



LabVIEW (acrónimo de Laboratory Virtual Instrumentation Environment Workbench) es una plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico. Recomendado para sistemas hardware y software de pruebas, control y diseño, simulado o real y embebido, pues acelera la productividad. El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico.

Este programa fue creado por National Instruments (1976) para funcionar sobre máquinas MAC, salió al mercado por primera vez en 1986. Ahora está disponible para las plataformas Windows, UNIX, MAC y GNU/Linux. La última versión es la 2012, con la increíble demostración de poderse usar simultáneamente para el diseño del firmware de un instrumento RF de última generación, a la programación de alto nivel del mismo instrumento, todo ello con código abierto.

Los programas desarrollados con LabVIEW se llaman Instrumentos Virtuales, o VIs, y su origen provenía del control de instrumentos, aunque hoy en día se ha expandido ampliamente no sólo al control de todo tipo de electrónica (Instrumentación electrónica) sino también a su programación embebida, comunicaciones, matemáticas, etc. Un lema tradicional de LabVIEW es: «*La potencia está en el Software*», que con la aparición de los sistemas multinúcleo se ha hecho aún más potente. Entre sus objetivos están el reducir el tiempo de desarrollo de aplicaciones de todo tipo (no sólo en ámbitos de Pruebas, Control y Diseño) y el permitir la entrada a la informática a profesionales de cualquier otro campo. LabVIEW consigue combinarse con todo tipo de software y hardware, tanto del propio fabricante -tarjetas de adquisición de datos, PAC, Visión, instrumentos y otro Hardware- como de otros fabricantes.

Su principal característica es la facilidad de uso, válido para programadores profesionales como para personas con pocos conocimientos en programación pueden hacer programas relativamente complejos, imposibles para ellos de hacer con lenguajes tradicionales. También es muy rápido hacer programas con LabVIEW y cualquier programador, por experimentado que sea, puede beneficiarse de él. Los programas en LabView son llamados instrumentos virtuales (VIs) Para los amantes de lo complejo, con LabVIEW pueden crearse programas de miles de VIs (equivalente a millones de páginas de código texto) para



aplicaciones complejas, programas de automatizaciones de decenas de miles de puntos de entradas/salidas, proyectos para combinar nuevos VIs con VIs ya creados, etc. Incluso existen buenas prácticas de programación para optimizar el rendimiento y la calidad de la programación. El LabVIEW 7.0 introduce un nuevo tipo de subVI llamado VIs Expreso (Express VIs). Estos son VIs interactivos que tienen una configuración de caja de diálogo que permite al usuario personalizar la funcionalidad del VI Expreso. El VIs estándar son VIs modulares y personalizables mediante cableado y funciones que son elementos fundamentales de operación de LabVIEW.

Presenta facilidades para el manejo de:

- Interfaces de comunicaciones:
  - Puerto serie
  - Puerto paralelo
  - GPIB
  - PXI
  - VXI
  - TCP/IP, UDP, DataSocket
  - Irda
  - Bluetooth
  - USB
  - OPC...
- Capacidad de interactuar con otros lenguajes y aplicaciones:
  - DLL: librerías de funciones
  - .NET
  - ActiveX
  - Multisim
  - Matlab/Simulink
  - AutoCAD, SolidWorks, etc
- Herramientas gráficas y textuales para el procesado digital de señales.
- Visualización y manejo de gráficas con datos dinámicos.
- Adquisición y tratamiento de imágenes.



- Control de movimiento (combinado incluso con todo lo anterior).
- Tiempo Real estrictamente hablando.
- Programación de FPGAs para control o validación.
- Sincronización entre dispositivos.

Como se ha dicho es una herramienta gráfica de programación, esto significa que los programas no se escriben, sino que se dibujan, facilitando su comprensión. Al tener ya prediseñados una gran cantidad de bloques, se le facilita al usuario la creación del proyecto, con lo cual en vez de estar una gran cantidad de tiempo en programar un dispositivo/bloque, se le permite invertir mucho menos tiempo y dedicarse un poco más en la interfaz gráfica y la interacción con el usuario final. Cada VI consta de dos partes diferenciadas:

- *Panel Frontal*: El *Panel Frontal* es la interfaz con el usuario, la utilizamos para interactuar con el usuario cuando el programa se está ejecutando. Los usuarios podrán observar los datos del programa actualizados en tiempo real (como van fluyendo los datos, un ejemplo sería una calculadora, donde tu le pones las entradas, y te pone el resultado en la salida). En esta interfaz se definen los *controles* (los usamos como entradas, pueden ser botones, marcadores etc..) e *indicadores* (los usamos como salidas, pueden ser gráficas ....).
- *Diagrama de Bloques*: es el programa propiamente dicho, donde se define su funcionalidad, aquí se colocan íconos que realizan una determinada función y se interconectan (el código que controla el programa -). Suele haber una tercera parte *ícono/conector* que son los medios utilizados para conectar un VI con otros VIs.-

En el panel frontal, encontraremos todo tipos de controles o indicadores, donde cada uno de estos elementos tiene asignado en el diagrama de bloques una terminal, es decir el usuario podrá diseñar un proyecto en el panel frontal con controles y indicadores, donde estos elementos serán las entradas y salidas que interactuarán con la terminal del VI. Podemos observar en el diagrama de bloques, todos los valores de los controles e indicadores, como van fluyendo entre ellos cuando se está ejecutando un programa VI.

LabVIEW se puede usar para crear muchos tipos de programas sin estar enfocado a un



ámbito en particular.

- Su alter ego es LabWindows/CVI de National Instruments, el cual permite usar las mismas funcionalidades pero con la programación en lenguaje C y el acceso a las bibliotecas Win32 gracias al SDK de Windows.
- Las mismas funcionalidades de instrumentación, análisis y controles gráficos son también accesibles en Visual Basic, C++ o C# con Visual Studio gracias a Measurement Studio de National Instruments. Así, se puede disfrutar de la programación orientado a objetos y el framework .NET.
- Las otras alternativas van desde lenguajes genéricos como C o Visual Basic pero sin la ayuda de la bibliotecas de funcionalidades de National Instruments, a otras herramientas gráficas como HP-VEE, ahora Agilent-VEE.

Una de las alternativas en software libre es MyopenLab.

Fue en 1986 que la primera versión de LabVIEW se realiza sobre Macintosh. Se sigue un trabajo incesante para añadir funcionalidades:

- 1986 : LabVIEW 1.0, primera versión en Mac OS
- 1990 : LabVIEW 2.0, máximo aprovechamiento de los resultados
- 1992 : LabVIEW 2.5, primera versión en Windows 3.1 y Solaris
- 1993 : LabVIEW 3.0
- 1994 : LabVIEW 3.0.1, primera versión en Windows NT
- 1994 : LabVIEW 3.1
- 1995 : LabVIEW 3.1.1, integración del Application Builder (creación de archivos ejecutables)
- 1996 : LabVIEW 4.0
- 1997 : LabVIEW 4.1
- 1998 : LabVIEW 5.0, multitarea, contenedores ActiveX, asistente para la adquisición de dato (tarjetas de adquisición DAQ) e asistente para el control de instrumentos
- 1999 : LabVIEW 5.1, primera versión para Linux, primera versión de LabVIEW RT (Real Time)



- 2000 : LabVIEW 6.0, controles gráficos en 3D, referencias de controles
- 2001 : LabVIEW 6.1, mejoramiento y correcciones, primera versión en Palm OS
- 2003 : LabVIEW 7.0, VI Express, primera versión en Windows Mobile 2003
- 2004 : LabVIEW 7.1, traducción en francés, alemán y japonés
- 2005 : LabVIEW 8.0, Project Explorer, XControls, shared variables
- 2005 : LabVIEW 8.1, mejoramiento y correcciones
- 2006 : LabVIEW 8.20, Programación orientada a objetos
- 2007 : LabVIEW 8.5, primera versión del toolkit FPGA y del toolkit Statechart
- 2008 : LabVIEW 8.6, limpieza automática de los diagramas
- 2009 : LabVIEW 2009, MathScript RT, LabVIEW de 64 Bits, Recursividad Nativa, Orientación de Objetos en LabVIEW RT/FPGA, SSL (Seguridad) para Servicios Web, Limpieza Parcial de Diagrama de Bloques, Fácil Acceso a Archivos TDMS desde Microsoft Excel, Nuevos Controles de Visualización de Datos en 2D y 3D, Métricas de Complejidad de Código, Referencias de Datos, Acceso más Rápido a Archivos TDMS, Crear Montajes .NET, Herramientas de Réplica de Sistemas y VIs PDE (ecuación diferencial parcial)
- 2010 : LabVIEW 2010, - Combinación de instrucciones.

- Salto de hilos. - Reemplazo escalar de conjuntos. - Propagación condicional. - Eliminación del "Tail Call". - Re-asociación de expresiones. - Movimiento de la curva de código invariable. - Curva inmutable e índice de separación. - Simplificación de variables de inducción. - Curva de desenrollado. - Eliminación de código muerto. - Propagación de constantes condicionales dispersas. La miscelánea tecnológica integrada a la edición 2010 compuesta mayoritariamente por DFIR y LLVM suministra mejoras de desempeño mediante un esquema de optimización. Lo anterior hace del LabVIEW 2010 un sistema más rápido cuyos niveles de eficiencia van desde el 20% al 200% dependiendo de la aplicación. Incluso si se trata de controles de tiempo real, los beneficios de acuerdo a los detalles técnicos de la compañía se pueden administrar desde el compilador.