Mike Santori
NI Business and Technology Fellow

Making the IoT Work for Test

- The IoT and IIoT are making test more complex.
- IoT technologies can help address automated test challenges.
- Engineers need to understand and focus on the use cases with the most business value.

Internet of Things (IoT) devices and Industrial IoT (IIoT) systems are increasing in complexity, from semiconductors to electronic subsystems to the smart machines at the heart of Industry 4.0. Test is a hidden but critical function in this product chain, and increasing IoT device complexity is, in turn, increasing test complexity. The IoT can also greatly enhance automated test. Applying IoT capabilities such as systems management, data management, visualization and analytics, and application enablement to the automated test workflow can better equip test engineers to overcome the challenges of the IoT.

Managing Test Systems

Connected, managed devices are fundamental to the IoT and IIoT. Many test systems, however, are not connected or well managed, even as they become more distributed. Often, test engineers have difficulty tracing the software running on any given hardware or just knowing the whereabouts of systems, much less tracking performance, utilization, and health.

Fortunately, most modern test systems are based on a PC or PXI and can directly connect to the enterprise, which enables additional capabilities such as managing software and hardware components, tracking usage, and performing predictive maintenance to maximize the value of test investments.

Ingesting and Managing Data

The business value of the IoT derives from massive amounts of data generated by connected systems. Consuming test data is difficult, though, because of the many data formats and sources, from raw analog and digital waveforms in time and frequency to parametric measurements often gathered at significantly higher rates and volumes than from consumer or industrial devices. To make matters worse, test data is often stored in silos with little standardization. Consequently, this data is "invisible" to a business, making it easy to miss out on valuable insights at other phases of the product life cycle. Prior to implementing a comprehensive, IoT-enabled data management solution, Jaguar Land Rover (JLR) analyzed only 10 percent of its vehicle test data. JLR Powertrain Manager Simon Foster said, "We estimate that we now analyze up to 95 percent of our data and have reduced our test cost and number of annual tests because we do not have to rerun tests."

Applying IoT capabilities to automated test data begins with ready-to-use software adapters for ingesting standard data formats. These adapters must be built with an open, documented architecture to enable ingestion of new and unique data, including non-test data from design and production. Test systems must be able to share their data with standard IoT and IIoT platforms to unlock value from data at the enterprise level.

Visualizing and Analyzing Data

Using general business analytics software for test data can be difficult because this data is often complex and multidimensional. Also, typical business charting capabilities don't include common visualizations in test and measurement, like combined graphs of analog and digital signals, eye diagrams, Smith charts, and constellation plots.

> "It will soon become standard that our customers require the management and maintenance of test assets from around the world. We must reshape our test architectures to integrate IoT technologies, especially to evolve configuration management and data analytics and support the digitalization of our business for Industry 4.0."
>
> Franck Choplain, Digital Industry Director, Thales

Test-oriented schemas with appropriate metadata enable tools to provide visualization and analysis for test data and correlate it with design and production data. Well-organized test data allows engineers to apply analytics from basic statistics to artificial intelligence and machine learning. This enables workflows that integrate and leverage common tools, like Python, R, and The MathWorks, Inc. MATLAB® software, and generates greater insights from data.

Developing, Deploying, and Managing Test Software

The world is moving from exclusively using desktop applications toward augmenting with web and mobile apps. This transformation can be difficult to realize for test. Computing at the device under test (DUT) is needed to process large amounts of data and make real-time pass/fail decisions, and local operators need to interact with the tester and the DUT. At the same time, companies want to remotely access testers to see results and operating status such as utilization. To address this, some companies have built one-off architectures to manage software centrally, and they download software to testers based on the DUT. But because of this, they must maintain their custom architecture, which requires additional resources that could be applied to activities with higher business value.

Higher level test management is a good candidate to move from the local tester to a cloud deployment. Web-based tools allow for viewing tester status, scheduling tests, and examining test data pushed to a cloud or server. Higher level management capability complements existing test systems built with common tools like NI LabVIEW, Microsoft .NET languages, NI TestStand, and Python. A modular test software architecture (test management, test code, measurement IP, instrument drivers, hardware abstraction layers) enables companies to evaluate the trade-offs of moving different software capabilities from local to server or cloud-based execution. As more of the test software stack moves to cloud deployments, companies will realize the benefits of cloud computing for data storage, scalable computing, and easy access to software and data from anywhere.

Taking Advantage of the IoT for Test

Leveraging the IoT for test is not a futuristic idea; it can be done today. An organization's ability to do so depends on its current automated test infrastructure and most pressing business needs. Some common areas to consider are improving test system management, increasing test equipment utilization, gaining better insight from test data, and remotely accessing shared test systems. A software-defined approach with a high degree of modularity allows businesses to focus on the areas of greatest value without having to make an all-or-nothing decision.

CONNECTED INTELLIGENCE FOR AUTOMATED TEST