

Silverlight Tips, Tricks, and Best Practices



Take Performance Tips to Heart

- "Performance Tips for Silverlight-Based Applications"
 - Use Visibility rather than Opacity to hide objects
 - Don't use Width and Height on MediaElement
 - Don't use Width and Height on Path objects
 - Use transparent control backgrounds sparingly
 - Avoid long-running JavaScript tasks and more

<http://msdn2.microsoft.com/en-us/library/bb693295.aspx>

Detach Event Handlers

- Detach programmatically registered event handlers after use to avoid memory leaks

```
var downloader = _host.createObject('downloader');
_token = downloader.addEventListener('completed',
    DownloadCompleted);
downloader.open('GET', 'MyVideo.wmv');
downloader.send();
.
.
function DownloadCompleted(sender, args)
{
    _video.setSource(sender, '');
    sender.removeEventListener('completed', _token);
}
```

Use FindName Sparingly

- Avoid redundant calls to FindName

```
// Don't do this
function onClick(sender, args)
{
    var rect = sender.findName('Rect');
    rect.fill = 'red';
}

// Do this
function onClick(sender, args)
{
    _rect.fill = 'red'; // _rect holds cached XAML reference
}
```

Avoid "Wait and Wonder"

- Never make the user wait and wonder
- Avoid static declarations to large assets
 - Videos, images, XAML documents, etc.
 - In Silverlight 2, includes assemblies, too
 - Important in Silverlight 2; crucial in 1.0
- Use downloader object in Silverlight 1.0, or WebClient in Silverlight 2, to download assets and provide progress UI
- Optionally package in ZIPs or XAPs

What's Wrong with this XAML?

```
<Canvas
    xmlns=http://schemas.microsoft.com/client/2007
    xmlns:x=http://schemas.microsoft.com/winfx/2006/xaml>
    <MediaElement x:Name="Player" Source="Videos/Flight.wmv" />
</Canvas>
```

Downloading Videos (1.0)

```
var downloader = _host.createObject('downloader');
_token1 = downloader.addEventListener('downloadProgressChanged',
    DownloadProgressChanged);
_token2 = downloader.addEventListener('completed', DownloadCompleted);
downloader.open('GET', 'Flight.wmv');
downloader.send();
...
function DownloadProgressChanged(sender, args)
{
    var progress = sender.downloadProgress; // 0.0 to 1.0
    ...
}

function DownloadCompleted(sender, args)
{
    _player.setSource(sender, ''); // Assign bits to MediaElement
    sender.removeEventListener('downloadProgressChanged', _token1);
    sender.removeEventListener('completed', _token2);
}
```

Downloading Videos (2)

```
WebClient wc = new WebClient();
wc.OpenReadCompleted +=
    new OpenReadCompletedEventHandler(OnOpenReadCompleted);
wc.OpenReadAsync(new Uri("Assets.xap", UriKind.Relative));
...
void OnOpenReadCompleted(object sender,
    OpenReadCompletedEventArgs e)
{
    Player.SetSource(e.Result);
}
```

Downloading ZIP Video (1.0)

```
var downloader = _host.createObject('downloader');
token = downloader.addEventListener('completed',
    DownloadCompleted);
downloader.open('GET', 'Assets.zip');
downloader.send();
...
function DownloadCompleted(sender, args)
{
    _player.setSource(sender, 'Flight.wmv');
    sender.removeEventListener('completed', _token);
}
```

Downloading XAP Video (2)

```
WebClient wc = new WebClient();  
wc.OpenReadCompleted +=  
    new OpenReadCompletedEventHandler(OnOpenReadCompleted);  
wc.OpenReadAsync(new Uri("Assets.xap", UriKind.Relative));  
...  
void OnOpenReadCompleted(object sender,  
    OpenReadCompletedEventArgs e)  
{  
    StreamResourceInfo sri1 =  
        new StreamResourceInfo(e.Result, null);  
    StreamResourceInfo sri2 = Application.GetResourceStream  
        (sri1, new Uri("Flight.wmv", UriKind.Relative));  
    Player.SetSource(sri2.Stream); // MediaElement  
}
```

Downloading Assets On Demand



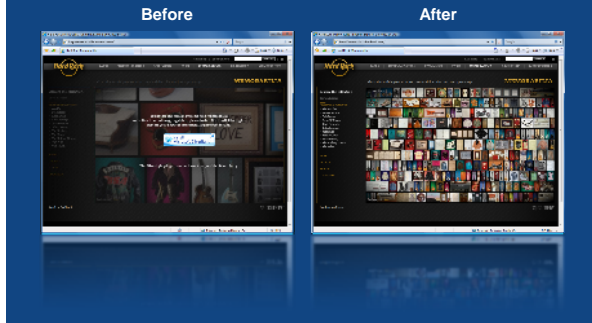
Optimize the Install Experience

- Default install experience is not compelling

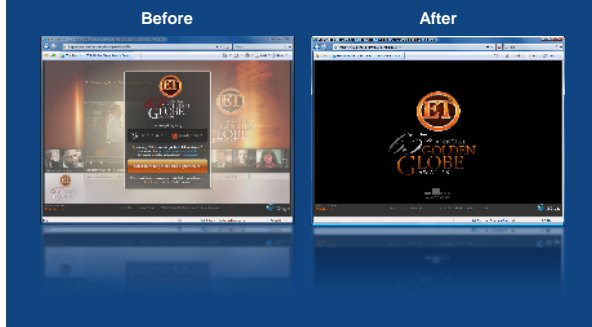


<http://www.microsoft.com/downloads/details.aspx?FamilyId=F487DF43-1AFB-4F76-82C8-BB5ACBFFBA1B&displaylang=en>

Hard Rock Memorabilia



Golden Globe Awards



Conventional Activation (1.0)

```
<div id="SilverlightPlugInHost">  
<script type="text/javascript">  
  createSilverlight();  
</script>  
</div>
```

*Instantiates Silverlight control or displays
install prompt if control not installed*

Enhanced Activation (1.0)

```
<div id="Container">
  <div id="SilverlightPlugInHost"></div>
</div>
<script type="text/javascript">
var _id;
if (!Silverlight.isInstalled('1.0'))
{
  document.getElementById('Container').className = 'AgNotInstalled';
  _id = window.setInterval('checkInstall()', 3000);
}
createsilverlight(); // Create control or display install prompt
function checkInstall()
{
  if (Silverlight.isInstalled('1.0'))
  {
    window.clearInterval(_id);
    document.getElementById('Container').className = 'AgInstalled';
    createsilverlight(); // Create control
  }
}
</script>
```

Conventional Activation (2)

```
<div id="silverlightControlHost">
  <object data="data:application/x-silverlight," ... >
    <param name="source" value="ClientBin/MyApp.xap"/>
    <param name="background" value="white" />
    <a href=http://go.microsoft.com/fwlink/?LinkId=108182
      style="text-decoration: none;">
      <img
        src=http://go.microsoft.com/fwlink/?LinkId=108181
        alt="Get Microsoft Silverlight" ... />
      </a>
    </object>
  </div>
```

Enhanced Activation (2)

```
<div id="silverlightControlHost">
  <object data="data:application/x-silverlight," ... >
    <param name="source" value="ClientBin/MyApp.xap"/>
    <param name="background" value="white" />
    <div class="install">
      <a href=http://go.microsoft.com/fwlink/?LinkId=108182
        style="text-decoration: none;">
        <img
          src=http://go.microsoft.com/fwlink/?LinkId=108181
          alt="Get Microsoft Silverlight" ... />
        </a>
      <div>
    </object>
  </div>
```

Customizing the Install Experience



Timers and Animations

- Use timers to drive animations that can't be performed declaratively
- Silverlight 1.0: Storyboard.Completed
 - Set Duration to desired timer interval
 - Call Storyboard.Begin to start timer
 - Process Completed event and call Begin again
- Silverlight 2: DispatcherTimer
 - Fires Tick events at specified intervals
 - Event handler executes on UI thread

Storyboard.Completed (1.0)

```
<!-- XAML -->
<Storyboard x:Name="TimerStoryboard" Duration="0:0:0.2"
  Completed="onTick" />

// JavaScript
function onTick(sender, args)
{
  // TODO: Update scene

  // Restart the timer
  _timer.begin(); // _timer refers to Storyboard
}
```

DispatcherTimer (2)

```
private DispatcherTimer _timer;
...
_timer = new DispatcherTimer();
_timer.Tick += new EventHandler(OnTimerTick);
_timer.Interval = new TimeSpan(0, 0, 0, 0, 50); // 50 ms
_timer.Start();
...
protected void OnTimerTick(Object sender, EventArgs e)
{
    // TODO: Update UI (called on UI thread)
}
```

Programmatic Animations



Demo

Page Turns Made Easy

- Use my framework for page-turning apps

Function	Description
addPage	Registers a page pair with the framework
dispose	Releases resources held by the framework
getCurrentPageIndex	Returns the 0-based index of the page pair displayed
getCount	Returns the number of page pairs added with addPage
goToPage	Displays the specified page pair
initializeFramework	Initializes the page-turn framework

<http://msdn.microsoft.com/en-us/magazine/cc507644.aspx>

XAML Structure

```
<Canvas
  xmlns="http://schemas.microsoft.com/client/2007"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  Background="color" Width="width" Height="height">
  ...
  <Canvas x:Name="PageTurnCanvas">
    <!-- First page pair -->
    <Canvas x:Name="EvenPage0">
      <!-- Content for left-hand page goes here -->
    </Canvas>
    <Canvas x:Name="OddPage0">
      <!-- Content for right-hand page goes here -->
    </Canvas>

    <!-- Additional page pairs go here -->

  </Canvas>
  ...
</Canvas>
```

Using PageTurnFramework

```
// Instantiate the framework
_ptf = new PageTurnFramework(_control,
  _control.content.findName('PageTurnCanvas'));

// Add pages
_ptf.addPage(_control.content.findName('EvenPage0'),
  _control.content.findName('OddPage0'));
.
.
.
// Initialize the framework
_ptf.initializeFramework();
```

Page-Turn Framework

Demo

Can We Talk?

- Pages can host multiple Silverlight controls
- Controls can communicate through bridge
 - Use `HtmlWindow.Invoke` to call JavaScript function from source control
 - From JavaScript, call scriptable method(s) in target control
- Bridge Silverlight and Flash ("Flashlight?")
 - <http://www.vinull.com/Post/2008/04/21/silverlight-interop-with-flashflex-flas.aspx>

Calling JavaScript Functions from C#

C#

```
HtmlPage.Window.Invoke("changeColor", "Blue");
```

JavaScript

```
function changeColor(args)
{
    var color = args; // color == 'Blue'
    ...
}
```

Exposing C# Methods to JavaScript

Allows class to support scriptable members

Name used to refer to this object in JavaScript

```
[ScriptableType]
public partial class Page : UserControl
{
    public Page()
    {
        ...
        HtmlPage.RegisterScriptableObject("magic", this);
    }

    [ScriptableMember]
    public void ChangeColor(string color) { ... }
}
```

Script-callable method

Calling C# Methods from JavaScript

```
var control = document.getElementById('SilverlightControl');  
control.content.magic.ChangeColor('Red');
```

ID of Silverlight control

Registered name *Method name*

Control Interop



Interactive Zoom

- Use ScaleTransforms to implement interactive zooms
 - Mousewheel zooms
 - Magnifying glasses
- Great for increasing the wow factor
 - Vector graphics scale really well
 - Bitmaps scale well with preplanning
- Great for enhancing accessibility, too

Interactive Zoom (1.0)

```
<!-- XAML -->
<ScaleTransform x:Name="Zoom" />

// JavaScript
function zoomIn()
{
    _zoom.scaleX += 0.1;
    _zoom.scaleY += 0.1;
}

function zoomOut()
{
    _zoom.scaleX -= 0.1;
    _zoom.scaleY -= 0.1;
}
```

Interactive Zoom (2)

```
<!-- XAML -->
<ScaleTransform x:Name="Zoom" />

// C#
private void ZoomIn()
{
    Zoom.ScaleX += 0.1;
    Zoom.ScaleY += 0.1;
}

private void ZoomOut()
{
    Zoom.ScaleX -= 0.1;
    Zoom.ScaleY -= 0.1;
}
```

Interactive Zoom



Deep Zoom

- Use Deep Zoom for awesome zooms
 - Silverlight implementation of "Seadragon" technology from Microsoft Live Labs
 - Interactive scaling of bitmapped images, paying as you go for bandwidth
- Deep Zoom Composer tool and MultiScaleImage control make Deep Zoom easy

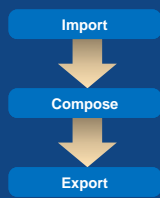
Deep Zoom Examples

- Deep Zoom Silverlight poster
 - <http://joestegman.members.winisp.net/DeepZoom/>
- Deep Zoom Hard Rock Memorabilia
 - <http://memorabilia.hardrock.com/>
- Deep Zoom Vertigo site
 - <http://www.vertigo.com/DeepZoom.aspx>



Deep Zoom Composer

- Free tool for building Deep Zoom scenes



<http://www.microsoft.com/downloads/details.aspx?familyid=457b17b7-52bf-4bda-87a3-fa8a4673f8bf&displaylang=en>

MultiScaleImage

- Control for presenting Deep Zoom content
- Consumes Deep Zoom Composer output
 - Source property points to XML file generated by Deep Zoom Composer
- Provides simple API for adding interactivity
 - Deep Zoom Composer emits code for panning and zooming
 - "Springiness" provided by control

Deep Zoom

Demo

Discussion