



Silverlight – Case Study

Author	Samiruddin Syed
Reviewer	P. Vasudeva Kiran
Date	28-JAN-2010

Document History

Version	Date	Author	Reviewed By	Changes done
1.0	02-Aug-2010	Samiruddin Syed	P. Vasudeva Kiran	

Table of Contents

- 1. Introduction 4
- 2. Client – ourFamily•ology 5
- 3. Problem – Photo Viewer 6
- 4. Solution 6
- 5. Problem – Video Viewer 7
- 6. Solution 7
- 7. Problem – File Uploader 8
- 8. Solution 8

1. Introduction

Silverlight is a framework for rich, browser-hosted applications that runs on a variety of operating systems. Silverlight works its magic through a browser plug-in. When you surf to a web page that includes some Silverlight content, this browser plug-in runs, executes the code, and renders that content in a specifically designated region of the page. The important part is that the Silverlight plug-in provides a far richer environment than the traditional blend of HTML and JavaScript that powers ordinary web pages. Used carefully and artfully, you can create Silverlight pages with interactive graphics, use vector animations, and play video and sound files.

Several other technologies use a plug-in to stretch bounds of the browser, including Java, ActiveX, Shockwave and Adobe Flash. Although all these alternatives are still in use, none of them has become a single, dominant platform for rich web development. Many of them suffer from a number of problems, including installation headaches, poor development tools, and insufficient compatibility with full range of browsers and operating systems. The only technology that's been able to avoid these pitfalls is Flash, which boasts excellent cross platform support and widespread adaptation. However, Adobe Flash only recently evolved from a multimedia player into a set of dynamic programming tools. It still offers far less than a modern programming environment like .NET.

That's where Silverlight fits into picture. Silverlight aims to combine the raw power and cross platform support of Flash with a first class programming platform that incorporates the fundamental concepts of .NET. Silverlight boasts a few architectural features that Flash can't match; most importantly the fact that it's based on a scaled-down version of .NET's Common Language Runtime (CLR) and allows developers to write client side code using pure C#.

2. Client – ourFamily•ology

ourFamily•ology is a Genealogy based Social Networking web application to build and organize a precise family history and a reliable family tree of a user that can be shared with family and friends. This unique and rich featured system contains many key components which are an added advantage from the other competitor websites.

Seanergy engineered the project, gathered requirements and made the system very robust, comprising of key features like anywhere access, Security, Collaborative research, Progress Rating Chart (unique among genealogy software), Publish Trees, Multi dimensional visual views, Organization, Support. Components such as data transferring EXIM (import and export) of GEDCOM and PAF files, external data transfer are enriched in the system. Dynamic link generator to share information with researchers and visitors are in the system along with many web services.

Mrs. Brandy Sacco,
ourFamilyology,
205 Stokes Farm Road,
Franklin Lakes, NJ 07

201-315-0228

brandy.sacco@ourfamilyology.com

www.family-genealogy.com

3. Problem – Photo Viewer

A Photo Viewer should have a full screen Slideshow and some animation when a photo changes. It should be easy even for a naïve user to switch from normal mode to full screen mode and vice versa and he/she need not make any browser specific actions to switch between modes.

This application’s design and layout should change in full screen mode and should rollback as it was when it is back in normal mode.

Slideshow of the Photos should be in full screen mode and should have animation effects similar to PowerPoint’s slideshow animation.

Design of this application should be easily configurable, so that the photo list can be place anywhere on the screen.

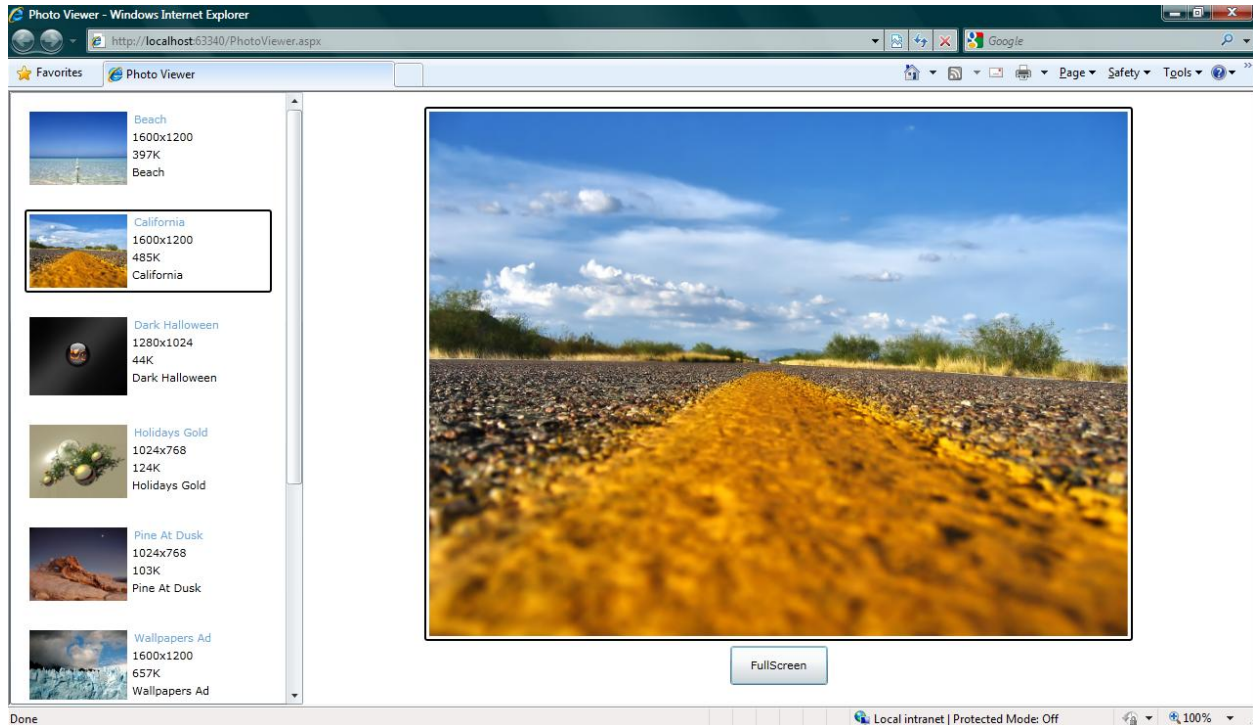
A user should be able to see previous or next photo using keyboard’s arrow keys, in normal mode as well as in full screen mode.

4. Solution

Showing any content in full screen mode using traditional HTML and JavaScript is not possible, but Silverlight enables a developer to show any content in full screen mode in a single line of code and a user has to make a simple action like clicking a button to switch it to full screen mode. And moreover an event is raised when the application switches between normal and full screen mode, so that a developer could make any necessary changes in the design or layout. And all these can be achieved independent of the browser a user might be using.

In Silverlight, any control can be rendered inside a Grid or a Canvas. Grid is similar to a HTML table where every object is places in rows and columns. In a Silverlight’s Grid, rows and columns can be created or modified dynamically and any control can be associated with any row or column.

Silverlight supports even complex interactive graphics, use vector animations. Using these two features, any type of complex animations can be created. It can be done in simple XAML or sometimes in C#/VB.NET.



5. Problem – Video Viewer

In a Video Viewer a user should be able to view videos in full screen. It should be easy even for a naïve user to switch from normal mode to full screen mode and vice versa and he/she need not make any browser specific actions to switch between modes.

This application’s design and layout should change in full screen mode and should rollback as it was when it is back in normal mode.

Quality of the video should be maintained automatically so that a user experience better video streaming regardless of user’s internet speed, as the video is not smooth for users with slow internet connections.

Design of this application should be easily configurable, so that the video list can be place anywhere on the screen.

A user should be able to see previous or next video using keyboard’s arrow keys, in normal mode as well as in full screen mode.

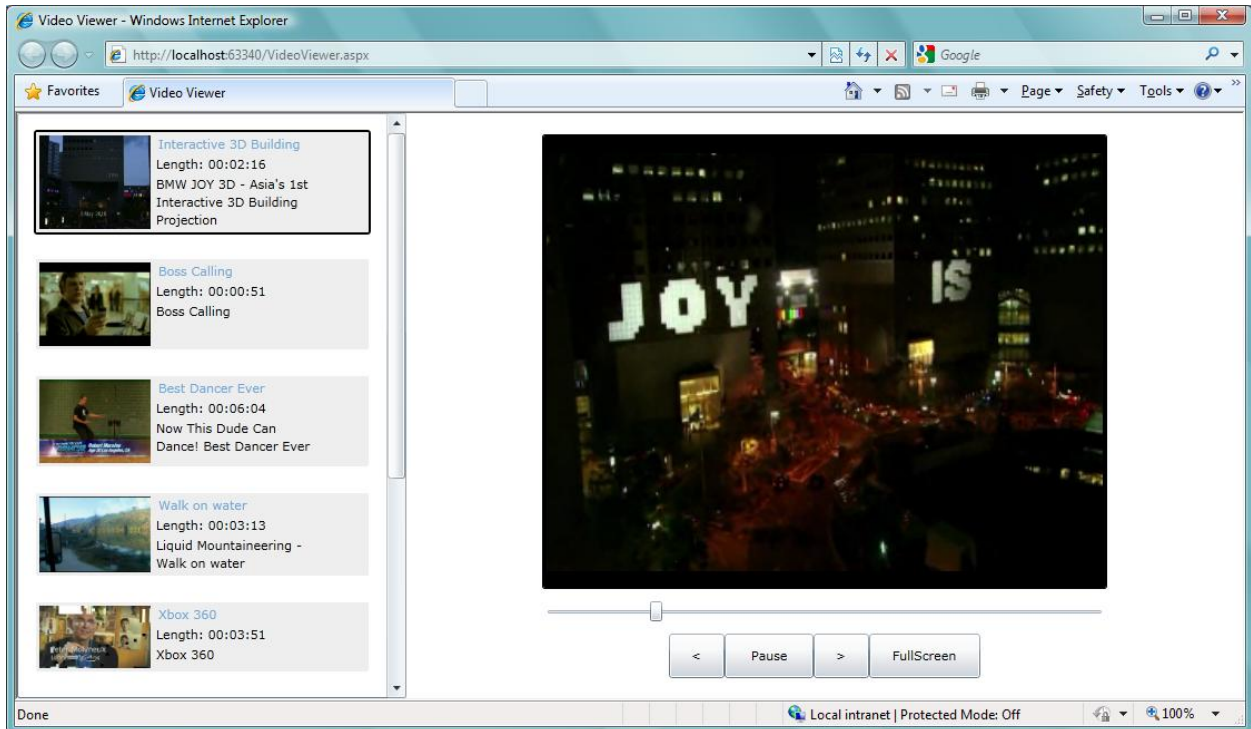
6. Solution

Showing any content in full screen mode using traditional HTML and JavaScript is not possible, but Silverlight enables a developer to show any content in full screen mode in a single line of code and a user has to make a simple action like clicking a button to switch it to full screen mode. And moreover an

event is raised when the application switches between normal and full screen mode, so that a developer could make any necessary changes in the design or layout. And all these can be achieved independent of the browser a user might be using.

In Silverlight, any control can be rendered inside a Grid or a Canvas. Grid is similar to a HTML table where every object is places in rows and columns. In a Silverlight’s Grid, rows and columns can be created or modified dynamically and any control can be associated with any row or column.

Quality of the video, buffering can be controlled so that video streaming is smooth.



7. Problem – File Uploader

Uploading multiple files simultaneously in HTML requires a user to select each and every file individually and then upload it. This is a very tedious process when a user has to upload too many files.

The application should be able to validate file information, like file type and file size on the client side itself, so that a user can experience a faster application.

8. Solution

In Silverlight, as the code is executed on the client’s system it has direct access to file being uploaded, so all the needed validations can be done on the client side almost instantly. After validation succeeds the file can be split into small chunks, serialized and transferred over the network. When the Server receives

this request it will deserialize the data, process it and return a success message if the process was successful.

Now, the Silverlight application which is hosted on the client reads the success message and updates the progress bar accordingly.

Silverlight supports multi threading, so each file can be sent to the server independently and simultaneously without conflicting with each other.

