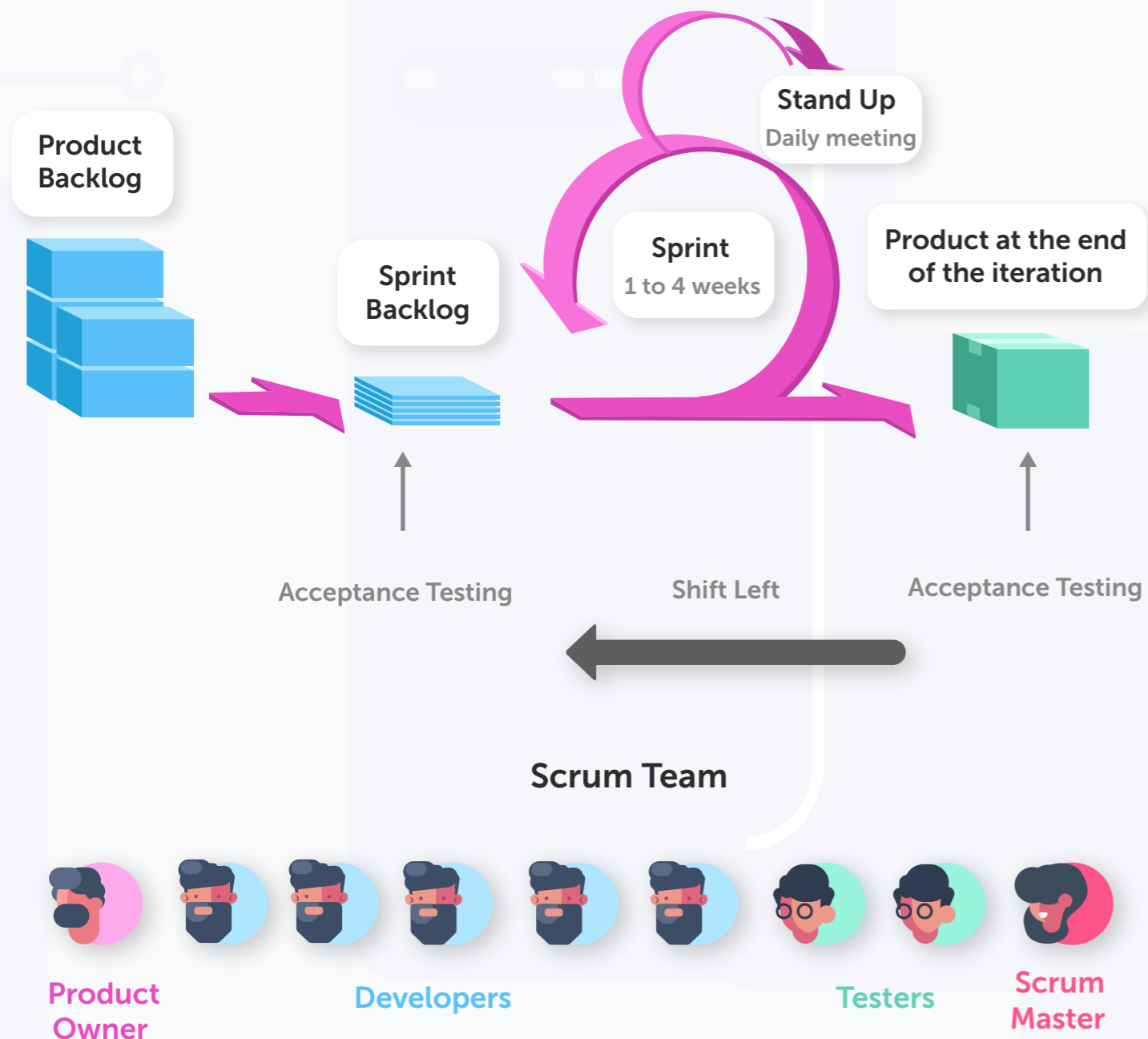




# DevOps Shifting Left Using Continuous QA

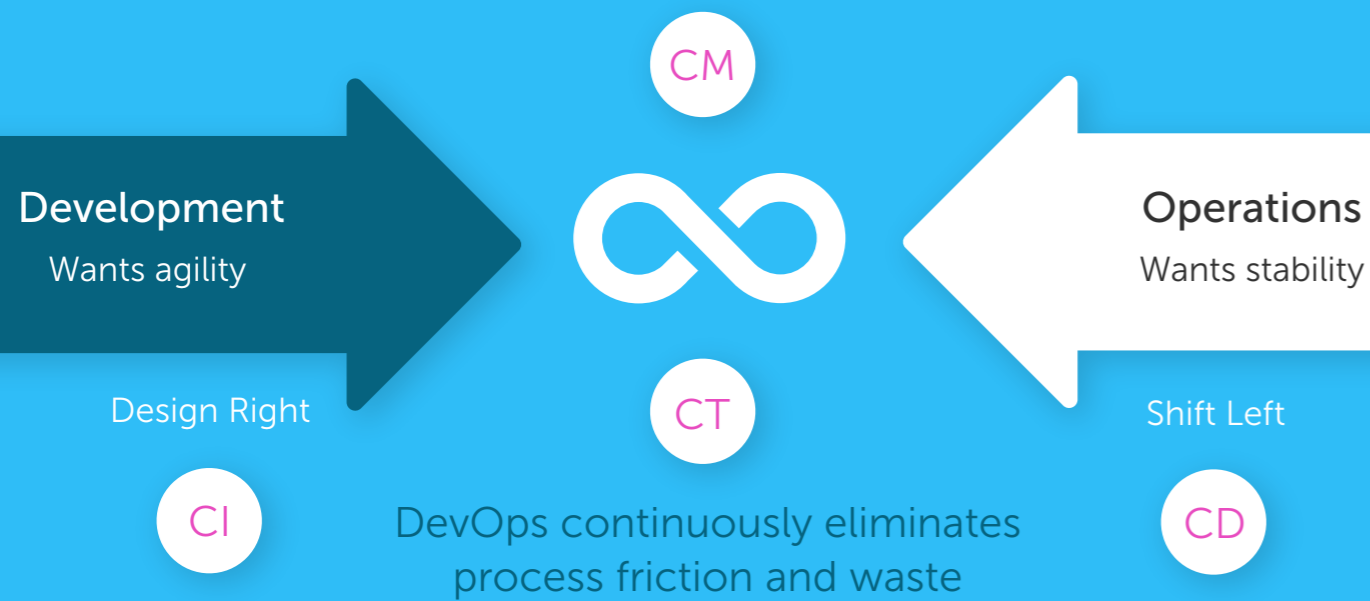


## Table of Content

- Is DevOps right for you?
- Shifting left and Continuous QA
- Business strategies for Continuous QA
- Continuous quality strategies
- Test acceleration strategies
- Steps to transform your enterprise
- Investment, Benefits and ROI
- Summary

# Is DevOps Right for You?

Your need for faster software delivery is the primary reason for DevOps, but there are notable other reasons why you may need DevOps. As indicated in State of DevOps Report, DevOps, when implemented according to best practices, delivers major and simultaneous benefits of dramatically improved agility, stability, efficiency, quality, security and employee satisfaction. If you have needs for any combination of these benefits then you should consider DevOps. If you need all of them then DevOps is compelling.



To be successful with DevOps, you will need to commit to a range of transformations that span people, process and technologies. While there are no formal DevOps standards, DevOps embodies best practices based on lean principles applied to software development and delivery with the purpose to continuously eliminate process friction and waste.

There are nine categories of best practices that apply and need to be mastered for effective DevOps: leadership, collaborative culture, modular design, continuous integration (CI), continuous testing (CT), security, continuous monitoring (CM), elastic infrastructure, and continuous delivery and deployment (CD).

DevOps is probably right for you if you have a need for the benefits that DevOps can deliver, and you have the means to implement best practices suited for transformations specific for your needs.

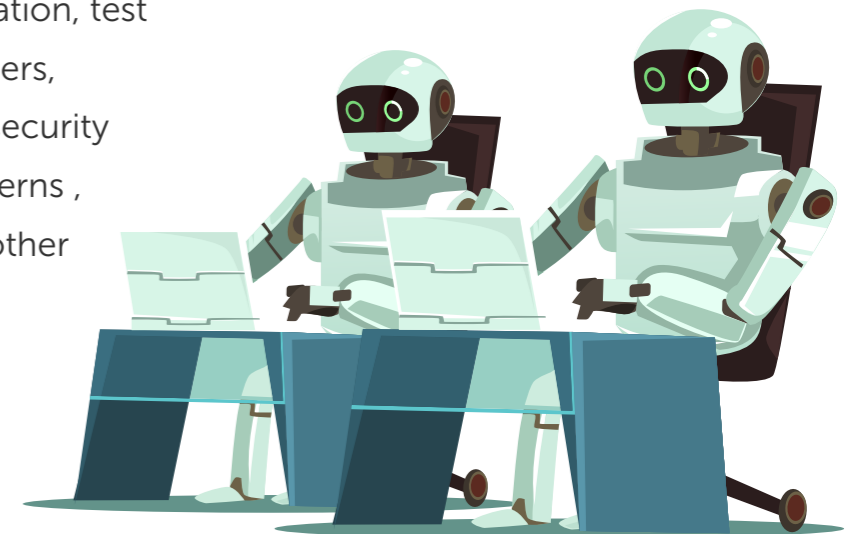
# Shifting Left and Continuous QA

“Shift Left” is a critical tenet of DevOps best practices. With DevOps, software development, testing and delivery tasks are arranged to be completed early in the pipeline of tasks from conception through deployment. In this way the lead time for delivery is shortened. Doing this is easier said than done. In traditional environments speed and quality are enemies of each other. It is the strategic application of DevOps best practices that enable continuous quality assurance as the speed of deployments accelerate.

While much has been written about Continuous Integration (CI), Continuous Delivery and Deployment (CD), it is continuous quality assurance and testing (CT) that is the heart of every successful DevOps implementation. After-all it is testing that produces test result and data such as verdicts critical for assessing the quality of artifacts from each stage in the pipeline from design, integration, build, packaging, delivery and deployment processes. **Without results data from continuous testing there would be no basis for promoting a software change to the next stage in the continuous delivery pipeline.**

To “Shift Left” quality assurance for DevOps requires a re-engineering of traditional QA practices. **QA teams working independently and running tests after the development hands over the changes does not work with DevOps.** Instead, with continuous QA many of the testing and QA tasks are pushed earlier.

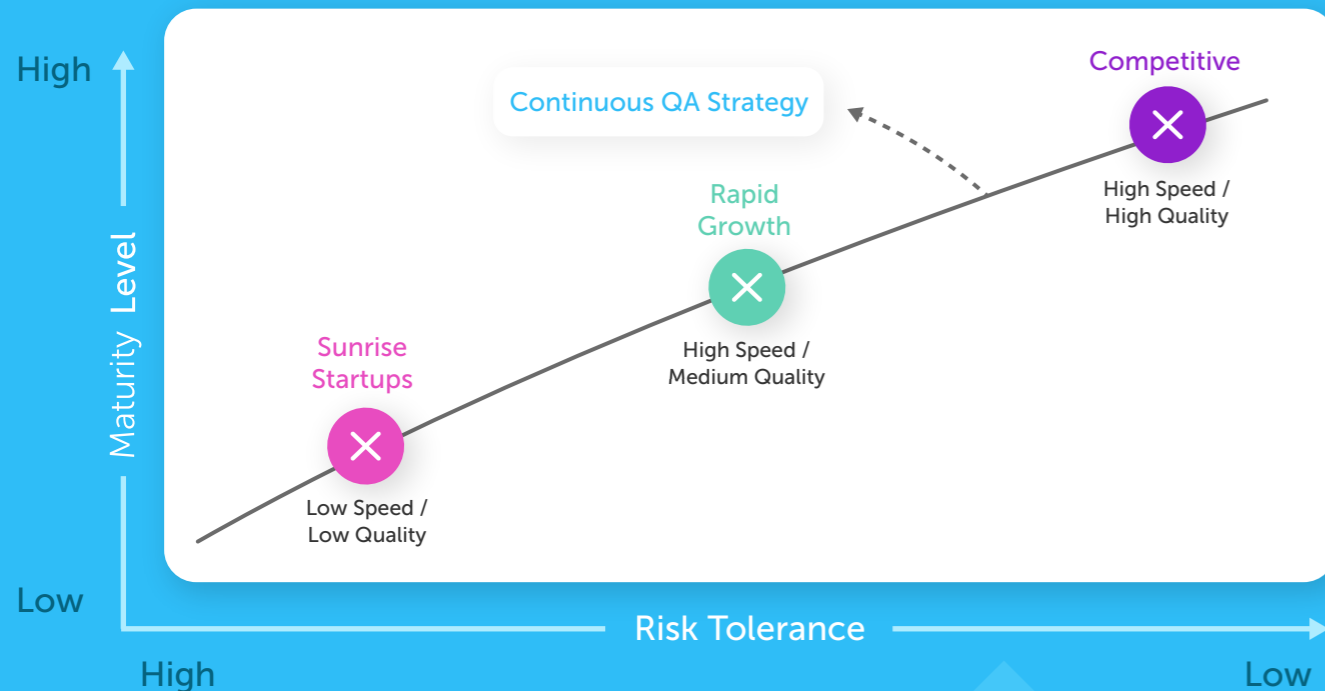
Effective Continuous QA requires understanding and application of best practices for DevOps. Practices apply to test culture, continuous test strategies, end-to-end test integration, DevOps test infrastructures, DevOps test frameworks, DevOps-ready test tools, test creation, test acceleration, test analytics, microservices and containers, database DevOps testing, DevOps security testing, continuous testing anti-patterns, continuous test management, and other advanced continuous test topics.



# Business Strategies for Continuous QA

Given the many variables and choices for Continuous QA practices listed on the prior slide, the best route for your DevOps continuous QA journey may not be obvious. Unfortunately some routes will lead to dead ends, the need to back-track, and costly failures and setbacks. A good strategy, that best suits your business is critical. Depending on your business some tactics may not be important, but others are critical. **A good strategy considers where your business is now, in terms of maturity (sunrise startup, rapid growth or competitive) and risk tolerance.**

If a quality failure affects the health of your clients then it is least tolerant to risk. For example life or death may be the result of a software failure. On the other hand if a failure only causes a temporary effect on the business itself or is contained to a specific application then it is more tolerant to risk.



Once you determine the maturity and risk tolerance, your business strategy for continuous QA can be determined by the speed of software change and level of quality assurance needed. For example - A highly competitive mature business has low tolerance for risk. This means it would need higher speed of delivery and high quality level at the same time. On the other hand the continuous QA strategy for a sunrise startup with higher risk tolerance can suffice with practices that support a lower level of continuous speed and quality.

# Continuous Quality Strategies

With a DevOps shift left approach, quality is directly correlated to mean-time-to-recover (MTTR) instead of Mean-Time-Between-Failure (MTBF) measures. Software releases generally have a short life. **The right strategy for your organization is determined by the time available to detect and deploy repairs.**

More sophisticated continuous QA practices for people, process and tech are needed when MTTR times are shorter.

Quality (MTTR)	✓ Days	✓✓ Minutes	✓✓✓ Seconds
People	Cross-Functional SLAs	QA integrated with Dev	QA & Ops integrated with Dev
Process	Dashboards	Automated reversals	Automated remediation
Tech	50% Automated	>80% Automated	100% Automated

- **If MTTR is measured in days**, then it may be sufficient to operate with Service Level Agreements(SLAs) between cross-functional Dev, QA and Ops teams. Dashboard displays of test progress and test results may be adequate to communicate test information. It can suffice to automate only half of the tests provided manual tests can be completed within a release schedule.
- **If MTTR is measured in minutes**, then shorter more efficient communication links are needed between Dev and QA. There is not enough time to process manual SLAs when failures occur. Automated testing (e.g. 80%) and reversal of changes are required.
- **If MTTR is measured in seconds**, then communication between Dev, QA and Ops must be extremely efficient and tightly integrated. Failures may need to be resolved by smart synthetic tools when people can not keep up with the pace of change. All tests need to be fully automated.

# Test Speedup Strategies

With a DevOps shift left approach, the time to conduct the entire range of tests for a software change must fit with the time target for the end-to-end continuous delivery pipeline. **More sophisticated continuous QA practices for people, process and tech are needed when pipeline times are shorter.**

Quality (MTTR)	✓ Weeks	✓✓ Days	✓✓✓ Hours
People	DevOps test training	DevOps test architect	DevOps test leadership team
Process	Test standards	Test creation tools	Smart test creation and analysis
Tech	Toolchain	Test tool frameworks	Release Automation

- **If the pipeline time is measured in days,** then it may be sufficient for people in the cross-functional Dev, QA and Ops teams to be trained on test driven development (TDD) and test scripting. Test standards can be used together with test tools integrated into the continuous toolchain.
- **If pipeline time is measured in minutes,** then more sophisticated and dynamic co-ordination of testing activities are needed. A test architect should be assigned to inject intelligence into test selection and result analysis. To keep up with the faster rate of testing more sophisticated test creation tools supporting model based or behavior based test methods are needed. Test execution may require a test tool framework with sophisticated test orchestration and automation methods integrated with the continuous tool chain
- **If the pipeline time is measured in seconds,** then testing requires a strategic test leadership team to ensure optimal testing strategies are employed at all times. The smartest methods and tools for test creation and analysis are critical. Test monitoring and failure remediation is fully integrated in the pipeline and automated.

# Steps To Transform Your Enterprise

While each DevOps journey is unique for each business, there are some logical steps for structuring continuous QA transformation projects. Following these steps will ensure measured successful outcomes.

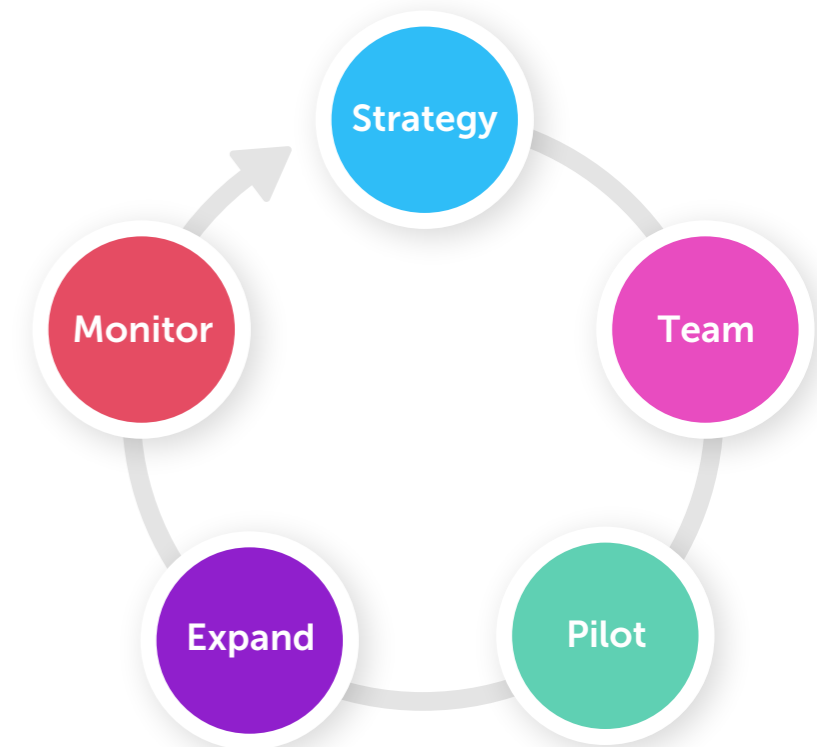
**1 Define your continuous QA strategy.** Consider engaging a senior DevOps consultant who has a track record in continuous QA to conduct a strategic DevOps QA assessment. Together with your cross-functional team decide the tactics that best fits your goals, business maturity, risk tolerance, quality and speed requirements. Define success metrics that match your goals. Sample metrics can be coverage, time-to-test, remediation rates, for example.

**2 Prepare the team.** Determine the team and organization structure to lead, implement and measure progress. Leaders and team members need DevOps test training according to the role they play.

**3 Execute a pilot project.** Pick an application to trial the strategy. Determine the scope and success measures for the pilot. Implement the tool chain and create tests. Measure results, conduct a retrospective and adjust the strategy according to lessons learned.

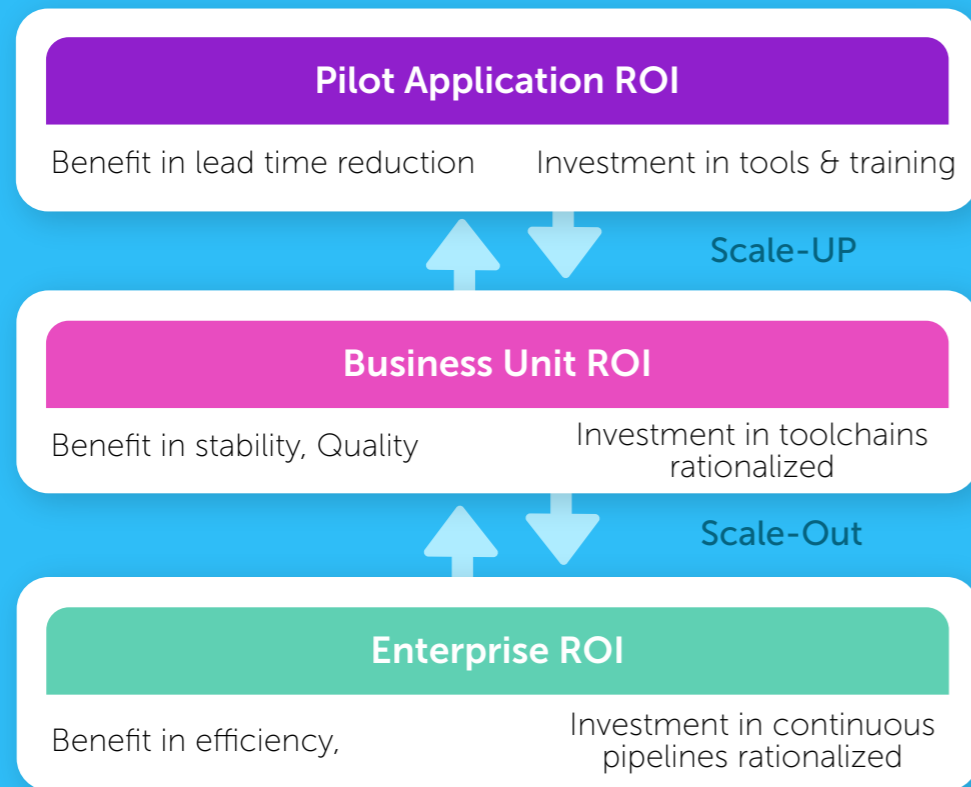
**4 Expand the implementation.** Expand the implementation to support other applications. Implement the expanded tool chain and create tests for each application. Measure results, conduct retrospectives and adjust the strategy according to lessons learned.

**5 Monitor progress.** Once the solution has been deployed to all applications continue to measure results, conduct retrospectives and adjust the strategy and solution according to lessons learned.



# Investments, Benefits and ROI

DevOps continuous QA requires initial and ongoing investments. Benefits must be visible to all stakeholders for them to buy-in and commit to the efforts required to build and sustain the implementation. Return-on-Investment (ROI) is required for business leaders to justify investments at all phases of the implementation lifecycle.



- **Pilot Application ROI.** Typically continuous QA pilot projects are justified by financial savings. Team effort savings result from reduced lead time for testing an application compared to the costs of incremental training and tooling costs for the application.
- **Business Unit ROI.** After a successful pilot, investments to **scale-up the implementation** to cover more applications are justified by larger scale team and other benefit factors such as savings from improved stability and quality metrics derived across the business.
- **Enterprise ROI.** As the enterprise recognizes the ROI of a business that has implemented continuous QA it makes sense to scale-out the implementation to other businesses in the enterprise. At this level overall enterprise efficiencies and employee satisfaction become a major benefit to help justify investment in enterprise-wide continuous QA.

## Summary

- DevOps is right for you if you need the benefits of DevOps and have the means to implement best practices.
- Shifting left and Continuous QA is a critical requirement for DevOps. A high % (E.g. 80%) of test automation is a key ingredient.
- Business strategies for Continuous QA are determined by considering business maturity and risk factors.
- Continuous quality strategies are determined according to MTTR requirements.
- Speed strategies are determined according to deployment lead time requirements.
- Logical steps are defined to transform your enterprise to continuous QA. Benefits and ROI must be measured to justify initial and ongoing investments in continuous QA.

# About the Authors



Tanya Kravtsov

Tanya Kravtsov is a director of QA at Audible, helping build a new QA organization to support innovative web product development at scale. Previously, as head of automation and continuous delivery at ROKITT, senior QA manager at Syncsort, and VP at Morgan Stanley, Tanya focused on quality, automation, and DevOps practices, working with internal and external customers to transform development and testing processes. Tanya is passionate about process automation, continuous integration, and continuous delivery. She is a founder of the DevOpsQA NJ Meetup group and a frequent speaker at STAREAST, QUEST, and other conferences and events. Follow her on Twitter @DevOpsQA.



Bob Crews

Bob Crews, President and Co-founder of Checkpoint Technologies, is a consultant and trainer with 29 years of IT experience in full life-cycle development and software testing. Bob and his organization provide services and solutions focused on QA with a concentration in functional, performance and security testing. He has assisted organizations such as Harvard University, Raymond James, the FBI, J.P. Morgan Chase, and Department of Veterans Affairs in developing teams, processes and solutions to improve the quality of their applications and systems. Bob has consulted for

# Thank You!

**testim**

[www.testim.io](http://www.testim.io)



## Learn More about DevOps Shift Left with Continuous QA

### Industry References

1. State of DevOps Report by Puppet Labs, IT Revolution, DORA | June 2017
2. The Phoenix Project by Gene Kim, Kevin Behr, George Spafford | January 2013
3. The DevOps Handbook by Gene Kim, Patrick Debois, Jez Humble, John Willis | October 2016
4. Out of the crisis, W. Edwards Deming| 1986
5. From QA to Continuous Testing, Marc Hornbeek| DevOps.com | February 2015
6. DevOps – it’s all about Continuous Testing, Marc Hornbeek| DevOps.com | February 2015
7. Seven Best Practices for Accelerating Continuous Testing, Marc Hornbeek| DOES 2015 |November 2015
8. DevOps Impact on Testing and Culture, Shirly Ronen-Harel| August 2015
9. Continuous Testing Accelerated, Marc Hornbeek| DevOps.com | March 2015
10. Test Results Analysis at the Speed of DevOps, Marc Hornbeek| DevOps.com | March 2015
11. DevOps Makes Security Assurance Affordable, Marc Hornbeek| DevOps.com | April 2015
12. Isn’t Continuous Testing a Longevity Test Anti-pattern?, Marc Hornbeek| DevOps.com | March 2015
13. Continuous Testing Stability, Marc Hornbeek| DevOps.com | April 2015
14. World Quality Report, 9th Edition, | 2017-2018
15. The DevOps Institute [www.devopsinstitute.com](http://www.devopsinstitute.com)
16. [www.DevOps.com](http://www.DevOps.com)
17. The Manager’s Guide to Continuous Delivery | Xebialabs| February 2014
18. The DevOps Adoption Playbook | IBM Press| Sanjeev Sharma| 2017