**Description**

The main functions of IoC.Configuration library are:

1) Container agnostic configuration of dependency injection using XML configuration file. The file has section where container can be specified, that will be handling dependency injection resolutions.

Currently two popular containers are supported, Ninject and Autofac, via extension libraries **IoC.Configuration.Ninject** **and IoC.Configuration.Autofac**, that are available in Nuget.org.

The dependency injection container (e.g., Autofac, Ninject) can be easily switched in configuration file.

In addition, the configuration file has sections for settings, plugins, startup actions, dynamically generated implementations of interfaces (see element iocConfiguration/dependencyInjection/autoGeneratedServices/typeFactory in example configuration files in test project that will be in GitHub soon).

2) Container agnostic configuration of dependency injection in code. The bindings are specified using IoC.Configuration chained methods, however the actual resolutions are done using one of the popular dependency injection containers, Ninject and Autofac, via extension libraries IoC.Configuration.Ninject and IoC.Configuration.Autofac.

Example of code based binding specification is:

Bind<IInterface5>().OnlyIfNotRegistered().To<Interface5\_Impl1>().SetResolutionScope(DiResolutionScope.**ScopeLifetime**);

**Documentation**

More documentation will be added soon, as well as a test project with number of tests demonstrating how to do file and code based configuration of dependency injection with Autofac and Ninject.

The XML configuration file is validated against schema file IoC.Configuration.Schema.2F7CE7FF-CB22-40B0-9691-EAC689C03A36.xsd, which is an embedded resource in IoC.Configuration.

The schema file is can be downloaded from <http://oroptimizer.com/ioc.configuration/IoC.Configuration.Schema.2F7CE7FF-CB22-40B0-9691-EAC689C03A36.xsd> or in source code project in Github.com.

A sample XML configuration file can be downloaded from <http://oroptimizer.com/ioc.configuration/IoC.Configuration.Template.xml>, or from the IoC.Configuration project, that will be soon available in github.com.