galax**e**solutions™

DEVOPS IN HIGHLY REGULATED HEALTHCARE ENVIRONMENTS

Devops for Highly Regulated Healthcare Environment

Agility in IT provides a way to deal with today's constantly changing business environment. A typical DevOps philosophy to deal with this volatility is to "fail fast" using a continuous software development approach. By launching a solution and testing it in the real world, valuable feedback is obtained to incorporate in the next iteration. In life-and-death industries such as healthcare, however, failing in the real world simply isn't an option. To address this issue, GalaxE has developed a method that allows true agility in this type of regulated environment. In this white paper, we will examine GalaxE's approach to DevOps and why GxDash™ and GxMaps™ are such important tools for the healthcare industry and other highly regulated businesses.

Background

More frequent and faster releases help create better software. It is possible to quickly develop more cost-effective and higher quality products by utilizing a combination of DevOps, Continuous Integration and Continuous Delivery (CI/CD), and Agile Engineering. This approach solves the problems of insufficient test environments, late identification of errors, and overreliance on manual testing. Quality will noticeably improve as well, due to earlier defect detection in the software development cycle.

SDLC CHALLENGES

- insufficient test environments
- manual testing
- late identification of errors
- limited feedback loop

DEVOPS SOLUTIONS

- automated testing
- automated data analysis
- robust feedback loop
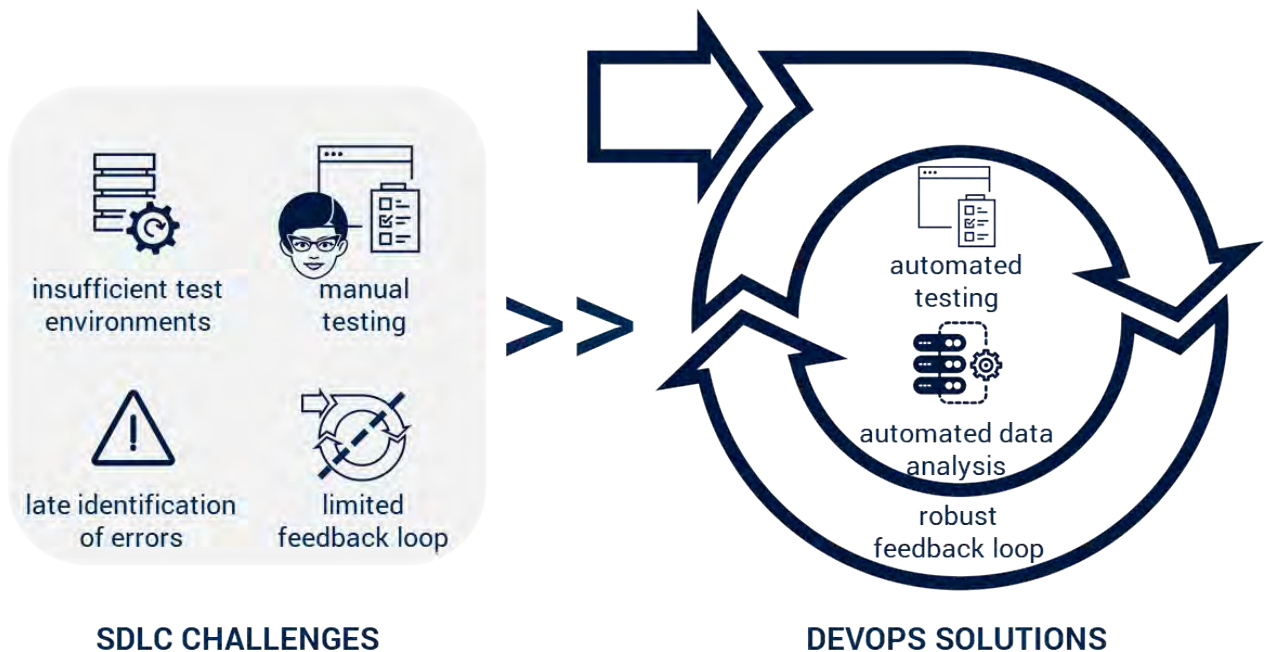
In software development, an efficient process feedback loop can mean the difference between success and failure. Feedback supplies teams with valuable information on what is or isn't working with a certain product, or what needs revision. By analyzing very large data sets via automation, it's possible to gather such feedback very rapidly without hours of human labor. Closed loop, continuous integration also enhances agile delivery in regulated systems.

It also makes sense for businesses to leverage their existing resources and subject matter expertise. Data-driven automation allows for QA and Development teams to draw on that deep knowledge to stay one step ahead in the integration process and provides more effective implementation. By establishing a customized framework for automated testing and tools, businesses will see increases in reliability and productivity, and a corresponding reduction in production errors. Additionally, all testing data produced can be saved to an extant bank to increase the quality of future analyses.

Tiers of Maturity

Looking at the tiers of maturity diagram can help an organization benchmark its current maturity level and provides a roadmap as to what steps it needs to take to reach its desired target level. Let's take a look at the Tiers of Maturity for software development.

Tier 0: Traditional Waterfall – This is the most basic form of software development. Traditional waterfall is a

DevOps Tiers of Maturity

0 Traditional Waterfall
Step by step, serial process

1 Iterative Waterfall
Mini releases, multiple releases

2 Early Reviews
Demos, process reviews, business simulations

3 Agile Team
Joint teams, all inclusive

4 Transition to Agile
Transition to the use of Agile artifacts, practices

5 Agile Practices
All artifacts and practices are Agile

6 CI/CV/CD
Test driven approach, automation in build, test, release mechanisms

7 DevOps
Integrated delivery of dev, support, operations

galax solutions

step-by-step, serial process where each phase is completed before moving on to the next phase of development. Completing each step before moving on to the next one creates easy-to-maintain recordkeeping as the organization works toward compliance. However, this approach is often more costly and less efficient than more agile approaches, since issues are typically discovered very late in the process.

Tier 1: Iterative Waterfall – Rather than creating one big software release at the end of the development process as in the traditional waterfall approach, an iterative waterfall uses multiple "mini-releases". There are still distinct phases, but they are intertwined and overlap with each other. Process decomposition and process isolation are required to protect each part of the process. As each phase is completed, it is released to make sure it is working properly. This helps reduce risk and provides incremental delivery of value.

Tier 2: Early Reviews – Demos, visualization, process reviews, and business simulations provide ways to gather valuable feedback early in the development process. These early reviews give visibility to the project, provide confirmation of value, and help engage the user community. Early reviews allow the software project team to focus on the right items and take preventative steps to improve the probability of success.

Tier 3: Agile Team – In this approach, separate agile teams are designated for each step of full integration of new software. The teams work together to ensure agility and continuity. As each dedicated team focuses on addressing issues in its part of the process, its members gain valuable expertise, reducing wait times and the need for rework.

Tier 4: Transition to Agile – As an organization makes the transition toward agile artifacts and practices, the day-to-day duties of personnel will shift. Team members will be expected to incorporate more QA measures to test the performance of their software as it is being developed, such as a focus on "Defined as Ready" (DoR) and "Defined as Done" (DoD). There may be additional costs for maintaining traceability between old and new practices and artifacts.

ITERATIVE WATERFALL "MINI RELEASE"
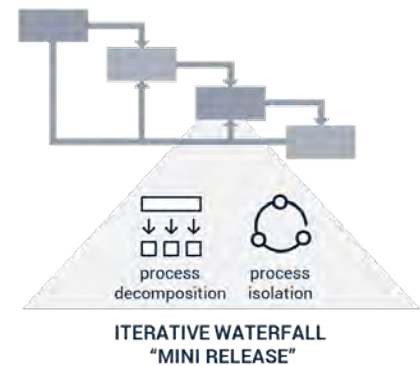
Isolating each phase in a mini-release through process decomposition and process isolation helps to reduce risk and provides incremental delivery of value.

SEPARATE AGILE TEAMS PERFORM ONGOING TESTING
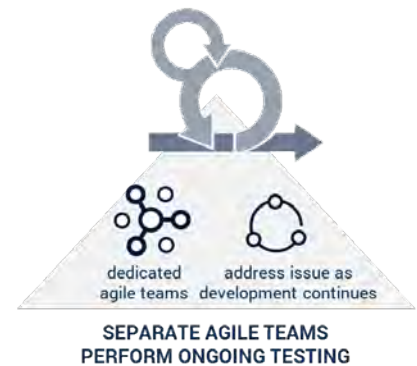
Teams focus on reducing wait times and the need for rework by testing throughout the SDLC.

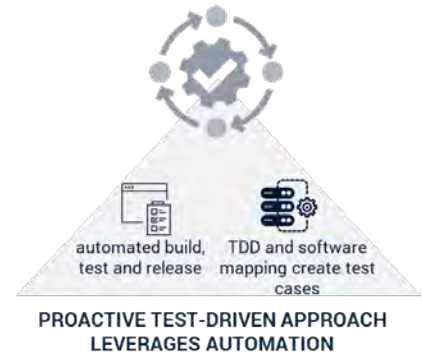**AGILE TRANSITION REQUIRES TRACEABILITY**
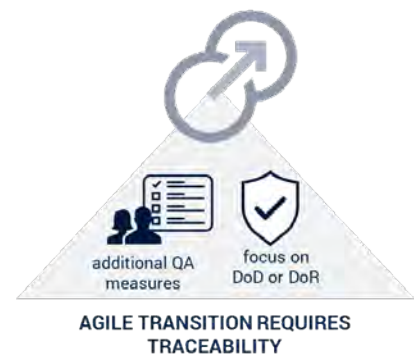
As teams transition to Agile, it is important to maintain traceability between old and new processes and artifacts

Tier 5: Agile Practices – The team completes the transition of any legacy artifacts and processes until they are all agile. The development team works in close collaboration with business stakeholders to deliver frequent, usable versions of the product. Teams use knowledge they've acquired through processes and artifacts to fix mistakes they've found and to build on the previous version.

Tier 6: CI/CV/CD (Continuous Integration/Continuous Validation/Continuous Development) – This is a proactive test-driven approach that uses automation in build, test, and release mechanisms to ensure continuity and accuracy. Process tools such as test-driven development (TDD) and software mapping are used to create a series of specific test cases. Each step is integrated into the software, validated, and then used to develop the next iteration.

Tier 7: DevOps – DevOps is a software development process that attempts to unify software development (Dev) and software operation (Ops) to create a fully integrated software delivery process. DevOps emphasizes automation and monitoring during all steps of software development, from integration through validation and deployment. In this way, DevOps aims to create shorter development cycles, increase deployment frequency, and deliver more dependable releases to meet business objectives. The financial model that supports DevOps combines capital and operational expenses, uses organizational shifts to manage, and reports on a unified delivery structure.

automated build, test and release | **TDD and software mapping create test cases**

PROACTIVE TEST-DRIVEN APPROACH LEVERAGES AUTOMATION

Automation plays a key role in reaching this tier in your DevOps maturity journey. Large data sets are used to test every step quickly and with zero tolerance for error

automation and monitoring | **increased accurate deployment**

**CREATE SHORTER DEVELOPMENT LIFE CYCLES WITH INCREASED ACCURACY**
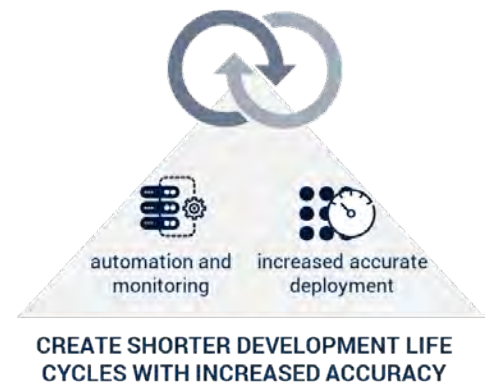
Reporting on a unified delivery structure, DevOps combines capital and operational expenses and allows your organization to fully realize its technology potential.

GalaxE Provides DevOps Assets for Regulated Environments

In order to be truly effective, DevOps relies on Continuous Delivery (CD), which, in turn, requires Continuous Integration (CI). The release of incremental software changes needs not only automated release mechanisms, but also testing, code quality, security and privacy controls, and validation steps. For organizations in a highly regulated environment such as healthcare, however, validation must be taken a step further through the process of Continuous Validation (CV). Healthcare deals with people's lives, so this environment must be continually tested and validated within a 0% tolerance.

GalaxE is the only technology services firm that offers automated mapping, test design, and execution for CD/CI/CV through our enhanced tools, GxMaps™ and GxDash™. GxMaps™ is an automated dependency mapping tool that quickly identifies and assesses the impact of programming changes on people, processes, and systems. GxDash™ offers centralized command and control to

provide holistic information and analysis to aid in decision making.
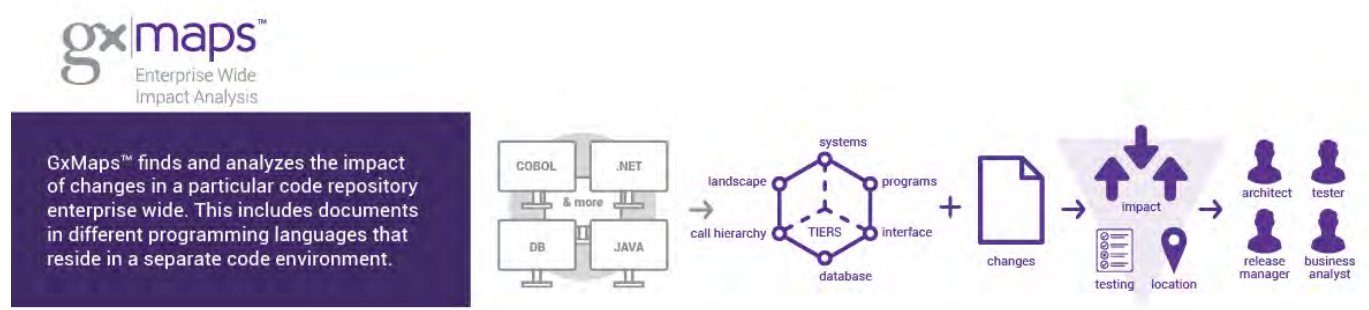
Benefits and features of GxMaps™ include:

- Automated parsing and building of dependency maps in a database
- Automated impact report generation for each of SDLC e.g. requirements, code, and QA
- Integration with tools such as McCabe to generate test cases and provide objective data on code coverage
- Provides different views for business analysts, development team, and QA
- Provides impact in terms of areas such as Medicare and Medicaid
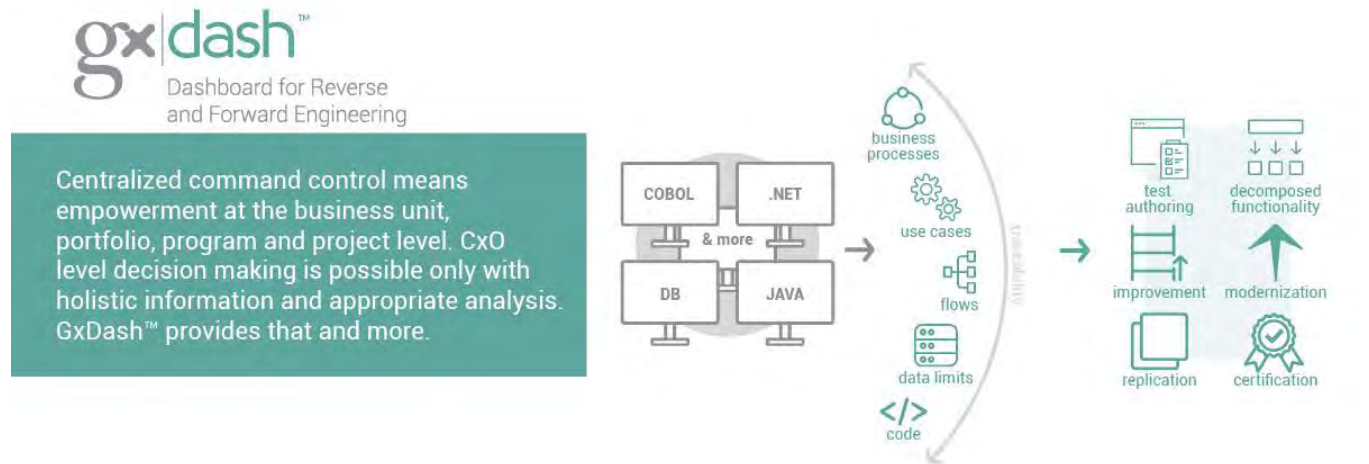
Benefits and features of GxDash™ include:

- Implementing an enterprise dashboard for all technology tiers (6-12 weeks)
- Provides automation for building dependency maps
- Provides automation for re-platforming, upgrades, and consolidation
- Predictive analysis, code, and test case generation

Only GalaxE has the necessary tools to help automate both User Access Testing (UAT), also known as Blackbox Testing, and System Test Pipelines, or Whitebox Testing. Maps generated by the cross-platform reverse engineering tools in GxMaps™ and GxDash™ provide test cases that represent real world

activities, such as when a user would click a button on the screen. These tools also automatically record system behavior, via internal branching of logic, to provide system and integration test cases. Automatic generation of test data makes this process fast, practical, and very real.
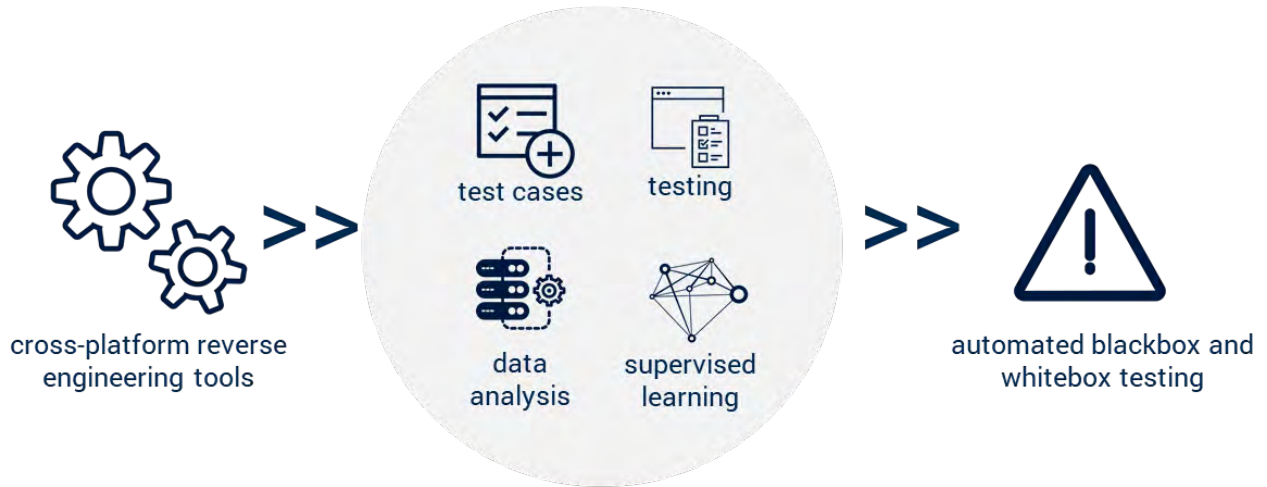
AUTOMATION GENERATION OF TEST DATA MAKES THIS PROCESS FAST, PRACTICAL AND VERY REAL

In addition, GalaxE's mapping tool sets have now started using Support Vector Machine (SVM) techniques to classify branches of code by their architectural significance, regulatory impacts, and user behavior. An SVM is a supervised learning model with associated learning algorithms that helps analyze the data used for classification and regression analysis. This type of advanced machine learning helps reduce noise in the data.

To capitalize on the power of DevOps using massive automation, supporting organization structures need enhancements. System Development Life Cycle (SDLC) provides navigation for a risk-based approach. There are many instances of these capabilities not only enabling continuous delivery of value with mitigated risk but also eliminating run costs of platforms maintained by dedicated DevOps teams.

Conclusion

The only way to perform DevOps in a regulated environment like the healthcare industry is through the use of AI/automation. GalaxE is this solution. GalaxE works with some of the largest corporations in the world, achieving superior results with lower costs, quicker delivery times, higher accuracy and increased reliability. Our healthcare accelerator solutions and software development lifecycle options meet regulatory and compliance requirements while achieving speed to market, accuracy, and agility.

WHAT ARE THE BENEFITS TO GALAXE'S TIERS OF MATURITY?

1 Traditional Waterfall
Step by step, serial process

Benefits
Easy to maintain records towards compliance

Requires
Experienced Team of IT Professionals

2 Iterative Waterfall
Mini releases, multiple releases

Benefits
Reduction of risk, incremental delivery of value

Requires
Process decomposition, process isolation

3 Early Reviews
Demos, process reviews, business simulations

Benefits
Early visibility and confirmation of value, engagement of user community

Requires
Visualization, demos

4 Agile Team
Joint teams, all inclusive

Benefits
Focused attention of dedicated team, reduction in rework and wait times

Requires
Dedicated team

5 Transition to Agile
Transition to the use of Agile artifacts, practices

Benefits
Traceability maintained between old and new practices and artifacts

Requires
Additional costs for maintaining traceability, focus on DoR and DoD

6 Agile Practices
All artifacts and practices are Agile

Benefits
Lower cost, faster delivery

Requires
Transition of any legacy artifacts and processes

7 CI/CV/CD
Test driven approach, automation in build, test, release mechanisms

Benefits
Continuous delivery while maintaining continuous validation

Requires
CI/CV/CD, TDD, SoftwareMapping, validation within iterations

8 DevOps
Integrated delivery of dev, support, operations

Benefits
Right sized production lines supporting value delivery, opportunity to eliminate run expenses on typical platforms

Requires
Financial model that supports combined capital and operational expenses, organizational shifts to manage and report on a unified delivery structure

galax e solutions

To learn more, please visit: https://galaxe.com/solutions/