Continuous Integration

Build Scripts

Overview

- Why do you need a build script?
- What belongs in a build script?
- Writing build scripts
- Running build scripts locally and on the CI server

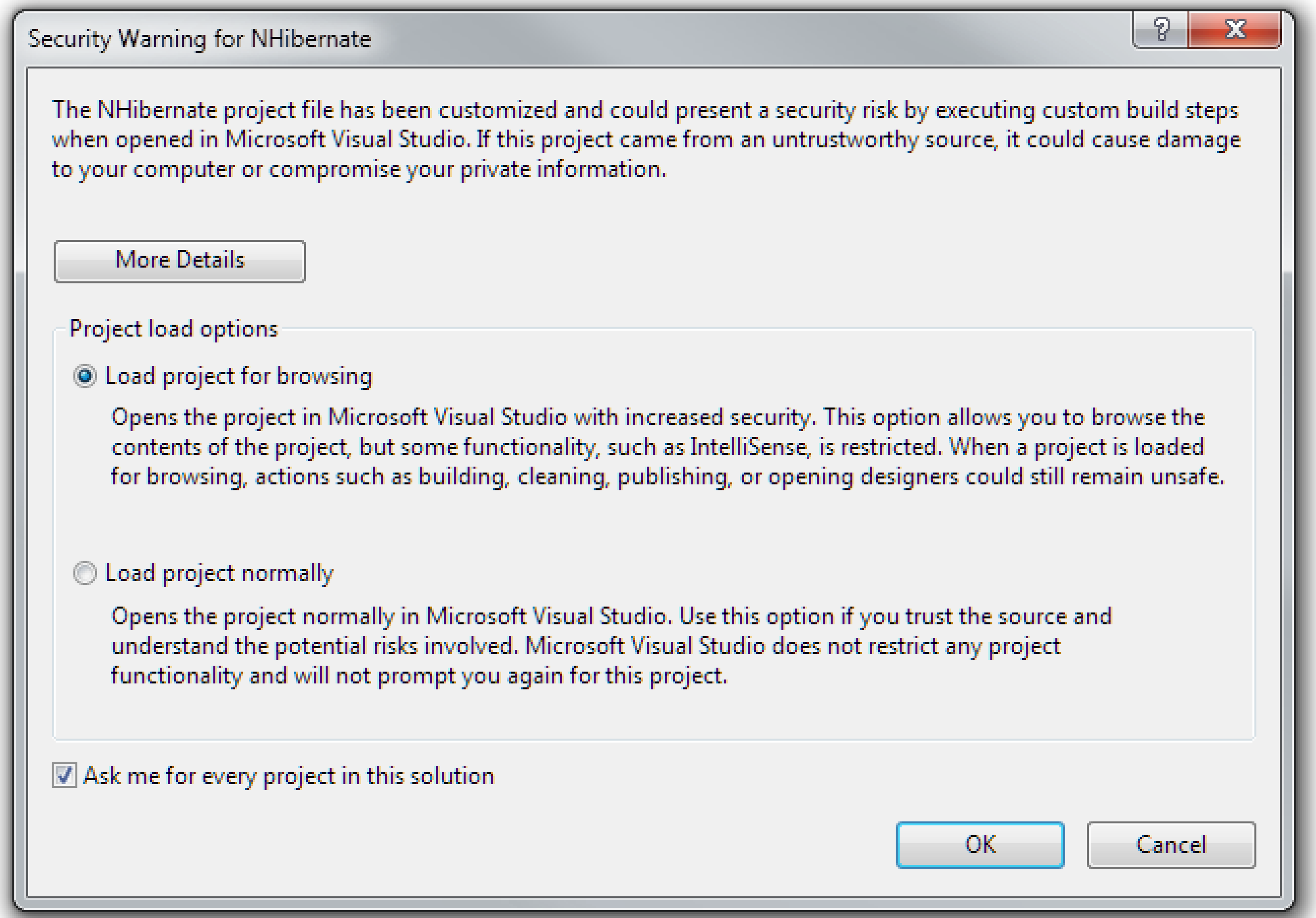
Why Do We Need a Build Script?

- **More to building a solution than compiling**
  - Clean
  - Create AssemblyInfo.[cs|vb] to avoid duplication and auto-version
  - Run automated tests, code coverage, and code metrics
  - Generate build reports and release notes
  - Build help files and documentation
  - Deploy to DEV, TEST, or PROD environment
  - Build an installer (MSI package)
  - Tag builds in version control
  - Launch website, debugger, or other application

Decision Point: How to Add Build Steps

- Modify csproj/vbproj files
- Script everything, except compile with MSBuild against csproj/vbproj
- Script everything and compile with csc.exe/vbc.exe

Recommendation: Do Not Modify csproj/vbproj



Security Warning for NHibernate

The NHibernate project file has been customized and could present a security risk by executing custom build steps when opened in Microsoft Visual Studio. If this project came from an untrustworthy source, it could cause damage to your computer or compromise your private information.

Project load options

● Load project for browsing

Opens the project in Microsoft Visual Studio with increased security. This option allows you to browse the contents of the project, but some functionality, such as IntelliSense, is restricted. When a project is loaded for browsing, actions such as building, cleaning, publishing, or opening designers could still remain unsafe.

○ Load project normally

Opens the project normally in Microsoft Visual Studio. Use this option if you trust the source and understand the potential risks involved. Microsoft Visual Studio does not restrict any project functionality and will not prompt you again for this project.

[✓] Ask me for every project in this solution

Recommendation: Do Not Script csc.exe/vbc.exe

- **Need to keep code files synchronized**
  - Simple Solution: Wildcard files via ***.cs or ***.vb
- **Need to keep references synchronized**
  - No good solution
- **Synchronization requires time and effort for often little value**

Recommendation: Script Everything, but Compile with MSBuild

- Create a separate build script
- No need to keep code files synchronized
- No need to keep references synchronized
- Additional build steps can be added easily

Why the Command Line?

- Automate common development tasks
- Ensure consistent build results
- Easily debug build problems

Hello, MSBuild

Summary

- Advantages of build scripts
- Created a build script using MSBuild
- Executed the build script locally and on the CI server