ASP.NET

Edit page on GitHub (https://github.com/mono/website/blob/gh-pages/docs/web/aspnet.md)

Mono has an implementation of ASP.NET 2.0, ASP.NET MVC and ASP.NET AJAX.

Quick Resources:

- ASP.NET FAQ (/docs/faq/aspnet/) for common questions on ASP.NET.
- Hosting on Apache servers (/docs/web/mod_mono/).
- Hosting with FastCGI-based servers (/docs/web/fastcgi/).
- Hosting with CGI-based servers (/archived/cgi).
- Hosting with Nginx (/docs/web/fastcgi/nginx/).
- Porting ASP.NET applications (/docs/web/porting-aspnet-applications/).

Mono's ASP.NET implementations supports two kinds of applications:

- Web Forms (Web Applications infrastructure).
- Web Services (/archived/web_services) (the SOAP-based RPC system).

Status and tests for ASP.NET 2.0 are available in our ASPTests (/archived/asptests) page.

Running ASP.NET applications

To run your ASP.NET applications with Mono, you have three classes of options:

- Apache hosting: use mod_mono (/docs/web/mod_mono/), a module that allows Apache to serve ASP.NET applications.
- FastCGI hosting: use the FastCGI (/docs/web/fastcgi/) hosting if you have a web server that supports the FastCGI protocol (for example Nginx (/docs/web/fastcgi/nginx/)) for extending the server. You also may use a web server that only has support for CGI (/archived/cgi) using **cgi-fcgi**.
- XSP: this is a simple way to get started, a lightweight and simple webserver written in C#.

For deploying applications, we recommend the use of the mod_mono (/docs/web/mod_mono/) or FastCGI (/docs/web/fastcgi/) approaches, as that will give you all the configuration options and flexibility that come with using Apache or a FastCGI server.

For quickly getting started, get familiar with Mono and ASP.NET, XSP is the ideal solution. Keep in mind that XSP is a very limited server and is only useful to get acquainted with ASP.NET and Mono, it only support HTTP 1.0 and does not provide much extensibility or configuration.

More advaned users can use the HttpListener and the ASP.NET hosting to create their own hosts for ASP.NET applications.

ASP.NET hosting with Apache

The mod_mono (/docs/web/mod_mono/) Apache module is used to run ASP.NET applications within the Apache (http://httpd.apache.org) web server.

The mod_mono module runs within an Apache process and passes all the requests to ASP.NET applications to an external Mono process that actually hosts your ASP.NET applications. The external ASP.NET host is called “mod-mono-server” and is part of the XSP module.

To use this, you must download and install the mod_mono and xsp components of Mono. mod_mono contains the actual Apache module, and xsp contains the actual ASP.NET hosting engine, both are available from our download page (/download/).

See the mod_mono (/docs/web/mod_mono/) page for details on installation and configuration.

ASP.NET hosting with Nginx

Nginx (http://wiki.nginx.org/) is a high-performance HTTP server which support running ASP.NET and ASP.NET MVC web applications through FastCGI protocol. See the FastCGI Nginx (/FastCGI_Nginx) page for details on installation and configuration.

ASP.NET hosting with XSP

XSP is a standalone web server written in C# that can be used to run your ASP.NET applications with minimal effort. XSP works under both the Mono and Microsoft runtimes. The code is available from our download page (/download/) (look for XSP web server) or from the git (/community/contributing/source-code-repository/) repository (module name: xsp).

The easiest way to start XSP is to run it from within the root directory of your application. It will serve requests on port 8080. Place additional assemblies in the bin directory. Other XSP options can be set on the command line, such as the application directory and the port to listen on.

XSP comes with a set of pages, controls and web services that you can use to test the server and see what ASP.NET looks like.

For example, once you install XSP, you can try some samples like this:

```
$ cd /usr/lib/xsp/test
$ xsp
Listening on port: 8080
Listening on address: 0.0.0.0
Root directory: /home/cvs/mcs/class/corlib/Microsoft.Win32
Hit Return to stop the server.
```

You can now browse to http://localhost:8080 (http://localhost:8080) and see various sample programs

SSL support in XSP

XSP supports SSL and TLS Client Certificates. For further details about setting it up, see the UsingClientCertificatesWithXSP (/docs/web/using-clientcertificates-with-xsp/) document.

Configuration

Applications can be configured through the web.config file, the full documentation is available from MSDN (http://msdn2.microsoft.com/en-us/library/b5ysx397.aspx), and also a Mono-specific version is available on this site here (/archived/config_systemweb).

Additionally, you can configure Mono-specific ASP.NET settings (to have applications that behave differently depending on the operating system they are deployed in) using the ASP.NET Settings Mapping (/archived/aspnet_settings_mapping) engine.

Other extensions

Check out ASP.NET Modules (/archived/aspnet_modules) for details on how to support deflate/gzip encodings and authentication.

Debugging

By default xsp and xsp2 run in Release mode, which means that debugging line-number information will not be available in stack traces when errors occur.

To obtain line numbers in stack traces you need to do two things:

1. Enable Debug code generation in your page. 2. Run Mono with the –debug command line option.

You must enable debug code generation in your page using the **Debug=”true”** in the top of your page, or setting the compilation flag in Web.config (compilation option (/archived/config/#compilation)).

Use the –debug command line option to Mono, this is done by setting the MONO_OPTIONS environment variable, like this:

```
$ MONO_OPTIONS=--debug xsp2
Listening on port: 8080 (non-secure)
Listening on address: 0.0.0.0
Root directory: /tmp/us
Hit Return to stop the server.
```

To do the same with the Apache mod_mono module, use the **MonoDebug true** directive in your apache configuration file.

Supported Versions

Mono supports ASP.NET 2.0, ASP.NET AJAX and a handful of 3.5 controls.

Limitations

Mono's ASP.NET does not implement the following features:

- Precompiled updatable web sites.
- WebParts APIs.

Work in Progress

git access

Users interested in the latest version of mod_mono and xsp can retrieve these from our public git (/community/contributing/source-code-repository/) repository. The module names are **mod_mono** and **xsp** respectively. You will also need to check out the **mcs** module as the System.Web classes are in mcs/class/System.Web.

Designer

There is work in progress on an ASP.NET Designer (/archived/aspnet_visual_designer) the designer will eventually be integrated into the MonoDevelop IDE (/docs/getting-started/development-environments/).

Legal Notices (/docs/about-mono/legal/)

(/atom.xml) (https://github.com/mono/mono)