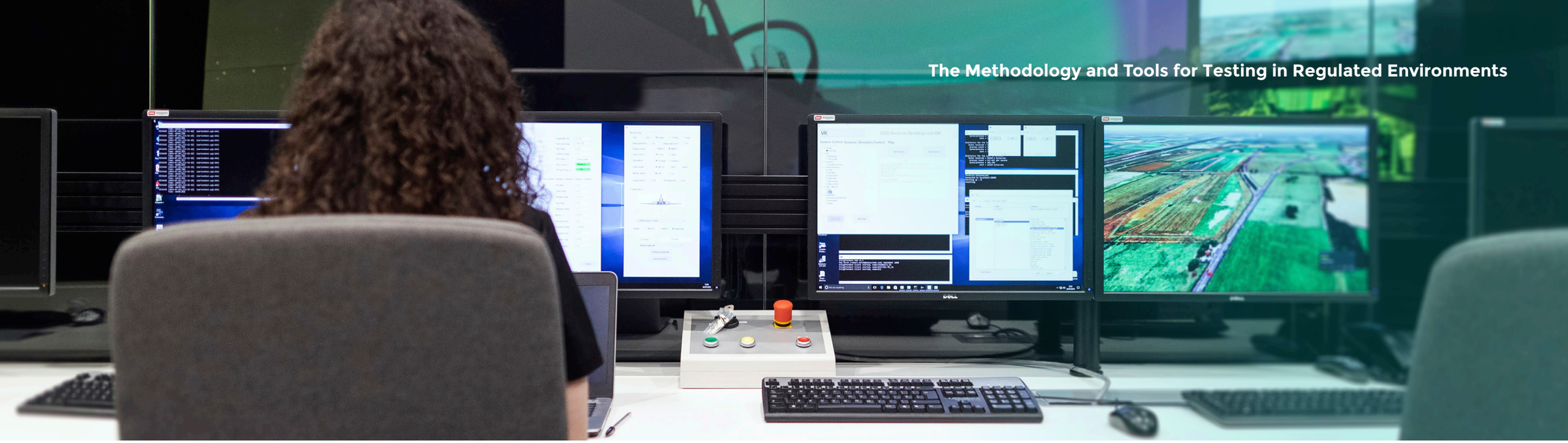
The methodology and tools for testing in regulated environments

Helping teams release software faster and more reliably

Just as in the corporate world, speed of release is a critical driver in regulated environments. DevOps creates the environment in which teams can build, test, and release software faster and more reliably. But in the quest for speed, security is in danger of being compromised. And while it's a major concern for the corporate world, it's a mission-critical one in regulated environments. The solution is DevSecOps – a culture that brings together development, operations, and security. This playbook explores the rise of DevSecOps, looks at the importance of the culture in regulated environments, and explains why the U.S. government is mandating automation as an essential tool in delivery.



Contents

Introduction

Until relatively recently, a new release of an application was a big event. These days, that's all changed.

It's changing in the corporate environment where speed of release is a critical driver in obtaining market advantage.

It's also changing in regulated environments where updates are also being released daily or even hourly.

The DevOps methodology was designed to cope with this gear change. By combining the Development and Operations functions, organizations can harness agile techniques to iterate and bring products to market more quickly.

The problems balancing speed of development requirements with quality and security requirements

Corporates are faced with a dilemma when it comes to security testing. It's software – not hardware – that defines and delivers competitive advantage.

So how far can they cut the corners of security testing to maintain speed of release to retain or gain market advantage? Making the wrong call could result in negative headlines and a loss of investor confidence.

But in regulated environments, the dilemma is much more acute. Mission critical scenarios such as space flight demand that computers and software systems have zero defects. Testers only have one opportunity to get it right.

So no matter how quickly a product moves through its development stages, pace slows once it reaches the security testing stage because corners cannot be cut.

It's a problem that's only getting worse. Systems such as weapons, command and control, spacecraft, communications, and RADAR are all running millions more lines of code than ever before. There are more vulnerabilities than ever before too. Security testing has become an enormous task.

*Testers only have **one opportunity** to get it right.*

In recent cybersecurity tests of major weapon systems the U.S. Department of Defense is developing, testers playing the role of adversary were able to take control of systems relatively easily and operate largely undetected.[1]

[1] https://www.gao.gov/products/GAO-19-128

The most advanced fighter aircraft in the world used to rely on less than three million lines of code.

Today, the F-35 runs eight million lines of code.

In the corporate world, there's a judgment call to be made around security – a balance of risk versus reward.

Regulated environments don't have that luxury. Security is not a nice-to-have. In many cases, this has hampered the adoption of DevOps practices because its methodology presents too many risks.

Yet as the speed of release in regulated environments gathers pace, the benefits of an agile methodology can no longer be ignored. Speed of release must not be compromised but neither must system integrity.

The only way to square this circle is to say security testing can no longer be a standalone part of the process. And just as development and operations have merged, so there's a growing recognition that security needs to become part of the culture too.

It's a movement that's being driven in no small part by security teams themselves. They recognize the importance of their role but also see they're in danger of being bypassed unless they can deliver at the same speed and scale as their DevOps colleagues.

Extract from the DevSecOps manifesto[2]

We have and will learn that there is simply a better way for security practitioners, like us, to operate and contribute value with less friction. We know we must adapt our ways quickly and foster innovation to ensure data security and privacy issues are not left behind because we were too slow to change.

We are therefore seeing the rise of DevSecOps – a multifunctional methodology that breaks another silo and bakes security in from the start. It's a movement that presents exciting opportunities for regulated industries and environments.

[1] https://www.devsecops.org

DevSecOps harnesses an agile methodology to solve the conundrum

DevSecOps can be seen as the new generation of secure development. It is defined as:

"The philosophy of integrating security practices within the DevOps process. DevSecOps involves creating a 'Security as Code' culture with ongoing, flexible collaboration between release engineers and security teams." [6]

Projected percentage of development teams harnessing DevSecOps practices

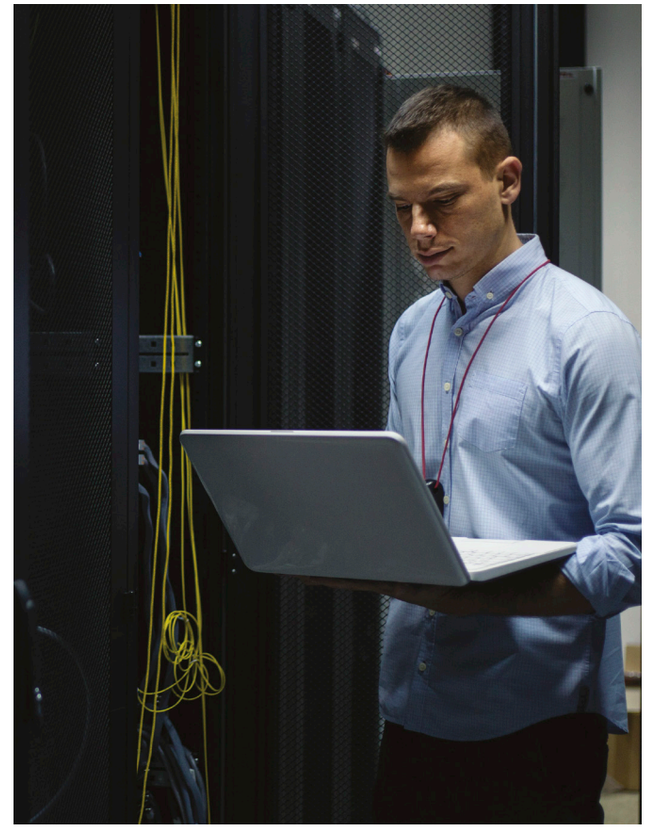
2017 - **15%** 2021 - **80%**[4]

%

[3] https://www.linkedin.com/groups/1807520/
[4] https://www.gartner.com/en/documents/3811369

The benefits of DevSecOps are clear – it improves quality, reliability and security – characteristics that are important in any environment, but especially so in regulated ones. In fact, so important is the need to adopt this Agile approach that is mandated by the U.S. Department of Defense for all defense software.

But just as DevOps required shifts in cultures and ways of working, so does the embracing of DevSecOps.



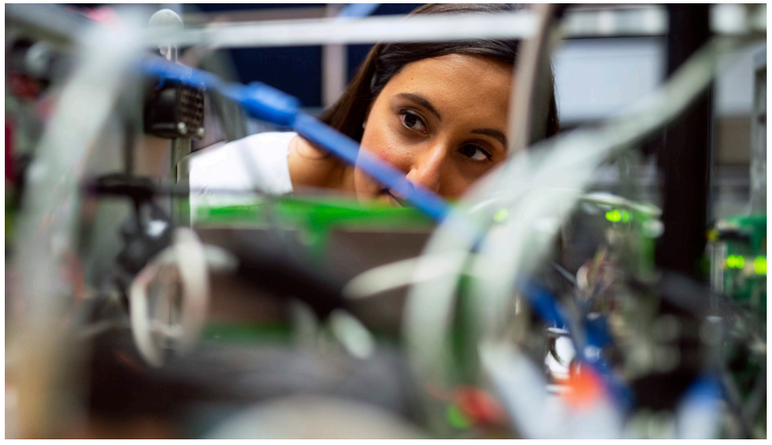
Best practice in DevSecOps

Successful adoption of DevSecOps practices involves a cultural shift. It also requires the adoption of the latest testing tools.

In terms of the cultural shift, at the most basic level, security teams need to be involved in DevOps initiatives from the very earliest stages. Security teams will have exceptional knowledge of their landscape and it is important this knowledge is respected and acted on. Developers themselves may need training in one form or another so coding with security in mind becomes second nature.

It is also important to recognize the importance of automation in the DevSecOps environment. It's widely understood to be a key tool in the DevOps environment. The need for its use becomes even more critical in DevSecOps. Quite simply, running security checks entirely manually is not an option if all the benefits of DevSecOps are to be realized because it simply takes too long and the risk of human error is too great.

Indeed, so important is the need for automated testing that it's something being highlighted at U.S. government level.



Extract from the National Defense Authorization Act for fiscal year 2020[5] SEC. 231. Digital engineering capability to automate testing and evaluation.

Digital engineering capability:

(1) IN GENERAL – The Secretary of Defense shall establish a digital engineering capability to be used:

(A) For the development and deployment of digital engineering models for use in the defense acquisition process; and

(B) **To provide testing infrastructure and software to support automated approaches for testing, evaluation, and deployment** throughout the defense acquisition process.

(2) REQUIREMENTS – The capability developed under subsection (a) shall meet the following requirements:

(A) The capability will be accessible to, and useable by, individuals throughout the Department of Defense who have responsibilities relating to capability design, development, testing, evaluation, and operation.

(B) The capability will provide for the development, validation, use, curation, and maintenance of technically accurate digital systems, models of systems, subsystems, and their components, at the appropriate level of fidelity to ensure that **test activities adequately simulate the environment in which a system will be deployed.**

(C) The capability will include **software to automate testing throughout the program life cycle,** including to satisfy developmental test requirements and operational test requirements. Such software may be developed in accordance with the authorities provided under section 800, and shall support:

(i) security testing that includes vulnerability scanning and penetration testing performed by individuals, including threat-based red team exploitations and assessments with zero-trust assumptions; and
(ii) high-confidence distribution of software to the field on a time-bound, repeatable, frequent, and iterative basis.

One other consideration must be the load that testing places on an application. In many regulated environments, a split second delay is dangerous. Testing needs to be rigorous and robust but without interfering with normal operation.

Eggplant interacts with a number of different security automation tools to bring the security insights into the DevOps world.

Eggplant's abilities are such that it was recommended by the Defense Science Board in its Design and Acquisition of Software for Defense Systems paper.[6] Eggplant works with numerous companies in the aerospace sector, including the team behind Orion, NASA's latest spacecraft, to ensure that technology will perform as expected even in harsh climates and high-stakes situations.

[5] https://docs.house.gov/billsthisweek/20191209/CRPT-116hrpt333.pdf; bold text is used to highlight the key elements of this legislation.
[6] https://apps.dtic.mil/dtic/tr/fulltext/u2/1048883.pdf

The advantages of Eggplant in regulated environments

Eggplant uses Artificial Intelligence and Machine Learning to automate testing, helping teams in regulated environments deliver DevOps at scale and fulfill legislative requirements.

Non invasive

Eggplant is the only automation platform that is completely non-invasive. This means there is no impact to the system under test and live environments are unaffected by testing. It also means we ensure the security and privacy of mission-critical technology in the testing process.

Agnostic

Eggplant is 100% agnostic to device, language, and application architecture, whether Native, Rich Client, Client Server, Mainframe, Embedded or Cloud. It means it works in any and every environment.

Highest level of interoperability

Eggplant operates in conjunction with most common tool chains including Selenium, Junit, UFT tests and most CI/CD and DevOps tool sets and industry standard protocols so it fits into most setups with ease.

*There's **no impact** to the system under test and live environments are unaffected.*

Durability

Eggplant's intelligent image and text understanding algorithms mean factors such as location, resolution, and screen size/orientation do not negatively impact tests. It gives testing a robustness teams can rely on and ensures speed of production can be maintained.

Accuracy

Testing through the eyes of the user means that our tests can 'see' and 'do' what the user can. This approach enables testers to implement various situations that might occur on board and test the performance of technology under these conditions.

Scalability

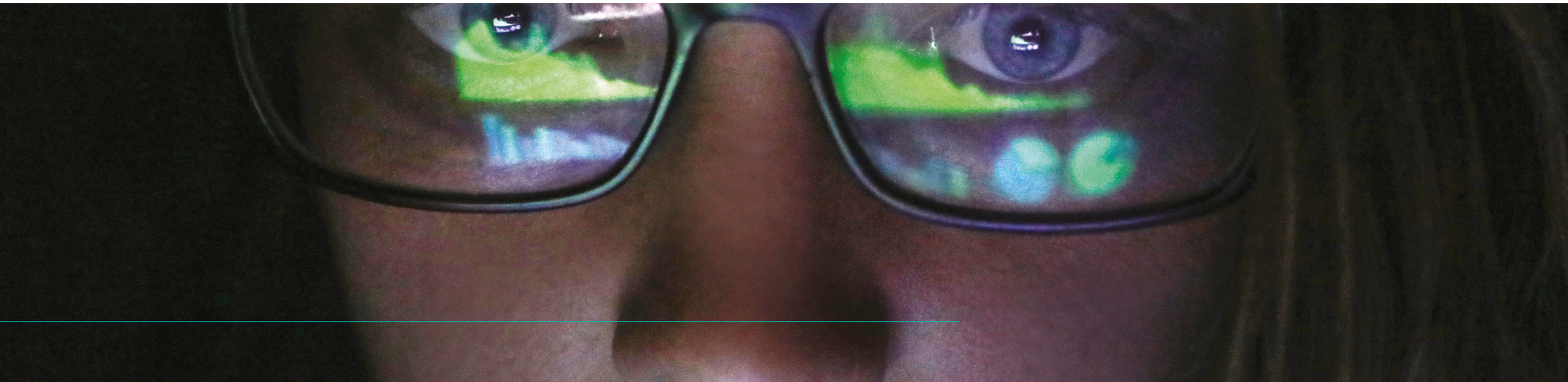
Eggplant's AI functionality and modular, extensible architecture enables intelligent test automation. It means Eggplant can test thousands of scenarios quickly and scale as technology matures.

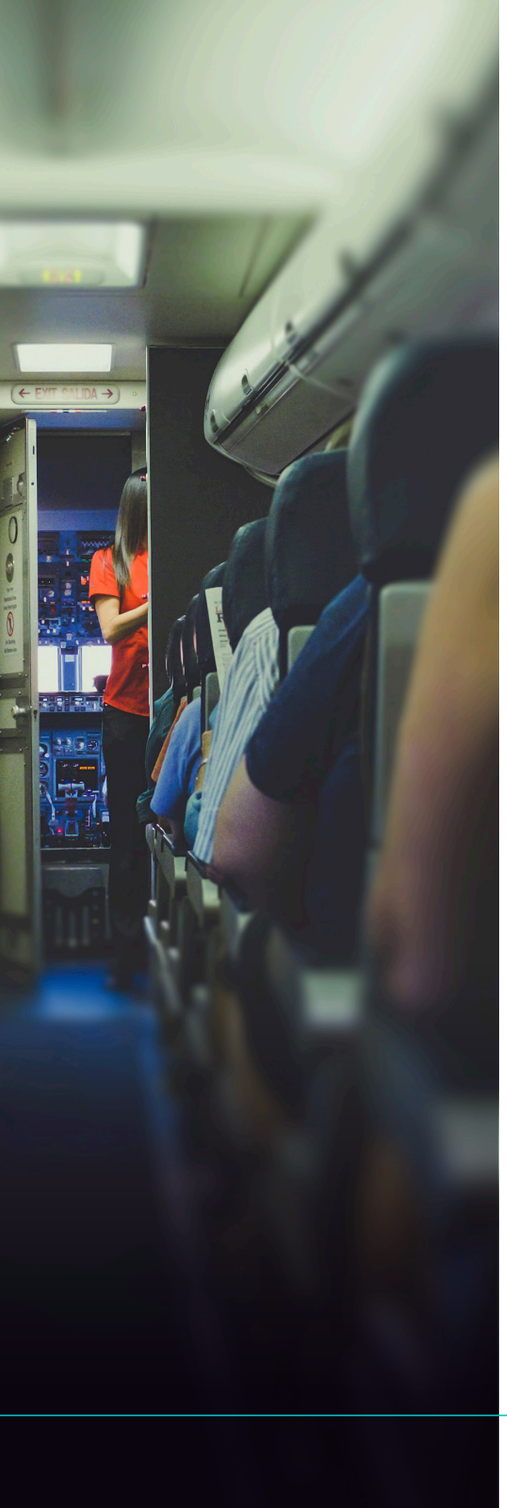
Simplicity

Eggplant uses SenseTalk, an English-like scripting language that is easy for beginners to learn, easy for experienced users to remember, and easy for everyone to read and understand. It means non-developer end-users can author tests themselves and create user journeys based on their needs, not just requirement-driven 'happy paths'.

Completeness

Eggplant is the only non-invasive platform that combines the ability to test all languages, architectures and devices with both script and model-based engines, combined with AI powered test creation, coverage analysis, and bug hunting.





Case study[7]

The Challenge: Comprehensive verification of flight attendant panels by testing GUI functionality in combination with the LRU communication

Ever since Flight Attendant Panels (FAPs) were first introduced, the number of checking and monitoring functions which are controlled via FAPs has increased. Improved touch-screen technologies are continually making the human-machine interface more efficient and convenient. Further trends, such as extending functionality by adding new software components or flexibly adapting the user interface, are increasing the complexity of testing FAPs. Therefore, the verification of FAPs requires an interdisciplinary approach. Tools from both GUI testing and LRU testing have to be combined in order to assure comprehensive verification in all development stages.

The Solution: Eggplant and CANoe deliver a simple tool combination to automate testing of FAPs

Eggplant, the software for the functional GUI test, enables testing of embedded software applications by the user interface. This approach assures greater testing depth compared to testing on the program code or functional level. Here, test automation utilizes intelligent image and text recognition algorithms to detect switch surfaces and displays. Remote control mechanisms are used to transfer screen contents and initiate user interactions. With this, there is no need to modify the testing software to achieve testability.

CANoe from Vector handles the remaining bus simulation for the LRU under test and the analysis of bus parameters. In addition, it provides a test environment that includes a test sequencer and test reporting. The tests themselves are created in the Vector tool vTESTstudio, an authoring tool for editing test flows for embedded systems.

CANoe and Eggplant are linked via XML-RPC (Extensible Markup Language Remote Procedure Call) using a DLL interface. This lets CANoe call functions and test scripts in Eggplant and read out individual results of the overall test report.

[7] https://blog.eggplantsoftware.com/case-studies/canoe-and-eggplant-functional

The Advantage:
Simple and reliable verification of FAPs

- Simplified testing through an easy-to-use interface for combining CANoe and Eggplant.
- In addition to stimulate and analyze parameters transmitted over the data bus, the integrated approach of CANoe and Eggplant gives the user a test environment in which the FAP's graphic user interface can also be monitored and stimulated.
- Test designer create the LRU tests in their familiar environment such as vTESTstudio. This eliminates the effort involved in creating additional tests.
- Functions of Eggplant are accessed in the same way as CANoe standard functions.
- Automated test execution allows shorter development cycles with frequent software releases.
- Testing is performed independently of the bus physics. Therefore, the same models can be used to test the next generation of FAPs, even if a different bus system is used.
- Even in early development stages, when the target hardware of LRUs which communicate with the FAP is still unavailable, the FAP can be tested because CANoe can be used to simulate these LRUs.

Conclusion

DevOps was the natural answer to the demands of an environment where speed of release was prioritized. But while security was still a standalone element, harnessing its full value often meant making a judgment call. This dilemma meant its full value could not be harnessed in regulated environments. The rise of DevSecOps means regulated environments can benefit from rapid software development. But as with any new methodologies, wider shifts in thinking and ways of working are required. While the cultural shifts may take time to bed in, harnessing what automation tools have to offer in the practical execution of the strategy is much easier. And in Eggplant you have a tool that is best placed to help regulated environments embrace this new way of working.

"At Eggplant you have a tool that is best placed to help regulated environments embrace this new way of working."

eggplant®

eggplantsoftware.com

For more details about Eggplant AI, please send an email to sales@eggplant.io Or, contact us in the USA +1 720 890 0211/ UK +44 20 7002 7888

About Eggplant

At Eggplant we help businesses to test, monitor and analyze their end-to-end customer experience and continuously improve their business outcomes. We provide business with award winning software such as the winner of Codie's Best DevOps tool 2019 – Eggplant Digital Automation Intelligence Suite.

Companies worldwide use Eggplant to surpass competitors, boost productivity, and delight customers. How? By dramatically enhancing the quality, responsiveness, and performance of their software applications across different interfaces, platforms, browsers, and devices – including mobile, IoT, and desktop – in agile, DevOps, and innovative application and data environments.

We are a global company serving more than 650 enterprise customers in over 30 countries. Sectors include automotive, defense and aerospace, financial services, healthcare, media and entertainment, and retail.

Eggplant is backed by The Carlyle Group (NASDAQ: CG).