

# Microsoft VB.NET Compiler Action

This action executes the Microsoft .NET Framework VB.NET command line compiler.

On this page:

## Files & Directories

### Input Files

#### Project Directory

The directory in which the project to be compiled resides.

#### Files to compile

The list of files to compile.

#### Find source files in sub-directories

##### Sub-directories

The sub-directories to search for referenced source files required for the compile.

##### Files

The file mask to use when looking in the sub-directories for source files. Typically left as \*.vb.

### Output files

#### Output file

The file that the resulting build should be written to.

#### Target type

The form in which the output file should take. This allows specifying if the resulting file is an executable, library, etc.

#### Target the .NET compact framework

Specify whether the compact .NET framework should be referenced for this compile.

### Framework Version

#### Use default framework version (vX.X) / Use vX.X

Allows the selection of the .NET version to use for the tool. The minimum is .NET v2.0.

#### Use 64-bit tools

This options specifies as whether to force the usage of 64-bit tools on a 64-bit system or not. As FinalBuilder is currently a 32-bit application it will default to running the 32-bit version on a 64-bit system. Check this option if you require the 64-bit version to be run.

## Options

### Optimization

#### Optimize

Specifies that the common language runtime should optimise code at runtime.

### Pre-processor

## Define pre-processor symbols

Specify the symbols which should appear as pre-processor symbols in the compile.

## Miscellaneous

### Base address

The base address to load the the binary from.

### Main class

The name of the type that contains the main method.

### Root namespace

The base namespace of the project.

### Suppress compiler banner information

Turns off the banner displaying for the command line compiler. This simply reduces the amount of information logged for the build.

### Display compiler output using UTF-8 encoding

The output from the compiler is generated with UTF-8 encoding.

### Output extra information during compilation

Specify to raise the level of information reported during the compile.

### SDK path

The path to the SDK referenced by the project. These are the Microsoft mscorlib and Microsoft visualbasic library files which the project was created on.

## Debugging

### Debugging/Error checking options

#### Promote warnings to errors

When a warning is raised it will be treated as an error. Unless specifically ignored warnings will cause the build to fail.

#### Prevent compiler from displaying code for syntax-related errors and warnings

The output log for the compile will have source code removed for syntax errors. This reduces the size of the output log and makes it easier to read without the source lines included.

#### Disable integer overflow checking

Turns off checking that integer variables stay within their bounds.

#### Debug information

The resulting files generated from the compile will contain debug information.

#### Suppress warnings

Turn off seeing any warnings from the compilation process.

## Assemblies

### Assemblies

#### Partially sign the assembly

Whether to only partially sign the assembly or not. When only partially signing the assembly users of the library will have to manually install it into the global assembly cache for it to be picked up.

## **Modules to be part of this assembly**

A list of modules that are to be part of the assembly. Each module appears on a separate line within the dialogue.

## **Import metadata from a file that contains an assembly**

Tells the compiler to import public type information from the supplied assembly. This allows referencing metadata from the imported types as though they were part of the assembly.

## **Location of assemblies referenced by means of /reference**

The locations of the assemblies which are to be referenced.

## **Import a namespace from a specified assembly**

All members within the assembly, to import namespaces from, will be available to the source being built.

## **Specify a key container name for a key pair to give an assembly a strong name**

The key container to use for signing the product of the compile.

## **Specify a file containing a key or key pair to give an assembly a strong name**

The key file to use for signing the product of the compile.

## **Resources**

### **Resources**

#### **Create link to a managed resources**

Generates a link to a managed resource.

#### **Embed .NET framework resources into the output file**

The .NET resource for the project is embedded into the generated output file.

#### **Insert .ico files into the output file**

The supplied icon file is embedded into the output file. Typically an executable.

#### **Insert Win32 resources into the output file**

The Win32 resource for the project is embedded into the generated output file.

## **Language**

### **Require explicit declaration of variables**

### **Enforce strict type semantics**

### **String comparisons**