Explore LabWindows/CVI Features

The LabWindows/CVI ANSI C integrated programming environment helps you create custom engineering applications. You can use it to manage your project, edit and debug source code, build a user interface, and test code output and performance in one streamlined, tabbed workspace. LabWindows/CVI includes tools for advanced debugging, code documentation, and system deployment so you can integrate source code control, requirements, and data management systems. The software also makes it easier to quickly acquire data from GPIB, USB, serial, Ethernet, PXI, VXI, and FPGA instruments using the built-in instrument I/O libraries, built-in instrument drivers, or the two interactive measurement assistants. LabWindows/CVI users with an active Standard Service Program (SSP) membership are eligible to upgrade to the latest version LabWindows/CVI.

What Can I Do With LabWindows/CVI?

LabWindows/CVI provides an all in one work environment for ANSI C development to create test and measurement applications. Learn more about how LabWindows/CVI can be used in common applications below.

Develop Tests for Production ATE Systems

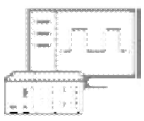
With LabWindows/CVI, you can create and deploy test applications by controlling virtually all your hardware and designing custom user interfaces to improve your development time.

See How

Optimize Your Test Code for Creating Production ATE Systems

Using LabWindows/CVI, create higher performing test applications by taking advantage of built-in performance analysis tools, parallel execution options and a comprehensive data storage system to optimize your code and your manufacturing test throughput.

See How



Set Up Tests to Characterize Electronic Devices

With LabWindows/CVI, you can automate the validation of your devices to meet challenging time-to-market and performance requirements. Create flexible applications by taking advantage of seamless hardware integration, custom UI creation and in-depth hardware analysis libraries.

See How

Debug Code for Characterizing Electronic Devices

Working with LabWindows/CVI, you can take advantage of the intuitive debugging tools, for single and multi-process tasks, to make refining your test and measurement applications easier and minimizes downtime due to broken code.

See How

Software Benefits

Standard Service Program

Every purchase includes a renewable, one-year membership to the Standard Service Program (SSP) for software, which offers the following:

- Live phone and email technical support from local, degreed engineers
- Automatic version updates to LabWindows/CVI
- 24x7 access to selected online training and virtual demonstrations
- Access to historical versions in case you need to share code with your team

Renew SSP

Which LabWindows/CVI Edition Is Right for My Project?

LabWindows/CVI Base

Starting from $ 1,389.00

- Recommended for desktop and real-time measurement applications
- Includes device drivers for NI hardware and third-party instruments
- Includes basic mathematics, signal processing, and debugging tools

LabWindows/CVI Full

- Recommended for inline advanced mathematics, signal generation and signal processing
- Includes an OpenMP library and memory leak detection for test optimization
- Includes SQL tools for database connectivity

- Compare Features and Pricing

Developer(s)	National Instruments
Initial release	January 1989; 31 years ago
Stable release	2019 / May 2019; 9 months ago
Operating system	Windows 10/Windows 8/Windows 7/Vista/XP with Linux run-time support and Pharlap real-time run-time support
Type	Data acquisition, instrument control, test automation, analysis and signal processing
Website	ni.com/cvi

LabWindows/CVI (CVI is short for C for Virtual Instrumentation) is an ANSI C programming environment for test and measurement developed by National Instruments. The program was originally released as LabWindows for DOS in 1987, but was soon revisioned (and renamed) for the Microsoft Windows platform. The current version of LabWindows/CVI (commonly referred to as CVI) is 2019.

LabWindows/CVI uses the same libraries and data-acquisition modules as the better known National Instrument product LabVIEW and is thus highly compatible with it.

LabVIEW is targeted more at domain experts and scientists, and CVI more towards software engineers that are more comfortable with text-based linear languages such as C.

Release history

Starting with LabVIEW 8.0, major versions are released around the first week of August, to coincide with the annual National Instruments conference NI Week, and followed by a bug-fix release the following February.

In 2009 National Instruments started to name the releases after the year in which they are released. The bug-fix is called a Service Pack (for instance, the 2009 service pack 1 is released in February 2010). [1]

Name/version	Build number	Date
LabWindows/CVI project begins		1987
LabWindows/CVI 1.0 (for DOS)		Jan 1989
LabWindows/CVI 2.0 (for DOS; GUI Tools and Memory Extender)		Apr 1991
LabWindows/CVI 3.0 (for DOS/Windows 3.1/Solaris)		Mar 1994
LabWindows/CVI 3.1 (generate codes automatically)		Jul 1995
LabVIEW 3.1.1 (first release with "application builder" capability)	??	1995
LabWindows/CVI 4.0 (External C/C++ compiler compatibility)		May 1996
LabWindows/CVI 4.0.1		Aug 1996
LabWindows/CVI 5.0 (support VXI and IVI)		Feb 1998
LabWindows/CVI 5.5 (Multithreaded libraries, debugging)		Feb 2000
LabWindows/CVI 6.0 (support ActiveX, improved presentation)		Oct 2001
LabWindows/CVI 7.0 (use Workspace)		Jul 2003
LabWindows/CVI 7.1 (completion automatically)		Sep 2004
LabWindows/CVI 8.0 (support .NET assemblies)		Oct 2005

LabWindows/CVI 8.0.1		
LabWindows/CVI 8.1		2006
LabWindows/CVI 8.1.1		
LabWindows/CVI 8.5		2007
LabWindows/CVI 8.5.1		
LabWindows/CVI 9.0 (support ANSI C99)		2008
LabWindows/CVI 9.0.1		
LabWindows/CVI 2009 (create 64-bit applications)	9.1	2009
LabWindows/CVI 2009 SP1		
LabWindows/CVI 2010	10.0	2010
LabWindows/CVI 2010 SP1		
LabWindows/CVI 2012	12.0	2012
LabWindows/CVI 2012 SP1		
LabWindows/CVI 2013	13.0	2013
LabWindows/CVI 2013 SP1		
LabWindows/CVI 2013 SP2		
LabWindows/CVI 2015 (upgrade to Clang 3.3)	15.0	2015
LabWindows/CVI 2015 SP1	15.1	2016
LabWindows/CVI 2017 (tracepoints, word/semantic highlighting, thread-specific breakpoints, comment/uncomment)	17.0	2017
LabWindows/CVI 2019	19.0	May 2019