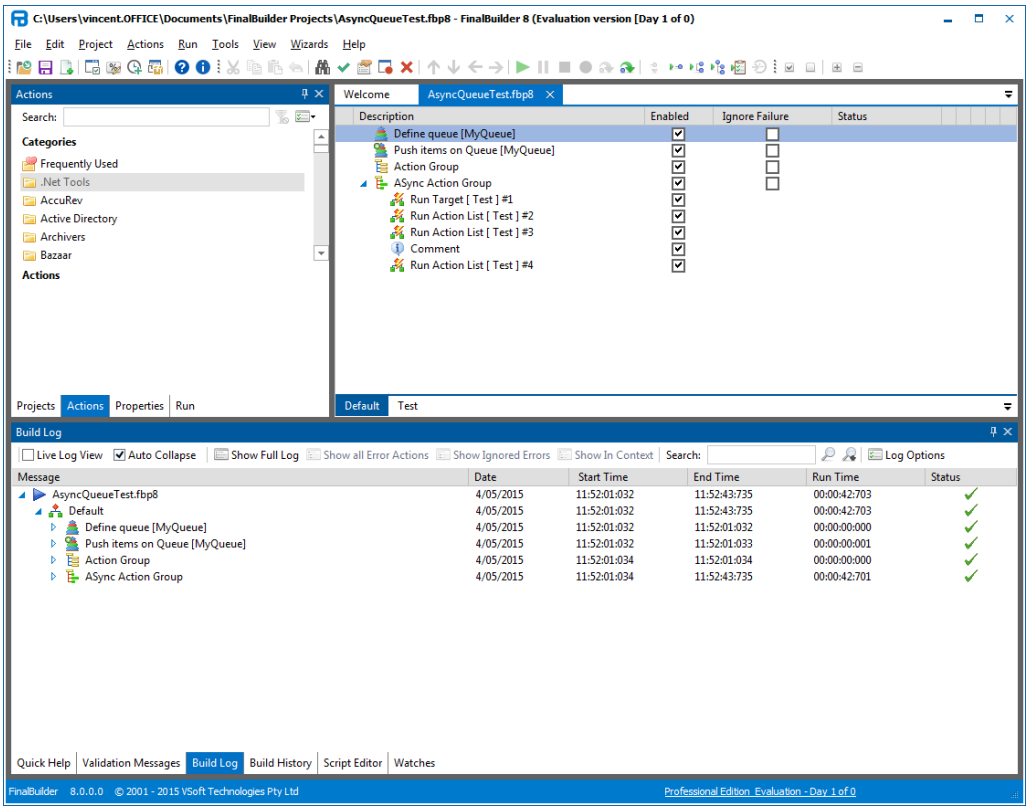
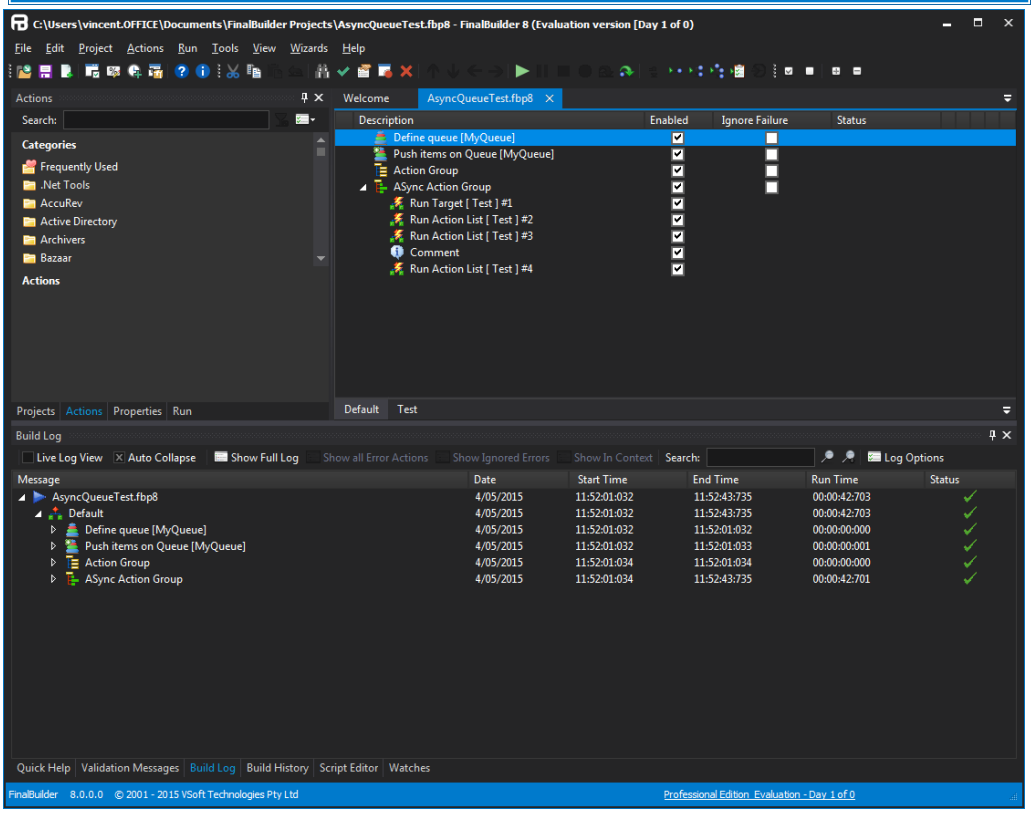
What's new in FinalBuilder 8?

IDE Themes

The IDE has two new themes, Dark and Light (yes, imaginatively named!). The IDE defaults to Dark on first run, however you can change the theme in the options quite easily. The themes are still a work in progress, we are waiting on an update from a third party control vendor to resolve some issues.

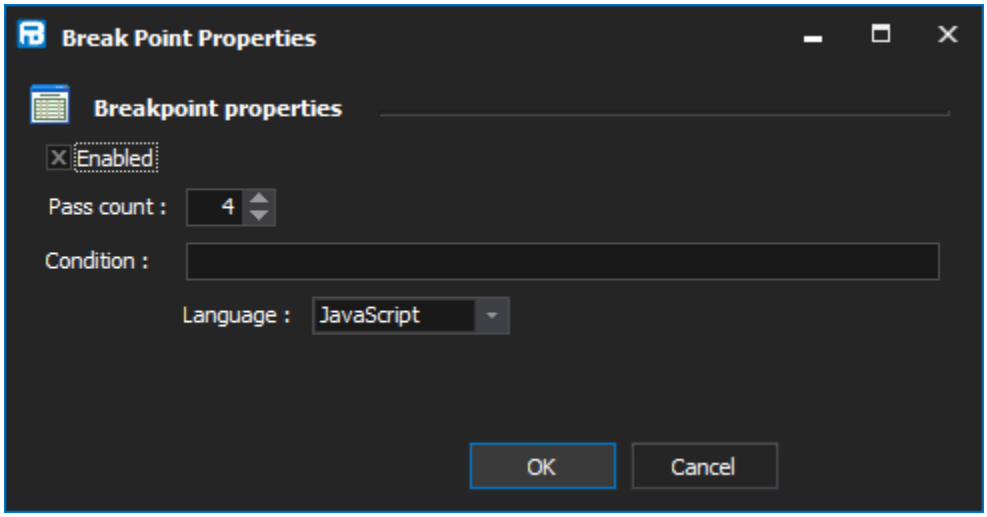




Debugger

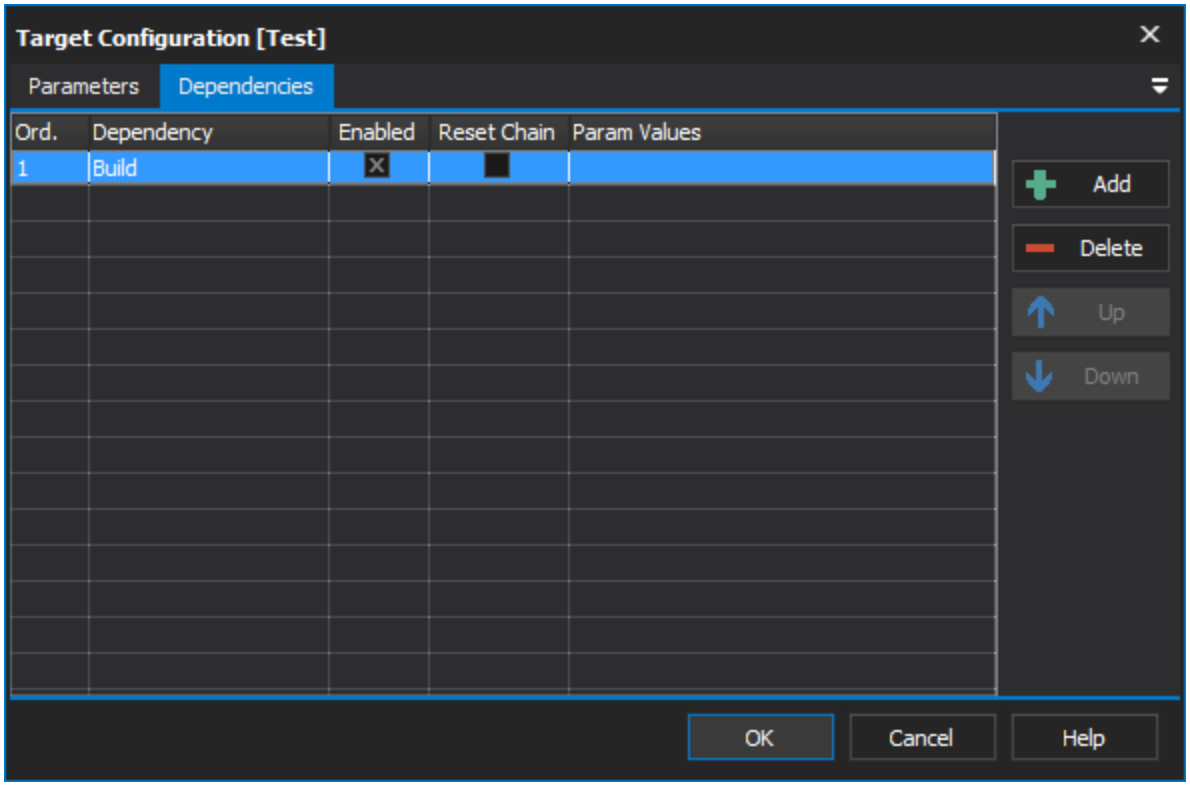
One of the most asked for features now available in FinalBuilder 8, **stepping into included projects**. In FinalBuilder 7 and earlier, you could only step over included projects, and wait for them to return. In FinalBuilder 8, you can step into the included project, if it is not already opened the IDE will open the project and switch to it automatically. To make this possible, there are now "Step Into" and "Step Over" functions. The Step into/over now also applies to targets (see below).

Debugger breakpoints now have conditions :

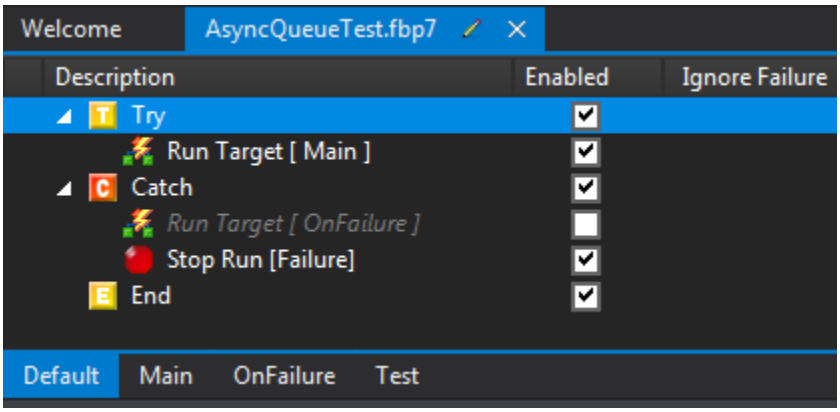


ActionLists renamed to Targets

ActionLists have been renamed to Targets. Targets can now also define dependencies, so you can for example define Clean, Build, Test, and have Test depend on Build. If you execute the Test target, and Build has not already been executed, it will be executed first before Test. Targets can be specified on the command line.

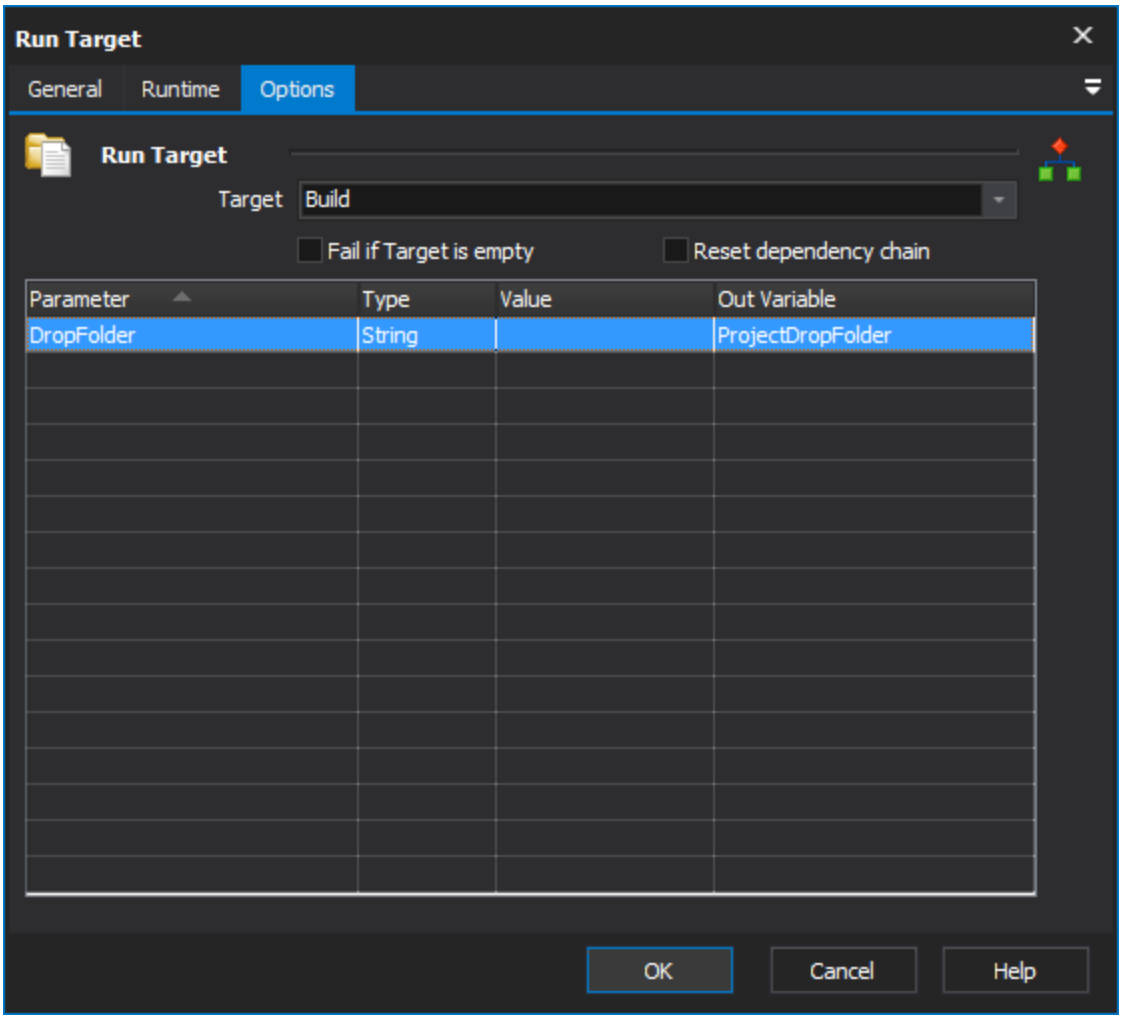


In FinalBuilder 7 and earlier, projects had a Main and an OnFailure (global error handler) actionlist. In FinalBuilder 8, projects just have a Default Target. Older projects will be imported such that the Main and OnFailure Targets are called from the Default Target inside a try/catch block.



Run Target Action

You can now return values from Targets (ie out parameters) .



Non Visible Changes

Stepping Engine

The stepping engine was rewritten to enable stepping into included projects, and to enable target dependencies. This, work, together with the new variables architecture is where the bulk of effort/time was spent in the FinalBuilder 8 development cycle.

Variables Architecture

The variables architecture and the expression evaluator were rewritten to resolve several corner case issues that we were not able to resolve in FinalBuilder 7. The expression evaluator has a new parser that will allow us to more easily extend the syntax in the future. The User variable namespace was removed, it caused too many problems with projects not running under other users, not running on the build server etc. Use Project variables instead.

Core Messaging

Changes to the messaging has allowed us to improve the performance of the stepping engine and logging, with much less thread switching. This also improved the IDE performance.

CLR Hosting

The minimum CLR version is now .NET 4.0 (ie FinalBuilder requires .net 4.0 to be installed).

Code Changes

In addition to the architectural changes, we also spent a lot of time refactoring the code, running static analysis tools over the source, looking for memory leaks, potential bugs etc. One of the results of this is reduced memory usage during a build compared to FB7. The FB8 IDE does use slightly more memory than the FB7 IDE at startup (mostly due to the heavy use of delphi generics), however the runtime memory usage is much lower.A large part of the refactoring involved unit testing (we created a new unit test framework to suite our needs!) and creating a suite of integration tests.

FBCmd

The command line parameters have changed to be more consistent and easier to specify. You can also specify one or more targets to execute (when not specified, the default target is executed).

New Project File Formats

FinalBuilder has used an XML file format since version 1, however a common complaint over the years, has been that it is difficult to diff file versions. FinalBuilder 8 has tackled this in two ways.

A new DSL style project file format (.fbp8) is now the default format, it is very easy to diff.

```
project
begin
        projectid = {04710B72-066E-46E7-84C7-C04A0D8BFE18}
        target
        begin
                name = Default
                targetid = {E6DE94D6-5484-45E9-965A-DB69885AA5E2}
                rootaction
                begin
                        action.group
                        begin
                                id = {D860420B-DE46-4806-959F-8A92A0C86429}
                        end
                end
        end
end
```

A new XML format (.fbx8), much less verbose than the old format.

```
<?xml version="1.0" encoding="UTF-8"?>
<finalbuilder>
        <project>
                <projectid>{6A717C24-D00F-4983-9FD0-148B2C609634}</projectid>
                <target>
                        <name>Default</name>
                        <targetid>{E6DE94D6-5484-45E9-965A-DB69885AA5E2}</targetid>
                        <rootaction>
                                <action.group>
                                        <id>{D860420B-DE46-4806-959F-8A92A0C86429}</id>
                                </action.group>
                        </rootaction>
                </target>
        </project>
</finalbuilder>
```

Compressed project files (.fbz8) use the DSL format internally (compressed projects are just a zip file with a project.fbp8 inside it).

The default project file encoding is now UTF-8, which is more version control friendly (some version control systems treat utf-16 as binaries).