Editor's Note: For the past several years, I've used the bagpipe tuners in the iOS App Store to demonstrate the power of vibrant ecosystems. Like Apple, NI has an ecosystem. It's built on our open LabVIEW platform and clearly defined APIs and hardware specifications. With ecosystems, users don't have to start from scratch unless they want to. Critical to the health and productivity of an engineering platform, a vibrant ecosystem delivers an order of magnitude more value faster than any test vendor's R&D department ever could. Understand the ecosystems surrounding your test systems and embrace them.
—Dr. James Truchard

Software-Centric Ecosystems

The transition under way in mobile devices offers insight into an important trend for test and measurement: the power of a software-centric ecosystem. Early-model mobile telephones were built to make calls first and later send text messages, but the capabilities were almost completely defined by the vendor. Once the software on these devices was opened up to the user, capability ranging from music players to cameras to email quickly followed. But the effectiveness of the transition was more than just an open software experience. Apple, and later Google, built robust ecosystems around their products and created a community of developers for "apps" that accelerated usefulness.

The inherent openness and community concept for mobile phones arguably could have been fostered by mobile phone providers themselves, but in this case it was Apple and Google that worked on software environments first and deployed hardware second. By exposing an appropriate level of customization to users or third-party developers, they succeeded in changing the way consumers view their mobile phones.

This same concept is making an impact on the test and measurement industry. Communities of developers and integrators, building on standard software platforms, are using commercial off-the-shelf technology to extend the functionality of complex hardware into applications previously impossible. The level of productivity and collaboration delivered by software-centric ecosystems will have a profound effect on test system design over the next three to five years.

Ecosystems Defined

In his book *The Death of Competition: Leadership and Strategy in the Age of Business Ecosystems,* James F. Moore defines a business ecosystem in the following way: "An economic community supported by a foundation of interacting organizations and individuals—the organisms of the business world. The economic community produces goods and services of value to customers, who are themselves members of the ecosystem. The member organisms also include suppliers, lead producers, competitors, and other stakeholders. Over time, they coevolve their capabilities and roles, and tend to align themselves with the directions set by one or more central companies."

For test and measurement, cross-industry collaboration is nothing new. Active industry groups such as the IVI Foundation, PXI Systems Alliance, and LXI Consortium have been bringing industry players together for decades but often with key gaps as outlined in Moore's description. With active participation in these groups now including software-specific, hardware-specific, and joint hardware/software vendors, the focus on enabling

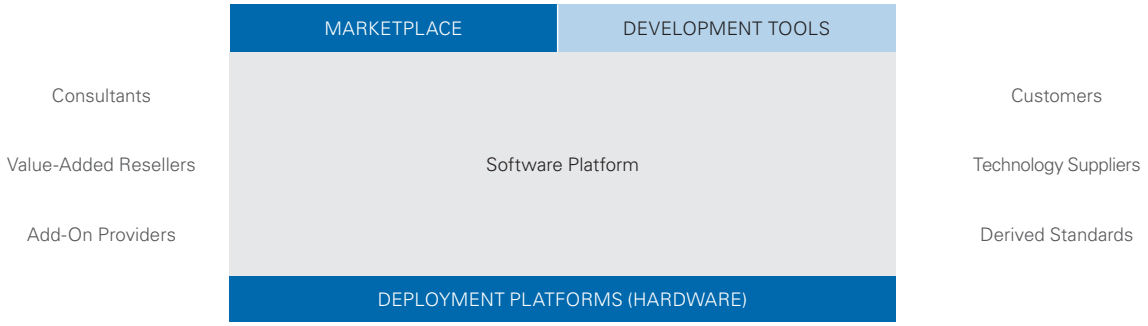
interoperability for proprietary architectures and ease of use for open architectures is fostering business ecosystems. The most successful examples of current ecosystems in this industry, though, are rooted in software. LabVIEW is an example of application software made more valuable through its ecosystem. Significant numbers of engineers have been trained on LabVIEW and developed add-ons suitable for private application needs as well as others through commercial vehicles like the LabVIEW Tools Network. System integrators in the NI Alliance Partner Network as well as LabVIEW Consultants work to deploy this ecosystem. With every additional supplier, producer, competitor, or other stakeholder, the value of the software to each user grows.

Ecosystems in Open and Proprietary Software/Hardware Architectures

An extremely useful ecosystem standardizes the way we communicate with instruments—Interchangeable Virtual Instrument (IVI) drivers. By offering a common means of communicating to similar instruments across multiple vendors at the application programming interface level, the IVI Foundation reduced the learning curve for users and the development cycle for vendors. This opened the door for third parties to create drivers, aggregation websites to house them (like IDNet on ni.com), and abstraction layers to be created on top of them. With well-architected hardware abstraction layers, technology insertion for systems designed to last decades became not only possible but routine. The ecosystem fostered by standardization was crucial in achieving this, and it continues to grow with the recent ratification of native Microsoft .NET implementations for IVI in the past few years. When programming FPGAs in applications like inline signal processing or DUT control, most test engineers practically require hardware and software from a single vendor to achieve the abstraction necessary to meet their skill levels. When these solutions are delivered in the context of a software-centric business ecosystem, the platform can retain as much user flexibility as a disparate or interchangeable hardware/software approach. For example, the FPGA programming capability of the LabVIEW reconfigurable I/O (RIO) architecture can incorporate third-party VHDL or Xilinx CORE Generator IP inside the LabVIEW system design toolchain. The LabVIEW Tools Network helps users exchange sample projects and compiled code to support different application spaces among users and vendors in automated test. This ecosystem opens the doors of FPGA programming to nontraditional automated test spaces and offers the IP necessary to be successful. Without a software-centric ecosystem, many viable open platforms have struggled. The xTCA platforms have seen adoption in telecommunication infrastructure and interest from the high-energy physics community, but they have failed to develop a strong ecosystem in automated test. The multiple form factor, communication bus, and software options presented by the platform have delayed or complicated adoption by leading vendors. While efforts to rein in those options and improve them for automated test are under way in the AXIe Consortium, success or failure will be dictated by the use of a software-centric ecosystem.

The Future of Ecosystems in Automated Test

Over the next three to five years, automated test systems will become more software-centric and ecosystems will have more impact on the value users derive from these platforms. The previous examples of instrument communication and abstracted FPGA programming are just the beginning for automated test ecosystems. As software vendors take greater advantage of their ecosystems and leverage commercialization models for third-party IP, the scenario unfolding for mobile devices will have a transformative effect on the test and measurement industry.



As software platforms develop ecosystems that grow with each additional customer, supplier, add-on provider, and so on, they become more valuable to each user. Software-centric ecosystems will make a large impact on the value that engineers derive from software-based test platforms.