

## How to modify the file lock timeout for the assembly loader

For OPC “Classic” (COM-based) specifications, QuickOPC.NET internally loads additional assemblies that you do not reference explicitly. The whole process is rather complicated.

In some cases, especially when multiple applications written with QuickOPC are starting up concurrently, you may receive an exception similar to the following:

```
System.TypeInitializationException: The type initializer for 'OpcLabs.EasyOpc.DataAccess.EasyDAClient' threw an exception. ---> System.TimeoutException: Timeout waiting for exclusive access on execution mutex "Global\OpcLabs-{76505F20-51A1-43D3-8E2A-6AAEDAB5DBCA}-0xD6CC1689".
```

The procedure described in this application note will allow you to increase the timeout used by the concurrent applications to access the shared assembly file. Doing so reduces or eliminates the likeliness of the problem.

The file lock timeout can be controlled by a "[fileLockTimeout](#)" property in the "[OpcLabs.Reflection.AssemblyLoader](#)" configuration section. The value is an integer, and determines the duration in millisecond. Modifying the application configuration can be done without rebuilding the application.

The example below shows the application configuration file with the tracing for assembly loading enabled, and the listener configured to store the output into a comma-delimited file “Trace.csv” (in the same folder as the application).

```
<?xml version="1.0"?>
<configuration>
  <configSections>
    <section name="OpcLabs.Reflection.AssemblyLoader" type="OpcLabs.BaseLib.Reflection.AssemblyLoaderSection,OpcLabs.BaseLib,Version=5.23.1225.1,Culture=neutral,PublicKeyToken=6faddca41dacb409" />
  </configSections>
  <OpcLabs.Reflection.AssemblyLoader fileLockTimeout="90000" >
  </OpcLabs.Reflection.AssemblyLoader>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0,Profile=Client" />
  </startup>
</configuration>
```

The relevant parts of the file are highlighted. In this example, the file lock timeout is set to 90000 (milliseconds).

Note that the precise version and build number of QuickOPC must be stated in the “type” attribute. Make sure you use the right value. When servicing QuickOPC updates, the application configuration file needs to be updated as well.

Please refer to Microsoft documentation for details on the application configuration files. The application configuration file needs to be named the same as your application, with an added “.config” extension, and placed alongside the application, For example, for “MyApp.exe”, the configuration file is “MyApp.exe.config”. Note that the development tools sometimes provide “shortcuts” for the naming and placing procedure. For example, Visual Studio C# projects contain an app.config file, which becomes the application configuration file automatically – Visual Studio copies it to the output folder and renames it appropriately.

In hosted environments, such as ASP.NET, the name and location of the configuration file may be different as well.

---

The file lock timeout configuration is available in following QuickOPC versions and builds:

- QuickOPC 5.23.1255.1, and later builds of QuickOPC 5.23.
- QuickOPC 5.30.1255.1, and later builds of QuickOPC 5.30.
- QuickOPC 5.31 (not yet released) and later versions of QuickOPC.

Notes: The default file lock timeout (previously 10 seconds) has been increased to 30 seconds for the versions listed above.

When troubleshooting and resolving problems related to assembly loading in QuickOPC, you may also want to read the following document in the Knowledge Base: QuickOPC.NET-11-How to log the internal assembly loading.