work as expected. The solution above uses the transition features of WRT to perform a similar effect—in this case simply using the jQuery `show()` method.

4. Expand/Collapse

Problem summary Users require access to content or functionality but they only need it under certain circumstances and/or there is insufficient space to provide all of it at all times.

Use

- To reveal or obscure additional content associated with an element.
- Within a messaging application, to temporarily collapse non-critical information such as sender address, date, persons Ccd. etc.
- When needing to view content (i.e. text, images) that is currently represented in a thumbnail view.
- When making content editable. Ex. Expand a text field to allow for more visible content while editing.

Solution Hide a portion of this information then provide the user with a means to trigger the reveal of this information when required. This action should be easily discoverable, easy to trigger and the reveal should (ideally) occur in context of the original data.

(C) Copyright Nokia 2014. All rights reserved.

The area is expanded to reveal additional content

Use the `overflow: hidden` CSS attribute and then the `animate()` method within jQuery to increase or decrease the element's `width` and `height` attribute values over the duration specified (in milliseconds).

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
                    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
  <script src="http://code.jquery.com/jquery-latest.js" type="text/javascript"></script>
  <script type="text/javascript">
  $(function(){
    $("#btn_expand").click(function(){
      $("#item").animate({height: "15em"}, 1000 );
    });
    $("#btn_collapse").click(function(){
      $("#item").animate({height: "2.2em"}, 1000 );
    });
  });
  </script>
  <style type="text/css">
   p, h1 {
    padding: 0; margin: 0; }
 div#item {
  height: 2.2em;
  background:#f00;
  overflow: hidden; }
 div#more {
  background:#0f0; }
  </style>
</head>
<body>
  <button id="btn_expand">Expand</button>
  <button id="btn_collapse">Collapse</button>
  <div id="item">
   <h1>Header</h1>
   <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Maecenas lorem. Duis pellentesque. Cras nec tortor vel lacus fermentum facilisis. Nam id
pede eu eros posuere cursus. Mauris dolor ligula, eleifend vitae, aliquam sit amet,
bibendum vel, est. Cras sit amet augue sit amet ligula cursus scelerisque.</p>
  </div>
  <div id="more">
   <h1>More...</h1>
  </div>
</body>
</html>
```

WRT Widget for the example code can be found from File: File:ExpandCollapse JQuery.zip

Notes: Another way to create expanding and collapsing elements is to hide and show sub-elements causing the display block of

(C) Copyright Nokia 2014. All rights reserved.

the container block to expand and contract as necessary. This method does not typically incorporate animation as a means of transition.

(C) Copyright Nokia 2014. All rights reserved.