



SpatialMedia

Emaml.NET ASP.NET Server Control for ArcIMS

If you are familiar with ASP.NET, and write GIS applications you will no doubt be delighted to learn that SpatialMedia can supply a comprehensive .NET server control for ArcIMS/ArcMap server. It does exactly what you expect and lets you build a powerful ArcIMS application by simply dragging a control onto a form and setting a couple of properties. It exposes a rich API to really work ArcIMS and delivers on the ASP.NET promise of making web development as easy as building apps out of local COM controls.

EmamlNet features a Jscript/DHTML client that requires no plugins or applets but delivers functions to the user that would normally require these addins.

If you haven't yet learned why ASP.NET is a radical departure from "old world ASP" and other web development platforms please read on.

Why is VB everywhere?

Visual Basic has always allowed business developers to build windows apps far more simply and quickly than they could have using C++. The real winner for VB however was components. An entire industry sprang up of fabulous "controls" that VB developers could drop into their applications for instant functionality. This model of development provided far more developer control than customizing existing apps, without collapsing into the "building houses from matchsticks" of lower level windows programming. Simple, fast modular.

What about GIS?

GIS companies couldn't ignore this action and products such as MapInfo's MapX and ESRI's MapObjects opened up a new world of low cost GIS applications. Applications that did exactly what users needed. Forget messy DDE connections between separate applications. Embedded GIS was real. GIS developers no longer needed to fold their application into a complete GIS solution like Arcview or MapInfo. *Map in the App* was real.

What about the web?

This is where things get complex. In our view, the potential of windows component development (using MapObjects etc) has been overshadowed by the arrival of Internet based GIS. This model provided even lower cost per seat than components with simplified deployment and greater data security. The downside was functionality. For a variety of well known reasons it has been harder to implement a given function on the web than using a component.

Does ASP.NET offer the best of both worlds?

Yes! This is what it is all about. .NET is a lot of things, including a runtime engine and new language which are not unlike Java. There are some reworked languages (VB.NET, Jscript.Net ,J#) . There is a new emphasis on "loosely coupled", web service architectures.

But the really exciting part of .NET for GIS developers is ASP.NET or rather, Server Controls.

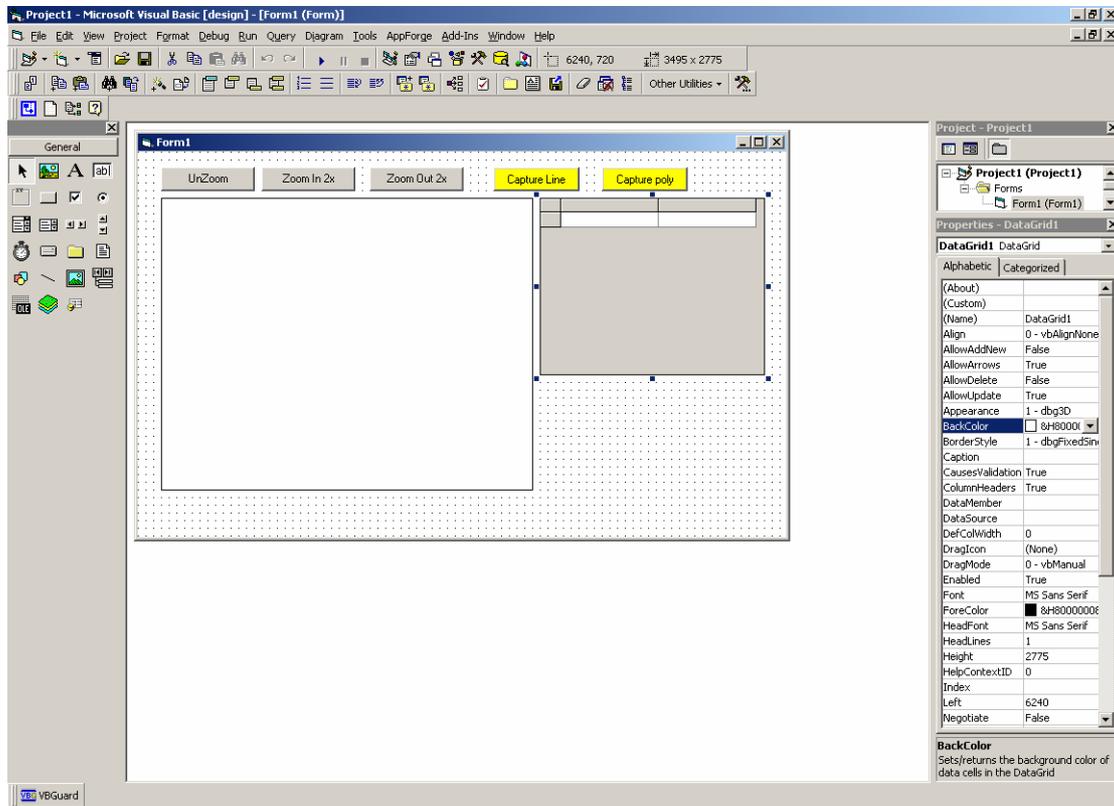
Server Controls?

Forget all those messy details like how they work. They work. The key benefits over old world ASP and similar technologies include:

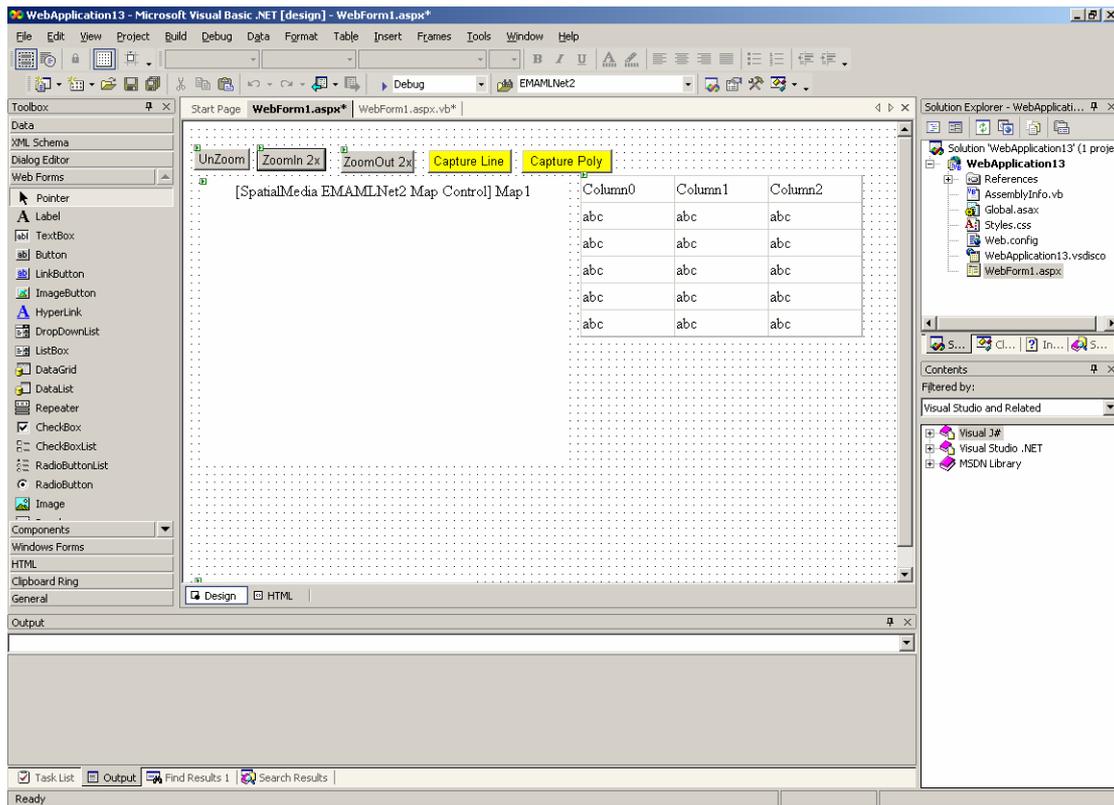
- Developers don't need any web experience to write web apps.
- Developers can focus on their application -NOT on maintaining the illusion of a stateful windows application via web forms.
- Powerful but modular functionality can be included by embedded server controls.

And let's not ignore the value of...

- Maintainable, reusable code. That mix of VBScript and HTML that worked OK for simple jobs but was unmanageable for serious jobs has gone.
- The .NET platform itself, languages, runtime, web services etc.
- A proper debugging environment. No more Response.Write "got this far OK."
- Visual Studio -probably the best IDE going.



Work in progress A. Simple Visual Basic 6 Application with a MapObjects Map Control and a Microsoft DataGrid Control.



Work in progress B. Simple ASP.NET Application with a SpatialMedia Emaml.Net control and a Microsoft DataGrid. Very similar programming experience, but this one runs on the internet.

So a server control is like a COM component such as MapObjects?

Well yes and no. Let's compare ESRI MapObjects and SpatialMedia Emaml.Net objects.

High level.

MapObjects: Embedded in a COM (eg VB6) User sees a dedicated application (eg a precision farming app as opposed to a "GIS" app) with integrated embedded, navigable maps.

EmamlNet. Drops into an ASP.NET (eg VB.NET) application and provides an intuitive API. User sees a dedicated web application with integrated embedded navigable maps.

Mid Level

MapObjects: Provides a comprehensive API (application programming interface -the various methods and properties the developer manipulates with their own code). Knows how to render itself in a windows application. Developer needn't learn windows GDI programming to draw maps.

EmamlNet: Provides an comprehensive API. Knows how to render itself as HTML and JavaScript. Developer needn't learn HTML and JavaScript to render interactive map content.

Low Level

MapObjects: Works with local data or database connections. Renders maps and can perform queries locally.

EmamlNet. Works with remote ArcIMS, ArcMap Server or map-TV map servers. Maps are rendered and queries executed by the remote server.

Rivers & Rainfall

Environment Southland
Te Taiāwhiri

Home
Contact Us
About Us
Whats New
Publications
Enviro Info
Rivers & Rainfall
Links
Site Map
Events
Vacancies
Public Notices
2003 Environment Awards

Map Tools
Identify Feature
Magnify
Zoom Out
Reset map
Find
Click on the map to:
In 2x & re-center

Site Name	Date/Time	Stage	Stage Change	Peak Stage	Peak Time	Above Normal	Peak Flow	Flow
Maitava River at Fairlight Station	23-Jul-2003 10:30:589	-3		712	20-Jul-2003 11:00:211			
Maitava River at Parawa	23-Jul-2003 10:15:256766	-1		256964	20-Jul-2003 10:15:466	24.0		15.4
Maitava River at Cattle Flat	23-Jul-2003 10:30:931	0		1078	21-Jul-2003 04:45:131			
Maitava River at Pyramid Bridge	23-Jul-2003 10:45:1071	1		1274	21-Jul-2003 08:15:171	59.5		42.6
Maitava River at Oore	23-Jul-2003 10:20:1070	-3		1247	20-Jul-2003 16:00:270	85.2		61.7

With Emaml.Net you can build an app with a map that takes up the whole page -but we do like this example implemented by Environment Southland using EMAML.NET. True map in the app.

More About EMAML.NET

Operation of Emaml.Net Objects is very similar to COM based mapping components. All you need to do to add mapping to your app is paint a map on your form and set a single property to tell the control what mapping service to connect to.

To enable client side DHTML navigation, simply add a single onload attribute to the page body to start the Emaml client, and the user will be able to drag rubber band boxes over the map and drag it around. These actions will cause events to fire in your application code.

For example, if vector based data capture is enabled the control can be set in a capture shape model (points lines or polygons). The user can then digitize a shape on the map (no trips back to the server the way). When she double clicks to finish, a MapCaptureShape event will fire and pass a Shape object. The Shape might then me added to the map's acetate layer for example or used to perform a selection.

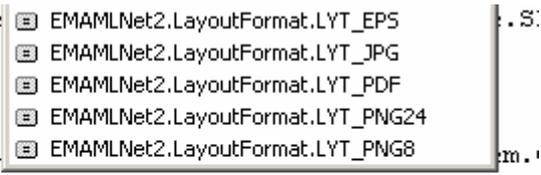
The control is designed to remove the need for any direct XML communication with the map server. All server functionality is access via the control's API. For example, you might execute a spatial query and the control would return matching feature attributes as a familiar .NET DataTable. This allows you leverage .NET functionality by sending these results, for example, to a DataGrid for powerful rendering with almost no code.

The control initializes itself using service information data sent from the server. It can therefore expose and API based on those properties (for example a collection of Layer objects).

```

labStatus.Te
End Sub
Private Sub Butt
Map1.GetLayout (
End
Class

```



Visual Studio.Net and the comprehensive Emaml.Net API make programming ArcIMS easy.

Should you require a service not supported by the control's API you can easily send direct AXL requests to the server but this kind of operation should be considered exceptional.

Beyond the Net

SpatialMedia can offer a further level of functionality that you won't find elsewhere. Emaml.Net applications can run in standalone mode, without a connection to ArcIMS or ArcMap Server.

How do we do this?

The key is that Emaml.Net can run against ESRI map servers, or our own "map-TV" server. Map-TV is lightweight and very fast, and can easily run for example on a Tablet PC. Map-TV doesn't support all the functionality of ArcIMS but it does provide the key services required to implement map browsing apps. Run Emaml.Net apps against ArcIMS/ArcMap Server on the Internet/Intranet -and run the same app in the field against map-TV.

Why is this important?

- Wireless connections are slow and expensive.
- GIS needs to go places that wireless coverage doesn't go.
- Standalone apps may be a key component of disaster recovery plans/

Finally

Forget all the benefits described above. If you write GIS apps for windows and want to take them to the net you should be writing ASP.NET. Old world ASP is simply legacy code. Trying to hack up JavaScript programs and kludgy frame based apps is a dead end. Unless your business has a commitment to an alternative, (UNIX/Java Linux/OpenSource etc) the question isn't whether you'll be writing ASP.NET but when.

Emaml.Net provides an opportunity to get into this game today, with an incredibly powerful model and level of functionality that leaves alternative ArcIMS development options in the dust.

Next Step

Contact Peter Fraser, SpatialMedia at pfraser@spatialmedia.com. If you are an experienced .Net developer we can organize an evaluation copy. If you are new to .Net we would be delighted to come to you to talk about our how SpatialMedia solutions can work for you -and optionally install Emaml.Net and provide some training.