

testhouse



WHITE PAPER

SHIFT – LEFT PERFORMANCE TESTING

www.testhouse.net

Executive Summary

The shift-left performance testing approach is revolutionizing the way organizations ensure the optimal performance of their software systems. By testing early and testing often, this proactive approach allows developers to address performance issues before they impact end-users, leading to improved software quality, enhanced user experiences, and faster time to market.

This white paper explores the concept of shift-left performance testing and its benefits, challenges, and best practices for implementation. It begins by defining shift-left performance testing as a proactive testing approach conducted as part of the daily development flow. By integrating performance testing activities early in the development process, organizations can identify and address performance bottlenecks, achieve cost and time savings, and enhance collaboration between development and testing teams.

The paper delves into the approaches for implementing shift-left performance testing, including unit testing, continuous integration and delivery (CI/CD) integration, profiling, performance monitoring, and load generation. It highlights the importance of incorporating these approaches into the software development life cycle to detect performance issues at an early stage.

While the benefits of shift-left performance testing are substantial, there are challenges to overcome. The white paper addresses these challenges, such as shifting organizational culture, acquiring specialized skills, dealing with limited availability of non-functional requirements, and integrating performance testing into Agile and DevOps workflows.

To successfully implement shift-left performance testing, organizations are encouraged to follow best practices. These include conducting training and knowledge sharing sessions, defining clear performance objectives, automating performance tests through pipeline integration, simulating realistic load conditions, testing in appropriate environments, and prioritizing continuous improvement and innovation.

This white paper empowers organizations to embrace the shift-left performance testing approach and unlock the potential for delivering high-quality software with superior performance. By adopting these strategies and best practices, organizations can create a competitive advantage, delight customers with exceptional user experiences, and achieve success in today's dynamic digital landscape.

Table of Contents

Introduction.....	3
Why read this white paper?.....	4
What is shift-left performance testing?	5
Why adopt shift-left continuous performance testing?	6
What are the major benefits of the shift-left performance testing approach?	8
What are some of the challenges implementing shift-left performance testing?.....	11
What are the "Best Practices" for Implementing Shift-Left Performance Testing?.....	13
Conclusion.....	15

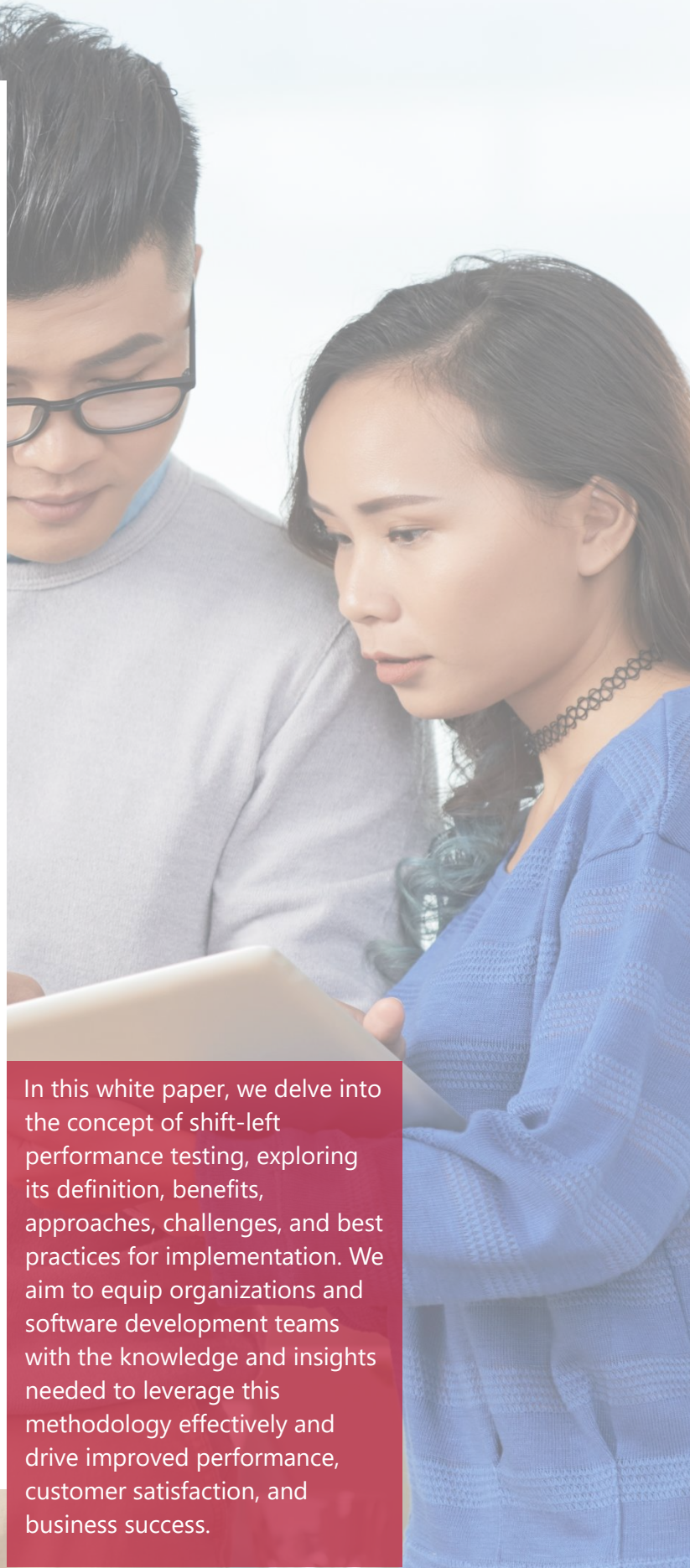
Introduction

In today's digital era, where speed, efficiency, and user satisfaction are paramount, ensuring optimal performance of software systems has become more crucial than ever. The traditional approach of conducting performance testing towards the end of the development cycle has proven to be inadequate in addressing the everincreasing demands of modern applications. Enter shift-left performance testing—an innovative and proactive approach that is revolutionizing the software development landscape.

Shift-left performance testing, often referred to as "Test Early & Test Often," involves testing the load and speed performance of code, features, and functions as close to the design and development stages as possible. By integrating performance testing activities early in the development lifecycle, organizations can gain valuable insights and mitigate performance-related issues before they become deeply ingrained in the codebase.

Imagine a scenario where a popular e-commerce website experiences sluggish response times and frequent outages during peak shopping seasons. Such performance issues can result in frustrated customers, lost sales opportunities, and damage to the brand's reputation. By adopting shift-left performance testing, organizations can identify and rectify these issues before they impact end-users, ensuring a seamless and satisfactory shopping experience.

In this white paper, we delve into the concept of shift-left performance testing, exploring its definition, benefits, approaches, challenges, and best practices for implementation. We aim to equip organizations and software development teams with the knowledge and insights needed to leverage this methodology effectively and drive improved performance, customer satisfaction, and business success.



Why read this white paper?

Organizations have prioritized customer experience as a key driver for their digital transformation. Why? Because enhancing the customer experience leads to differentiation in the market and contributes to increased revenue growth. Application development teams have already been successful in the adoption of the Shift Left practices for functional testing. In most organization performance testing is often conducted too late in the development cycle, leading to costly rework and schedule slippages.

In order to make performance testing more effective, application teams must embrace the concept of "shift left," which entails bringing performance testing earlier into the development process. In this whitepaper, we discuss the essential aspects of shift-left testing methodology, including its key benefits, challenges, and best practices. With a focus on the advantages of early software testing to assess performance, it is crucial for individuals considering its implementation to have a comprehensive understanding of the approach, process, benefits, challenges, and best practices associated.

What is shift-left performance testing?

Shift-left performance testing is a proactive and iterative approach that emphasizes testing the load and speed performance of code, features, and functions as early as possible in the development process. By incorporating performance tests into the daily development flow and executing them with every code commit, developers gain valuable insights into speed and load-performance issues. This enables them to address these issues promptly and prevents them from becoming deeply embedded in the codebase.

The concept of "shift-left" signifies the shift of performance testing activities to earlier stages of development. Rather than relying solely on post-development testing, the focus is on preventing performance issues from arising in the first place. Performance tests can be created and executed at different stages, such as unit testing or integration testing, to provide continuous feedback to developers.

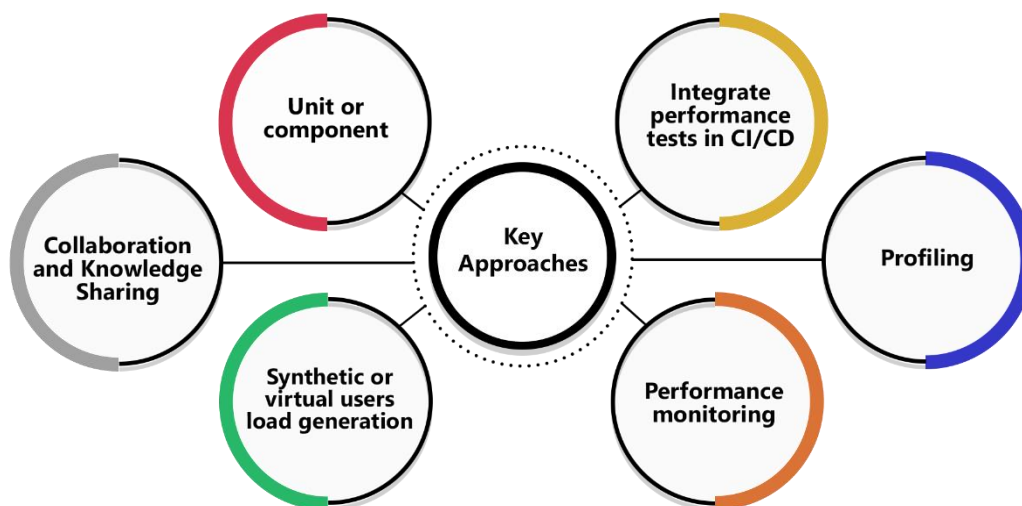
By adopting the shift-left approach, development teams can proactively identify and address performance-related problems before they escalate. This includes detecting issues like high load time (response time) of components or services, scalability limitations, and inefficient resource utilization. By identifying these issues early on, teams can optimize the performance of their code and ensure a smoother user experience.

Overall, shift-left performance testing enables a culture of continuous improvement, where performance issues are detected and resolved proactively throughout the development process. It empowers developers and testers to collaborate closely, identify performance bottlenecks, and optimize code for enhanced speed and efficiency. By catching performance-related defects early, organizations can deliver high-performing software that meets user expectations and avoids costly rework or delays in the later stages of development.

Why adopt shift-left continuous performance testing?

When adopting shift-left performance testing, organizations can choose from various approaches based on their specific needs and development processes. The key objective is to incorporate performance testing activities early in the development life cycle and ensure that performance is considered as an integral part of the software development process.

Here are some of the key approaches to consider:



Unit or Component Tests:

Evaluate the performance of individual units or components of the software system. By conducting these tests, you can identify and address performance issues at the code level. Establishing a baseline and benchmark during this process allows for continuous monitoring and improvement.

Integration of Performance Tests in CI/CD:

Continuous performance testing plays a vital role in the shift-left approach. By integrating automated performance tests into the CI/CD pipelines, organizations can regularly evaluate performance and receive continuous feedback on performance regressions. This approach enables the passing or failing of builds based on acceptance criteria such as response time of transactions and error rate.

Profiling:

Utilize performance profiling tools and techniques to analyse and measure the performance of the codebase. Profiling helps identify and address performance bottlenecks, memory leaks, and areas of code that require fine-tuning and optimization. By profiling the code, organizations can proactively prevent performance issues and ensure optimal code performance.

Synthetic or Virtual Users Load Generation:

Measure and assess system performance under various load conditions during the development process. This approach involves designing and executing performance tests that simulate multiple concurrent users, high data volumes, or complex transaction scenarios. By evaluating system performance under realistic load conditions, organizations can identify and address performance issues before they impact end-users.

Performance Monitoring:

Integrate performance monitoring tools into the development environment to measure and monitor key performance indicators such as response times, throughput, and resource utilization. Continuous monitoring helps identify performance deviations and trends, facilitating early detection and remediation of performance issues. By monitoring performance metrics, organizations can ensure the ongoing performance optimization of their software.

Collaboration and Knowledge Sharing:

Promote collaborative discussions and knowledge sharing among development and testing teams to build a performance-aware mindset and improve the overall performance testing approach. By fostering a culture of collaboration and knowledge transfer, organizations can leverage collective expertise to identify potential performance issues and implement effective performance testing strategies.

By implementing these approaches, organizations can effectively shift performance testing left in the development process, ensuring early identification and resolution of performance issues. This proactive approach leads to improved software performance, enhanced user experience, and overall software quality.

What are the major benefits of the shift-left performance testing approach?

In today's digital transformation landscape, customer experience plays a pivotal role in the success of any organization. The performance of software systems directly impacts customer experience, making it crucial for businesses to adopt a shift-left performance testing approach. This proactive testing methodology offers several significant benefits:



Enhanced Collaboration between Development and Testing Teams:

By incorporating performance testing early in the development cycle, the shift-left approach promotes collaboration between development and testing teams. Developers gain valuable insights into performance-related issues, allowing them to optimize code and address bottlenecks before they become deeply embedded. The iterative feedback loop between teams fosters a culture of continuous improvement and drives efficient collaboration throughout the software development process.



Early Detection of Performance Bottlenecks:

Shift-left performance testing enables the early detection of performance bottlenecks, allowing organizations to identify and resolve issues at an early stage. By leveraging performance testing tools and techniques, potential bottlenecks, such as high response times, scalability issues, or resource constraints, can be addressed proactively. This early detection mitigates the risk of performance-related failures and ensures that the software operates optimally under expected loads.



Cost and Time Savings:

One of the key advantages of shift-left performance testing is the significant cost and time savings it offers. By identifying and resolving performance issues early on, organizations can avoid costly rework and late-stage performance optimization efforts. The earlier performance issues are addressed, the lower the impact on development timelines and resources. Furthermore, by streamlining the performance testing process from the beginning, organizations can allocate their resources efficiently and optimize development cycles.



Improved Quality of Software:

Adopting a shift-left approach to performance testing results in improved software quality. By proactively testing the software's load and speed performance during the development phase, organizations can prevent performance-related defects from entering the codebase. Early identification and resolution of performance issues contribute to a more robust and stable software application, minimizing the risk of performance-related failures in production environments.



Enhanced User Experience:

User experience is closely tied to the performance of software systems. With shift-left performance testing, organizations can ensure that their software delivers a seamless and responsive user experience. By addressing performance bottlenecks early in the development cycle, organizations can optimize response times, minimize latency, and improve overall system performance. This translates into enhanced user satisfaction, increased user engagement, and improved brand reputation.



Faster Time to Market:

By integrating performance testing into the early stages of the development process, the shift-left approach accelerates time to market. Proactively addressing performance issues throughout the development cycle helps organizations avoid delays caused by last-minute performance optimizations. This allows businesses to release new features and updates more rapidly, meeting market demands and gaining a competitive edge.



Increased Revenue:

The shift-left performance testing approach directly contributes to increased revenue generation. By ensuring optimal software performance and user experience, organizations can attract and retain customers, leading to higher customer satisfaction, loyalty, and repeat business. Positive user experiences drive customer engagement, word-of-mouth recommendations, and ultimately, revenue growth.

Incorporating the shift-left performance testing approach empowers organizations to achieve significant benefits, ranging from improved collaboration and early issue detection to cost savings, enhanced software quality, and faster time to market. By prioritizing performance from the outset, businesses can deliver high-performing software applications that elevate customer experience, drive revenue, and gain a competitive advantage in the market.

What are some of the challenges implementing shift-left performance testing?

Implementing a shift-left performance testing approach brings numerous benefits, but it also presents certain challenges that organizations must address. These challenges arise during the journey of implementing this approach and require careful consideration and planning. The following are some key challenges associated with shift-left performance testing:

Shifting Mindset and Organizational Culture

Implementing a shift-left approach requires a shift in mindset and organizational culture. Development teams may initially resist taking on additional tasks and responsibilities. It is crucial for organizational leaders to educate and communicate the business value of shifting left. By promoting a culture of continuous improvement and emphasizing the benefits of early performance testing, organizations can overcome resistance and foster a performance-aware mindset among development teams. Ongoing job training and support are essential to help teams adapt quickly to this change.

Lack of Skills

Effective performance testing demands specialized skills such as DevOps expertise, API-based performance testing, and proficiency in performance monitoring tools. The lack of these skills within the organization can hamper the effectiveness of shift-left performance testing. To address this challenge, organizations need to invest in training and upskilling initiatives to equip teams with the necessary knowledge and skills. Collaborating with performance testing experts or partnering with external consultants can also help bridge the skill gap.

Limited Availability of Non-Functional Requirements

In the early stages of the software development life cycle (SDLC), detailed nonfunctional requirements may not be readily available. This poses a challenge in defining the key objectives and success criteria for performance testing. Without clear requirements, it becomes difficult to simulate realistic usage scenarios and establish performance benchmarks. To overcome this challenge, organizations should focus on gathering and documenting non-functional requirements early in the development process. Collaboration between stakeholders, architects, and business analysts can help define performance expectations and provide a solid foundation for performance testing efforts.

Limited Availability of Production-Like Environment

The availability of a production-like environment for performance testing can be limited. Non-production environments may not accurately reflect the complexity, scale, or load of the actual production environment. This discrepancy can impact the accuracy and reliability of performance test results. Organizations should strive to create realistic testing environments that closely resemble the production environment as much as possible. Investing in infrastructure and tools to replicate production conditions will ensure more accurate performance testing outcomes.

Integration with Agile and DevOps

Integrating performance testing activities into existing Agile and DevOps workflows can be a complex undertaking. It requires careful coordination, automation, and seamless integration with the continuous integration/continuous deployment (CI/CD) pipeline. Ensuring that performance testing is an integral part of the development process and aligning it with Agile and DevOps principles is essential. Developing comprehensive automated performance test frameworks and leveraging performance testing tools that integrate well with CI/CD pipelines will enable organizations to achieve continuous testing and analysis throughout the development cycle.

What are the “Best Practices” for Implementing Shift-Left Performance Testing?

Implementing shift-left performance testing requires careful planning and adherence to best practices. Consider the following recommendations to ensure successful implementation:

Training and Knowledge Sharing:

Conduct workshops or training sessions to educate team members on performance testing concepts. Share the benefits, challenges, methodologies, and tools associated with performance testing. Encourage cross-functional knowledge sharing and communication to foster a shared understanding of performance goals, test scenarios, and testing techniques.

Automation and Integration with CI/CD Pipelines:

Build a robust framework to automate performance tests and seamlessly integrate them into the continuous integration/continuous deployment (CI/CD) pipelines. Identify the appropriate tools for code profiling, performance testing, performance monitoring, and performance data analysis. Capture the performance of the system during each sprint and centrally store test assets. Set up automated alerts for any performance regressions by establishing performance trending.

Define Key Performance Objectives:

Clearly define key performance objectives and continuously fine-tune the strategy and plan as the project progresses. Well-defined performance objectives and metrics will set clear expectations for the test team. Ensure that the objectives are measurable and aligned with the expected performance of the system under development.

Simulate Realistic Load Conditions:

Identify business-critical scenarios and simulate realistic load conditions that mimic real-world usage patterns. Consider factors such as anticipated peak user load, data volume, network conditions, and other user behaviours. Use realistic data sets for various load conditions to obtain accurate performance insights.

Test Environment:

Ensure that performance tests are conducted in an environment that closely resembles the production environment in terms of infrastructure, configuration, and workload characteristics. Running tests in an environment that accurately reflects the production setup will enhance the accuracy and reliability of performance test results.

Prioritize, Improve, and Innovate:

Prioritize tasks that yield high value returns and focus your energy and time accordingly. Learn from each performance test cycle and identify areas for improvement. Incorporate feedback received and refine your approach to continuously enhance the effectiveness of shift-left performance testing over time.

By following these best practices, organizations can effectively implement shift-left performance testing, optimize their software's performance, and deliver high-quality products that meet customer expectations.

Conclusion

Shift-left performance testing emerges as a proactive and integral approach to software development, focusing on early and continuous testing of code, features, and functions. By incorporating performance testing activities early in the development lifecycle, organizations can reap numerous benefits and overcome challenges associated with traditional testing approaches.

The shift-left performance testing approach brings forth several advantages. Enhanced collaboration between development and testing teams ensures that performance issues are identified and addressed early, leading to improved software quality and user experience. Early detection of performance bottlenecks allows for timely optimizations, resulting in cost and time savings. Furthermore, the shift-left approach facilitates faster time to market, increased revenue, and a competitive edge in today's digital landscape.

However, implementing shift-left performance testing is not without its challenges. Shifting mindset and organizational culture, the lack of specific skills, limited availability of non-functional requirements, and integration with Agile and DevOps workflows require careful consideration and proactive measures.

The shift-left performance testing approach empowers organizations to proactively address performance issues, deliver high-quality software, and create exceptional user experiences. By embracing this methodology, organizations can position themselves for success in today's competitive and demanding digital landscape.

To successfully implement shift-left performance testing, organizations should adhere to best practices. These include training and knowledge sharing, defining key performance objectives, automation and integration with CI/CD pipelines, simulating realistic load conditions, ensuring the appropriate test environment, and prioritizing continuous improvement and innovation.

By embracing the shift-left performance testing approach and following best practices, organizations can elevate their software development processes, enhance customer experience, and achieve business success. The journey to shiftleft performance testing may present its share of challenges, but the long-term benefits and positive impact on software performance and user satisfaction make it a worthwhile endeavour.

Why testhouse?

At Testhouse, we take pride in our extensive experience in delivering large-scale performance engineering services for our clients. With the capability to deliver tests for up to 500,000 users within just 6 to 8 weeks, we are confident in our ability to optimize the performance and stability of your digital products. Our team consists of over 30 performance testing specialists who deliver services to customers across the globe. We have a wealth of experience in a variety of performance testing tools, including Jmeter, Gatling, HP LoadRunner, Neoload, and more. One of our key strengths is our ability to apply shift-left testing practices around performance, which has helped us to reduce the cost of quality by identifying defects early in the life cycle.

For support, reach out to us at contact@testhouse.net