IVI Web Forum Question and Answer

Note: Offering an IVI driver is a requirement for all LXI-conformant instruments. The IVI Web Forum is hosted by the IVI Foundation. You can sign up at www.ivifoundation.org.

Question:

If my instrument has both IVI-C and IVI-COM drivers available for it, how do I decide which one to use and on what basis?

Answer:

IVI drivers provide IVI-C or IVI-COM APIs to support a variety of development environments and programming languages. If you have both IVI-C and IVI-COM drivers available for the same instrument, you should choose which type of driver to use based on your preferred development environment. It is easiest to use an interface that is native to that environment.

IVI-COM is based on Microsoft's Component Object Model (COM) technology. IVI-COM drivers present an interface style that is optimized for environments that support this model, such as Microsoft Visual Basic. These drivers use methods and properties to access instrument functions and information. IVI-COM drivers that include .NET interop assemblies can be used in languages that support calling .NET assemblies, such as C#, VB.NET, and LabVIEW.

IVI-C drivers are an extension of VXI*plug&play* specifications. They provide an interface style that is best for C-friendly languages. ANSI C is a long established standard, and offers a platform portability advantage over COM. IVI-C drivers are optimal for use in C development environments, such as LabWindows/CVI, or C++ development environments, such as Microsoft Visual Studio.

If your preferred development environment is LabVIEW, either IVI-C or IVI-COM drivers can be used. However, the best option would be to use LabVIEW Plug and Play instrument drivers, which are written in native LabVIEW code, or an IVI driver with LabVIEW wrapper VIs.

When selecting an instrument driver you should consider all available choices and choose the driver that fits your application requirements and development environment.

Check out the IVI Getting Started Guide to learn more about using IVI drivers in a variety of development environments.