Platform Notes for Ice 3.5

This page provides platform-specific notes for Ice 3.5.

On this page:

Supported platforms for Ice 3.5

This release supports the operating systems, compilers, and run-time environments listed on the ZeroC web site.

IceGrid Admin GUI requirements for Ice 3.5

As of Ice 3.5.1, the IceGrid graphical administrative utility can be used with Java6 or Java7, however the metrics graph feature is only available when running with Java7 and JavaFX.

With Ice 3.5.0, the utility requires Java7 and JavaFX.

> ⓘ JavaFX is included in recent releases of Java7.

Linux notes for Ice 3.5

Linux users should be aware of the following issues.

Mono host names

Mono requires that a system's host name be correctly configured and that it resolves to an IP address. Otherwise, Mono is unable to determine the local IP addresses, which causes the creation of object adapters that listen on INADDR_ANY/0.0.0.0 to fail with an `Ice.DNSException`.

IBM JDK jurisdiction policy files

The IBM JDK ships with limited jurisdiction policy files for the cryptographic libraries. More information, including a link to download unrestricted jurisdiction policy files, are available at the DeveloperWorks site.

Once you have downloaded the policy files, unpack them into this directory:

```
/usr/lib/jvm-private/java-1_6_0-ibm/jce/vanilla
```

Windows notes for Ice 3.5

Windows users should be aware of the following issues.

Relocated .NET assemblies

In previous releases, the .NET assemblies (`Ice.dll`, `IceSSL.dll`, etc.) were located in the `bin` subdirectory of the Ice installation directory, such as

```
C:\Program Files\ZeroC\Ice-3.4.2\bin
```

As of Ice 3.5, the .NET assemblies are stored in a separate subdirectory named `Assemblies`, as in

```
C:\Program Files\ZeroC\Ice-3.5.1\Assemblies
```

The Ice assembly for .NET Compact Framework can be found in `Assemblies\cf`.

The Ice Visual Studio Add-in automatically adds or updates the appropriate assembly references in your projects. If you do not use the Ice Visual Studio Add-in, you must update your project references to use the new location of the Ice assemblies.

Also note that this change means the `iceboxnet` executable, which still resides in `bin`, is no longer in the same directory as the Ice assemblies. For example, if your IceBox server configuration uses the IceSSL plug-in and relies on the fact that `iceboxnet.exe` and `IceSSL.dll` are in the same directory, you may need to update your configuration. Suppose your `iceboxnet` configuration included the following property:

```
Ice.Plugin.IceSSL=IceSSL:IceSSL.PluginFactory
```

Since `IceSSL.dll` was in the same directory as the executable, no additional information was necessary to assist Ice in locating the plug-in's assembly. Now, however, you will need to do one of the following:

- Use a path name for the IceSSL assembly:

  ```
  Ice.Plugin.IceSSL="C:\Program Files\ZeroC\Ice-3.5.1\Assemblies\IceSSL.dll":IceSSL.PluginFactory
  ```

- Use a strong name for the IceSSL assembly:

  ```
  Ice.Plugin.IceSSL="IceSSL,Version=3.5.1.0":IceSSL.PluginFactory
  ```

- Install the IceSSL assembly in the Global Assembly Cache (GAC)

Visual Studio compiler warnings

In previous releases, public Ice header files disabled many Visual Studio compiler warnings. Now the header files disable only a minimal set of warnings, which means compiling your Ice application may produce compiler warnings that were not reported before. To eliminate these warnings, you can modify your source code, add pragmas or disable these warnings in your Visual Studio projects.

IPv6 not supported in Java 6

IPv6 is not currently supported in Java 6 on Windows due to a JVM bug.

Monotonic clock

Ice uses the `QueryPerformanceCounter` Windows API function to measure time with a monotonic clock. If you are experiencing timing or performance issues, there are two knowledgebase articles that may be relevant for your system:

http://support.microsoft.com/?id=896256
http://support.microsoft.com/?id=895980