

General Help

These are pages with general help on developing applications for .NET and Mono, and other topics not specific to boo itself but useful for a boo developer.

General Info about .NET and Mono

.NET is Microsoft's application framework and runtime engine. Most people using Windows XP have .NET version 1.1, but the 2.0 version was released in the fall of 2005. Boo now supports .NET 4.0.

There is also an open source clone of .NET called [Mono](#), that runs on Linux, Mac OS X, and Windows. It is derived from the open [Common Language Infrastructure](#) (CLI) specification used by .NET.

See also these resources:

- The [.NET Development](#) page at MSDN
- [Inside the .NET Framework](#)
- [.NET Class Library](#)
- [The Code Project](#) - samples and tutorials

Below are some resources for developing .NET/Mono applications (primarily the open source and free resources), as well as tutorials.

IDEs - Integrated Development Environments

-  [SharpDevelop](#) - free, open source, includes a forms designer/GUI builder. Windows only. SharpDevelop supports boo fully.
- [MonoDevelop](#) - Works with Mono on Linux or Mac. Peter Johanson has created a boo add-in for MonoDevelop.
- [Nant](#) - an XML-based build tool based on Java's ant build tool. Used to build boo itself.
- Visual Studio 2010 (and there is a free Express version of VS 2010 too). The standard commercial IDE used on Windows. People are interested in creating a boo plugin for VS.
- [Monolipse](#) There is a boo addin for Eclipse called monolipse. monolipse originated as another project named booclipse but has been expanded to support other EMCA languages such as C#.
- [SciBoo](#) - by Steve Donovan, a ScintillaNET based editor.
- [Smultron](#) - by Peter Borg. A Mac OS X text editor. An [add-in](#) for Boo support is authored by Grant Morgan from the Unity3d.com team. With the Smultron command window you can automatically execute your boo code without leaving the editor.

Web Development

See the [ASP.NET](#) page and tutorials.

ASP.NET is the standard, but see also:

- [Castle](#) and its MonoRail and Brail projects
- [Maverick.NET](#)

Developing Desktop GUI Applications

Windows.Forms

The standard Windows API, but you can call it from Linux as well, using Mono's Managed.Windows.Forms project.

Here are some code snippets and tutorials on using Windows.Forms:

- [Windows Forms Quickstart tutorial](#) - tutorials and code snippets for using every type of control
- [Code for Windows Forms Controls](#) - also has examples of using just about every control
- [Windows Forms FAQ](#)
- [Using .NET Windows Forms Controls](#) - article going over basic usage of each of the main form controls

The easiest way to create windows forms with boo is to use SharpDevelop. It has a form designer with support for boo, if you use the new version. If you use an older version or another C# form designer, you can convert the C# generated code to boo automatically.

GTK#

[GTK#](#) is the GUI API used in the GNOME project for Linux, but you can use it on Windows and Mac OS X too, see these articles:

- We have a [tutorial](#) on creating a sample GTK# application with the Glade GUI designer.
- [Cross platform GUI comes to .NET](#)
- [How to Write a Basic Gtk# Program with Mono](#)

GTK# on Windows

There are two versions of GTK# that run on Windows.

- One runs in .NET 1.1 (here is the [installer](#))
- the other runs in Mono, and is included in the combined [Mono installer](#) for Windows.

Note though, to compile apps that use the GTK# runtime for .NET 1.1, you may need to download the [.NET 1.1 SDK](#) to get them to compile.

Another item now available on Windows is the Gecko Runtime Engine (GRE, here is the [installer](#)). This lets you embed the Mozilla browser engine in your GTK# app as a WebControl.

See also

WX.NET and QT#

Databases

- [ADO.NET](#) - the standard API for working with relational databases, see also [Database Recipes](#) and [Database Design](#).
- Most relational database engines have interfaces to .NET/Mono, such as SQLite, MySQL, MS SQL Server, MS Access, Firebird...
- The [Database Recipes](#) page also shows examples of non-SQL-based database engines such as db4o.

Math / Statistics

- [dnAnalytics](#)
- [Math.NET](#)
- Sharp3D.Math

Graphics

- Generate PDF reports
 - [SharpPDF](#)

- [Report.NET](#)
- Structured Graphics
 - [Piccolo.NET](#)
 - SVG#
 - Diagram.NET
- Plotting / Charts
 - [NPlot](#)
 - [ZedGraph](#)
- Graph Libs / Graph Drawing
 - [QuickGraph](#)
 - [Netron](#)
 - [DiaCanvas#](#)

3D, Game Development

- For OpenGL and SDL, use:
 - [Tao Framework](#) - for examples see [OpenGL 3D Samples](#).
 - [SDL.NET](#)
- For DirectX, you need to download the DirectX SDK from Microsoft and use the Managed DirectX API. See:
 - [Managed DirectX Resources](#)
 - [Managed World](#)
- Game Engines
 - [RealmForge](#) - OpenGL and DirectX can be used interchangeably
 - [Irrlicht](#) - has .NET bindings, can also use OpenGL or DirectX
 - [Ovorp](#) - 2D game engine
- Physics
 - [ODE.NET](#)

Please add any game programming tips or samples you have to this site or the [boo page](#) at the game programming wiki.

Others

- [IndyProject](#) - networking libs for doing ftp, soap, http, smtp, pop, nntp, etc.
- Netspell

See also: [GnomeFiles](#) for some Mono apps and libs, [CSharp-Source](#) for open source .NET libraries and apps, and [CodeProject](#) for many useful .NET tutorials and code samples.