Building a Release

Updated for Release: 1.3.20.0

These notes should describe how to build the full release of NuPattern from an empty development environment with no source code or tools installed.

Note: These notes can also found at the project site [[Developer Notes]](http://nupattern.codeplex.com/wikipage?title=Developer%27s%20Notes&referringTitle=Documentation)

# Development Environment

Building a release requires the following development tools to be installed in a side-by-side configuration on Windows Visa/7

**Note**: Building this release has not yet been verified on Windows XP or Windows 8.

Visual Studio 2010

* Visual Studio 2010 (Ultimate)
* Visual Studio 2010 SP1
* Visual Studio 2010 SDK SP1
* Visual Studio 2010 Visualization and Modeling Tools
* Nuget Package Manager
* Git Source Control Provider

Visual Studio 2012

* Visual Studio 2012 (Ultimate)
* Visual Studio 2012 SDK
* Visual Studio 2012 Visualization and Modeling Tools
* Nuget Package Manager
* Git Source Control Provider

# Obtaining the Source

Create your own development fork of the code from: <http://nupattern.codeplex.com/SourceControl>.

Clone your fork to a local repo on your hard disk based at ‘C:\Projects’. For example: C:\Projects\CodePlex\<username>\<forkname>

**Note**: All you forks are unique across all CodePlex projects.

# Supressing Strong Name Verification

**Note:** The source code is Delay-Signed, and Visual Studio will fail to build the source if we don’t instruct it to skip verification of the delay-signed assemblies.

Open the 'Visual Studio Command Prompt' (as Administrator), and type one of the following commands (depending on your installation of Windows):

(x86): "C:\Program Files (x86)\Microsoft SDKs\Windows\v7.0A\Bin\sn.exe" -Vr \*,31bf3856ad364e35

(x64): "C:\Program Files\Microsoft SDKs\Windows\v7.0A\Bin\sn.exe" -Vr \*,31bf3856ad364e35

# Building the Source

**Note:** *Steps 1, 7, 13, 14* are only necessary if you wish to fully build the source code to release quality. Not necessary to get going, or fix or patch the code.

1. *Install the latest version of the ‘NuPattern Toolkit Builder’ extension into Visual Studio 2010.*
2. Start Visual Studio 2010
3. Open the ‘Runtime.vs2010.sln’ solution
4. Switch to the ‘Debug-VS2010’ solution configuration.
5. Clean the solution
6. Rebuild the solution.
7. *In ‘Solution Builder’, right-click on the ‘Guidance’ node, and select ‘Build Guidance’.*
8. Save & Close Solution
9. Open the ‘Authoring.vs.2010.sln’ solution
10. Switch to the ‘Debug-VS2010’ solution configuration.
11. Clean the solution
12. Rebuild the solution.
13. *In ‘Solution Builder’, right-click on the ‘Pattern Toolkit’ and ‘Library’ nodes for all toolkits, and select ‘Transform Templates’. See note below.*
14. *In ‘Solution Builder’, right-click on the ‘Guidance’ node for the Authoring.PatternToolkit and Authoring.HandsOnLabs nodes, and select ‘Build Guidance’.*

Repeat steps 1-14 for Visual Studio 2012, using appropriate version of the ‘NuPattern Toolkit Builder’, appropriate solution files (\*.vs2012.sln) and build configurations (Debug-VS2012).

**Note**: Step 13 is necessary to perform manually because we have disabled ‘TransformOnBuild’ for all toolkits in this solution.

# Packaging a Release

Signing Assemblies, VSIXes etc.

# Deploying the Release

Where to upload them, and what details to update.

What pages on codeplex to update.

What releases on codeplex to post

What notifications to post

What items on Issue Tracker to deal with