Replicating Dynamic Field Environments in the Lab for More Effective Regression Testing

Erik Org, Senior Marketing Manager, Azimuth Systems

In today's complex wireless ecosystem, as our mobile devices have continued to evolve, users have come to expect continually improving functionality. Accordingly, chipset vendors, device and infrastructure OEMs, and carriers are challenged to keep up if not accelerate the pace of innovation while maintaining high product quality. Their design and development teams are pressed for rapid and efficient development -- but must still ensure that the final product meets end-user requirements.

As the software that powers today's complex wireless devices is created, updated, improved and debugged during the deployment cycles, they need to continually benchmark performance in order to ensure quality. There are several aspects of the development process which are germane to this article. First, initial quality needs to be maintained, however stochastic channel models are often inadequate in evaluating the effects of design on basic functionality and performance since issues such as network loading or device memory leakage may only emerge during dynamic field conditions. Similarly, as updates to software are implemented, QA testing is needed to ensure that not only did the new updates improve functionality and performance, but also that they did not re-expose an old problem or create a new one. In many cases failure modes are exposed only after devices and networks are stressed with field conditions. Testing with dynamic field conditions should clearly be a priority since replicating dynamic field conditions in the lab would make for more effective test however, until recently, methods to do that have not been well defined.

Regression Testing

Regression testing is a key element in software quality assurance. It is often defined as the process used to validate preexisting functionality and to improve consistency and efficiency of test cycles in product development. Regression testing focuses on situations where the software may appear to function as intended but impacts the overall performance and functionality in an unexpected way. For example, the product may perform slowly, significantly increase the network overhead or perhaps use more memory than previous versions. The primary goal of regression testing is to ensure that a change, such as a bug fix or feature enhancement, did not introduce new faults into the software.

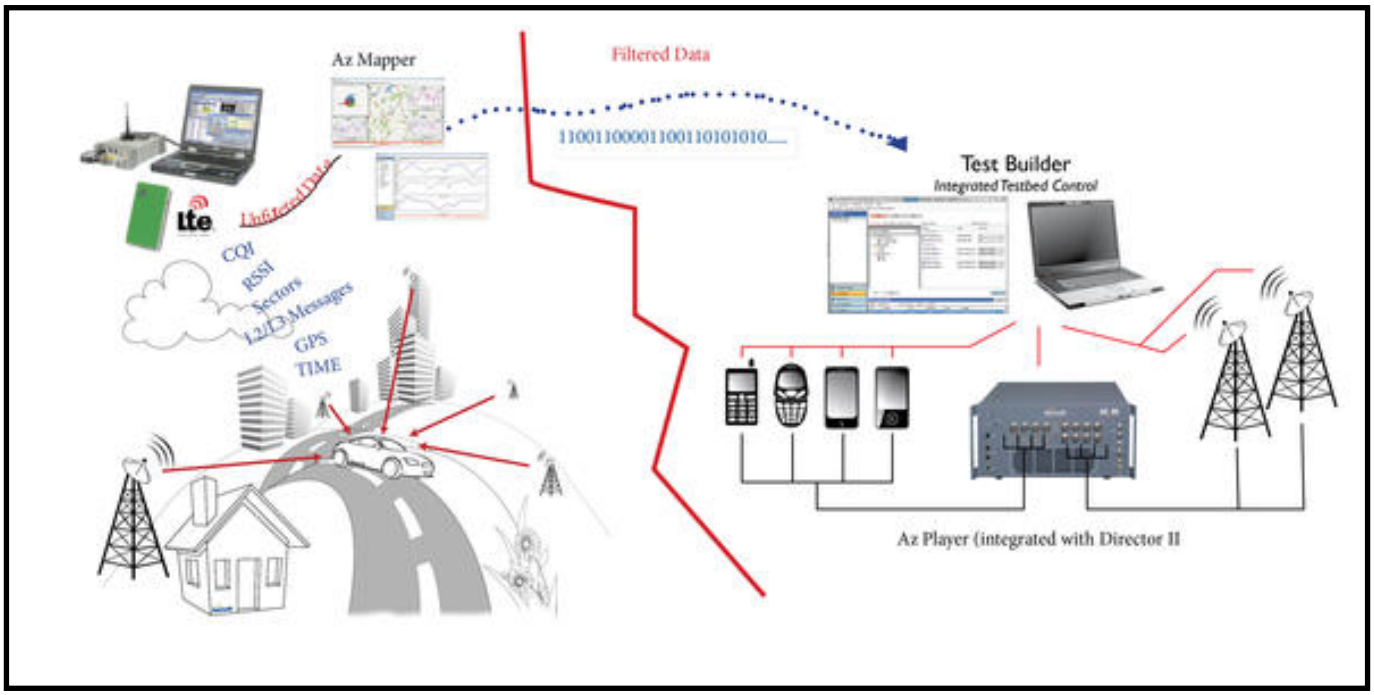
Regression testing commonly involves re-running previously conducted tests validating that no designed program behaviors have changed and that no previously fixed faults have re-emerged. When a performance issue is located and fixed, the test that exposed the bug is recorded and regularly retested after the

required changes to the program are implemented. Although this can be done through manual testing procedures using programming techniques, that process can be extremely time consuming and inefficient; thus, it is preferable to use automated testing tools.

An important consideration is that many of the performance and functionality deficiencies or bugs that regression testing attempts to uncover may only be exposed in the presence of dynamic (and stressful) field conditions. A comprehensive regression testing plan must not only iterate over tests that expose performance and functionality issues, but should do so over a range of challenging field conditions.

Lastly, it is also more effective to conduct testing before the development team has completed its work, as defects found later in the process – especially after the end-user has already encountered problems - are more costly to fix and can significantly damage a company's reputation for quality. Automated testing and well-written test cases can reduce the likelihood of additional software development or product regression test cycles as well as improve the test coverage of a regression test plan.

Using Field-to-Lab Methodologies for More Effective Regression Testing
One methodology that allows for more effective regression testing of wireless devices is Field-to-Lab, which captures the users real world experience in a controlled, repeatable and cost effective manner – and allows it to be replicated in the test lab -- so that devices and systems can be guaranteed to work as designed when delivered to the consumer.



Field-to-Lab test uses drive test data collected in the field to more accurately recreate key factors that are important for lab-based network testing and have never been previously factored into the testing. These variables include: rapid variation of signal strength, the multiple sectors visible to the user's device, and the

fluctuating speed of the device or the rate at which air effect phenomena change. Until now, these parameters have been typically ignored in lab testing.

Using this important information from drive test logs and combining this in an effective channel emulation environment offers a test solution that is closer to “real world” than ever before for existing and emerging 2G/3G/4G and LTE-Advanced technologies.

The capability to accurately predict -- in the lab -- the performance that will ultimately be experienced by a user in the field offers many benefits for regression testing in that the challenging field conditions that impact device and network performance can be comprehensively evaluated.

Field to Lab methods also enable:

· Testing against field conditions early in the design cycle to discover field related performance concerns as early as possible
· Benchmarking against the “golden routes” used by carriers to verify and validate device performance
· Sharing test conditions between partners to accelerate collaboration and reduce the time to troubleshooting or debug problems
· Performance benchmarking of possible firmware upgrades

Key Components of a Field-to-Lab Methodology

The key components of a Field-to-Lab methodology include a commercially available scanner, a tool to map and filter scanner results to a subset suitable for playback on a channel emulator, and a software tool that will replay the filtered data on the channel emulator.

After device and network performance data has been collected by commercial drive test tools, Field-to-Lab test methodologies allow the data to be conditioned, saved, and reused as many times as required with many different devices. To collect field data, Field-to-Lab solutions are integrated with both industry-standard and custom drive testing software to allow the re-use of data from drive tests. Because the test are virtual, test labs have the capability to concatenate multiple drive test logs and perform virtual field test across multiple geographies without ever starting a vehicle.

Drive test data is converted into a playback file for a given number of base stations and can be filtered, smoothed and filled to ensure realistic playback conditions. Next, the data is played back on the channel emulator; a playback control tool configures the channel emulator and streams the playback conditions to the signal path of the channel emulator between the mobile device and the base station.

This offers the ability to characterize expected performance, reproduce and verify field issues, benchmark interoperability with other devices, and optimize performance algorithms. This results in wireless devices that are more thoroughly tested before they are ever deployed in the field as well as improvements in post-deployment trouble shooting and resolution.

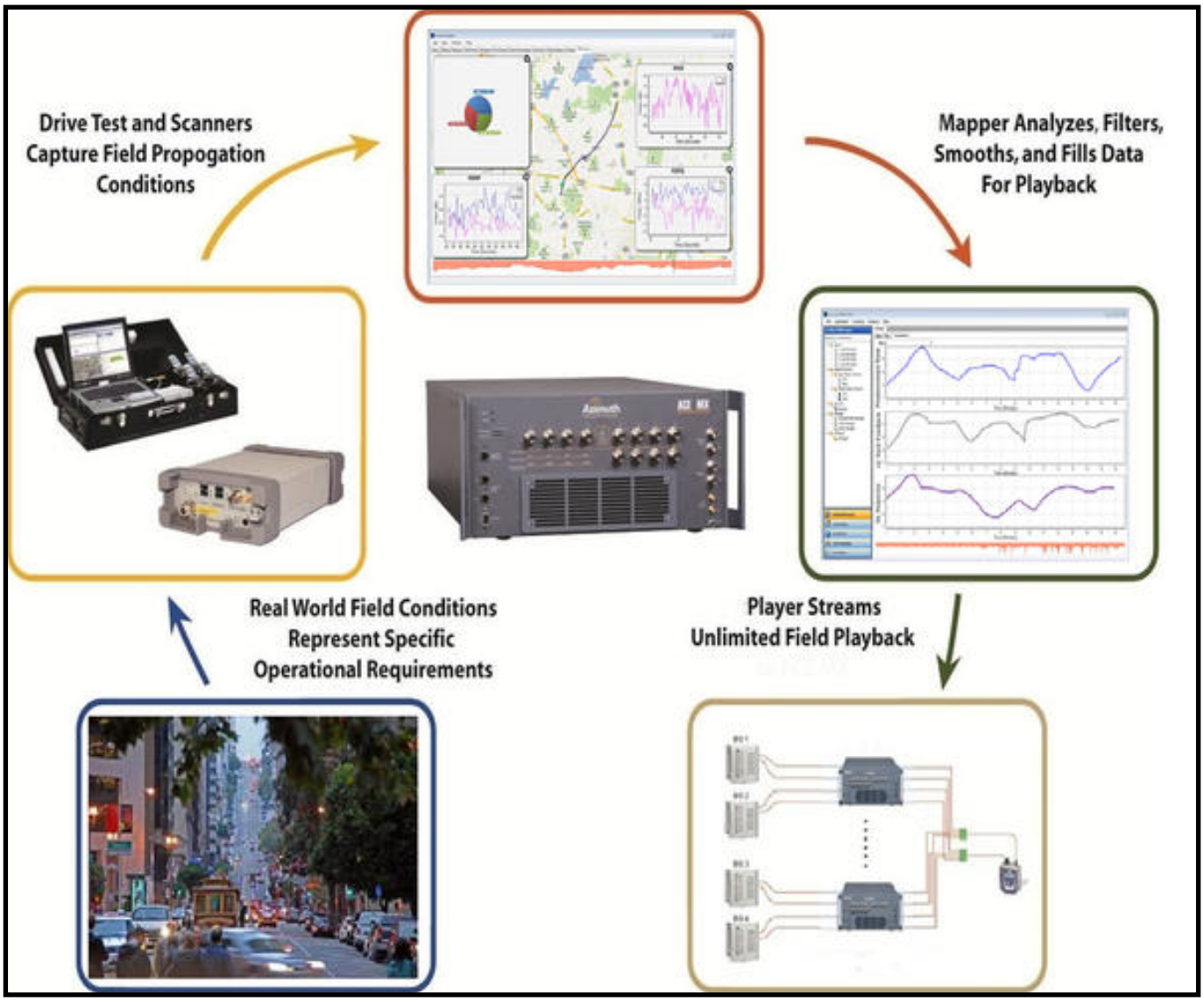


Figure 1: Mobile device testing process with Field-to-Lab

The Importance of Test Automation
Automation is extremely beneficial to effective regression testing, as test suites which contain software tools that allow the testing environment to execute all the regression test cases automatically dramatically cuts down on the time it take to deploy a successful product, and also significantly reduces the chances of human error in the process. Some projects even set up automated systems to automatically re-run all regression tests at specified intervals and report any failures, identifying when specific regression test models are outdated.

Field-to-lab testing integrates automation since most laboratory test beds are complex and have many components that need to be controlled/configured -- and also because there is a significant amount of data to analyze. Ideally, automation will reduce the need for manual intervention/operation to such a point that the user would be able to kick off a test case at the click of a single button; and have all the results presented when the execution is done.

Today, there are Field-to-Lab solutions that allow users to automate the entire

testbed in a modular, non-programming environment, enabling even novice users to write automation modules. With full end-to-end automation, the entire testing process from initializing the equipment to collecting the results is done automatically. Full automation also facilitates the setup of test suites so that a series of tests can be run one after another. The open, simple and powerful architecture of this type of automation also means that users can share it not only within their company, but also across the entire supply eco-system, helping drive consistency in the way things are tested.

Automation also plays a significant role in improving product quality as the savings allow users to do more testing, while reducing the chances of human error in the test process. Automation makes testing more affordable, increasing the number of people/amount of testing that gets done -- which is beneficial to everyone involved from the company, its ecosystem, and to consumers who ultimately receive higher-quality products in less time.

Regression Testing in Action
Customers have already seen significant benefits from using Field-to-Lab solutions for regression testing. One infrastructure vendor, prior to implementing Field-to-Lab, would conduct intensive regression testing on each firmware iteration prior to release to their operator customer. In one instance, a bug believed fixed in an earlier release was rediscovered only days prior to the scheduled release to their customer.

Shortly thereafter, they implemented Field-to-Lab for regression testing. With Field-to-Lab, they were able to take advantage of automation capabilities and were able to include a broader selection of dynamic real world conditions as they ran the regressions. In short, with automation, they are testing not only more scenarios, but also more realistic scenarios. In the months since implementing Field- to-Lab, the average number of bugs discovered per regression cycle has increased and the time for each regression test cycle has decreased.

Conclusion
Replicating dynamic field conditions in the lab can significantly improve the efficiency and effectiveness of regression testing by allowing conditions that the end-user will actually experience to be replicated and debugged in the lab at any stage of the development cycle. With Field-to-Lab, test engineers can better ensure that devices will operate as expected – and that no new or previous issues will arise. The automation environment of Field-to-Lab allows tests to be run around the clock, 365 days a year, with limited human supervision required, making the validation process much smoother and faster – ultimately delivering better quality products in less time.

October 4, 2012

Source URL (retrieved on *12/04/2013 - 7:38pm*):
http://www.wirelessdesignmag.com/blogs/2012/10/replicating-dynamic-field-environments-lab-more-effective-regression-testing?qt-most_popular=0&qt-blogs=0