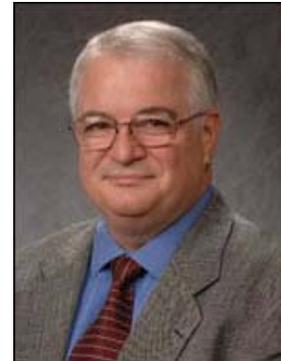


NI LabVIEW 8.20 Marks 20 Years of Innovation



In this article, Jeff Kodosky, the father of LabVIEW, celebrates 20 years of innovation with a letter to LabVIEW developers.

Dear LabVIEW Developers,

It was a little more than 20 years ago when we first formed ideas of what is now known as LabVIEW. We had a vision to create something that would do for engineers and scientists what the spreadsheet did for financial analysts. What started as an idea in 1984 turned into the first LabVIEW release – LabVIEW 1.0 on the Apple Macintosh Plus in 1986. While a lot about LabVIEW has changed since 1986, its purpose and core concepts remain the same.

Today, the 20th anniversary edition of LabVIEW, LabVIEW 8.20, adds the ability to combine textual math with graphical programming, features object-oriented programming, and offers improvements for designing and targeting embedded devices. LabVIEW 8.20 does this by building on the principles that led to the first version in 1986 and to the success of LabVIEW over its 20-year history.

I attribute much of the LabVIEW development ease of use to two fundamental concepts – the front panel and block diagram, both present since LabVIEW 1.0. Creating a UI by dragging and dropping controls and indicators and writing the code by wiring icons together are key elements of LabVIEW. The original LabVIEW block diagram has improved by including additional models of programming. We now have an event structure as well as custom timing structures, such as the timed loop and new diagram types, which include the simulation diagram. When we created LabVIEW 1.0, it was possible to control instruments over GPIB. With LabVIEW today, we have the power to run real-time applications, develop custom hardware using FPGAs, create communications and RF test systems, and design systems with embedded 32-bit processors. Throughout the development of LabVIEW, we have been committed to creating an open environment that provides connectivity to NI and third-party hardware and includes tools to reuse external software with LabVIEW.

As we have released new versions of LabVIEW, I am constantly amazed at how you have used the software to meet your application needs. One such application is from Christopher Atwood of Tristan Technologies, Inc. (see Figure 1). This application uses LabVIEW to monitor and diagnose brain activity in infants at severe risk of developing cerebral palsy and

epilepsy, so medical doctors can intervene at an early stage. Without formal software development training, a single engineer and domain expert developed this advanced application in less than one year using NI PXI hardware and LabVIEW software.

It also has been exciting for me to see how LabVIEW itself is increasingly developed in LabVIEW, sometimes referred to as “eating your own dog food.” Many of the features you see today in LabVIEW, including the application builder, are developed in LabVIEW. Its ease of use as well as our ability to develop within the software have been key to our collaboration with LEGO®, which led to the creation of LEGO MINDSTORMS® NXT, the next-generation robotics platform from LEGO. Based on LabVIEW, MINDSTORMS NXT uses intuitive icon-based programming and provides easy-to-use development for younger users and the extensibility to create sophisticated robotics programs (see Figure 2). I am excited to see what the future brings with this new crop of graphical programmers.

As both customers and LabVIEW enthusiasts, you not only have delivered applications with LabVIEW, you also have helped shape the features and direction of LabVIEW. LabVIEW 8.20 is no exception. As your applications have grown in complexity and the number of developers working on projects has increased, you have asked for better tools to manage your applications and development. LabVIEW 8.20 adds object-oriented programming so you can extend modular code development beyond subVIs and project libraries. Your need to develop textual math algorithms with LabVIEW has led to MathScript, an interactive textual math editor in LabVIEW, and the ability to embed your scripts into the block diagram of your VIs.

As we release each version of LabVIEW, we look at the future to see how LabVIEW can meet your evolving needs. The last 20 years have been an exciting time for all of us. We are just getting started with the next 20 years, and I look forward to continuing to work with you, the LabVIEW community, to build this future together.

Sincerely,
Jeff Kodosky

NI Fellow **Jeff Kodosky** cofounded National Instruments with Dr. James Truchard and William Nowlin in 1976 while working at The University of Texas at Austin. Today he is a respected mentor in the NI global R&D organization and continues to chart new directions for National Instruments flagship product, the LabVIEW graphical development platform.