

How End-to-End Payment Systems Testing *Really* Ends: Evaluating Performance, Reliability, and Recoverability

Do end-to-end test plans end too soon? Perhaps—if they do not include meaningful performance and stress testing. Perceived as expensive and difficult to execute, performance testing and stress testing are often eliminated from the test schedule in favor of other test options. However, as institutions face significant changes in systems, product offerings, and delivery channels; the ability to unequivocally determine system capacity, reliability, and recoverability is essential. This article illustrates that this “last mile” of payment systems testing doesn’t need to be the hardest.

When financial institutions seek end-to-end testing options, they hope to ensure infallible, reliable payment systems operation. However, end-to-end test plans typically do not include performance or stress testing—testing that is arguably the final step in guaranteeing solid payment systems operation. Why? Meaningful performance and stress testing of payment systems is perceived as expensive, complex, and difficult. Consequently, it often drops off the testing schedule. But the significant changes that financial institutions face today demand definitive testing to ensure system performance, reliability, and recoverability. This article relates financial industry trends with the system changes likely to result from them, and offers suggestions for planning to incorporate performance testing and stress testing into your organization’s end-to-end testing.

Testing System Performance, Reliability, and Recoverability

The terms “stress testing” and “performance testing” are frequently used interchangeably. In fact, these types of testing are conducted differently and each has different goals.

- **Performance testing** determines the speed at which a system can process a specific load and allows the tester to observe the effect on the system’s speed and stability as resources are taxed. This information is then used to determine where and how to tune the system. Performance testing can also be used to determine system capacity; specifically, the point at which demands on the system can degrade response time or result in unpredictable or erroneous processing. The goals of performance testing are to eliminate bottlenecks and improve system reliability.
- **Stress testing** is designed to stress the system to failure. Stress testing investigates whether the system crashes, or fails gracefully (i.e., provides meaningful error messages) and recovers from failure without critical data loss. (Because stress testing takes the system to failure, disaster recovery scenarios can also be tested to ensure that a financial institution can rely on their disaster recovery processes to take over processing in the event of the primary system’s failure.) The goal of stress testing is increased recoverability.

Both types of testing are important. This article addresses the need for and requirements of performance testing and stress testing of today’s payment systems.

How Industry Trends Increase the Need for Performance and Stress Testing

Most financial organizations agree in principle that performance and stress testing are good ideas, but often these categories of testing are displaced by other business concerns. Making a business case for this testing becomes much easier as you consider the effect that current market trends will ultimately have on systems.

Industry Trends	Examples of Resulting System Changes
<ul style="list-style-type: none"> <p>• Migration</p> <p>All financial institutions are engaged in some sort of migration project (hardware, software, communications, etc.) at any given time. In particular, payment systems that underwent their last major updates in preparation for the Year 2000 are now being evaluated for updates and upgrades.</p> <p>• New transactions, growing transaction volumes</p> <p>New products (such as contactless cards, mobile payments, chip cards, etc.) often result in changes to payment systems software. EMV processing can mean increased system overhead from more messages per transaction and more calls to the HSM.</p> <p>• Fraud</p> <p>Financial organizations must continue to combat fraud. Institutions that formerly used rules-based fraud detection are shifting to newer neural network fraud detection systems—new systems that will require thorough testing to validate that the fraud detection systems perform correctly even when system resources are stressed.</p> <p>• Compliance</p> <p>Complying promptly with mandates is often challenging, but especially as institutions try to cope with the demands of accomplishing more work with static (or dwindling) personnel resources.</p> 	<ul style="list-style-type: none"> <p>• Hardware changes</p> <ul style="list-style-type: none"> ✓ Consolidating hardware ✓ Switching host software platform ✓ Upgrading host hardware ✓ Adding new devices (ATMs, POS) ✓ Changes to communications infrastructure (new routers, new switches, new communications protocols) ✓ HSM upgrades (new vendors, EMV, new encryption methods) <p>• Software changes</p> <ul style="list-style-type: none"> ✓ O/S upgrades and updates (patches, new releases, security updates, etc.) ✓ Mandates (interchanges, card associations) ✓ Host software changes (patches, new requirements, updates, new product/transaction offerings) ✓ Payment system changes <p>• Database changes</p> <ul style="list-style-type: none"> ✓ Database relocation ✓ Database migration

When over-stressed systems fail, the resulting errors can quickly (and disastrously) multiply. Performance testing ensures that payment systems can successfully process the anticipated volume of transactions. Stress testing enables organizations to find the breaking point of their systems and ensure, if the worst happens and the system fails, that seamless recovery is possible.

Key Considerations When Planning Performance Testing and Stress Testing

Even though you may agree that your organization needs to conduct performance and stress testing, you may not know how to begin preparing for these tests. Key considerations for planning performance testing and stress testing include the following.

Access to a knowledgeable team

Assembling the right team will do as much to ensure the success of your performance test or stress test plans as any other factor. When developing a test plan, you must gather information and assure cooperation from many others in your organization: database administrators, systems engineers, network administrators, key settlement personnel, and so forth. Getting input from these team members is a critical first step in successful performance testing or stress testing, as your team members can provide valuable suggestions about what and how to test. Equally important is getting your team members committed to the testing process and goals, because your team can help ensure that the proper equipment and personnel are available when test time arrives.

Clear goals and objectives

It's impossible to know if your testing was successful if you don't have clear goals before testing begins. A critical step in identifying those goals is defining your project scope.

- In performance testing, for example, are you attempting to fine-tune an application, system, or network? Are all your system end-points (devices, interchanges, etc.) required for testing? Performance testing can be as specific as having your developers use a profiler to test a particular routine, modify it, and retest it to see if processing time in that routine improves; or as extensive as having systems engineers monitor the memory and disk I/O during processing of specific transaction types and loads. Regardless of the focus of your performance testing, you must target resource usage thresholds and processing times so you can use those during the test and when evaluating the test results.
- In stress testing, for example, are you testing in regard to capacity planning—testing your ability to handle increased transactions due to growth, acquisitions, or anticipated holiday/seasonal peak periods? Stress testing should determine the breaking point of your system as well as examine the usefulness of messages associated with system failures and the ability to recover data associated with the transactions that could not be processed.

Specific requirements for the test environment

Typically, your stress test or performance test plan will include details about the test environment. Obviously, testing requires a test system representative of your production system and the availability of the proper personnel and tools. In addition, depending on your test goals, your requirements might include factors such as:

- Testing occurring at a certain time of day, such as simulating peak processing while running reports and system backups
- Testing including specific transaction requirements, such as:
 - Specific ratios of transaction types (for example, a certain number of withdrawals, balance inquiries, transfers)
 - Specific ratios from various system end-points (for example, specific percentages of ATM transactions, POS transactions—perhaps even transactions from a specific interchange)
 - Specific BIN ratios (for example, a specific number of transactions per BIN or institution)
- Testing using specific monitoring tools (such as database profilers, hardware monitoring utilities, network sniffers, etc.)

Transaction testing also requires other preparation, such as:

- Ensuring a sufficient number of cards to generate meaningful test data
- Ensuring account balances are sufficient to cover the test duration
- Ensuring adequate card velocity limits (such as number of transactions, daily limits, etc.)

Task assignments based on test results

At the conclusion of your stress testing and performance testing, you must determine who will evaluate the results, what actions will be taken, and who is assigned the task, for example, to reconfigure hardware, tune software, or resolve bottlenecks.

Selecting Performance Testing and Stress Testing Tools

The “last mile” of testing doesn’t need to be the hardest. After you have a test plan with clearly defined goals, you can evaluate tools by measuring their usefulness in achieving those goals. Make certain your end-to-end test solution includes performance and stress testing tools that can mimic all the end-points in your system and deliver meaningful results on system performance, reliability, and recoverability.

About Paragon Application Systems

Paragon Application Systems is a leading global provider of ePayment simulation, configuration and testing software tools to the financial industry. More than 400 financial institutions in over 80 countries use Paragon tools to improve quality and reduce time-to-market. Paragon’s broad customer base includes major interchanges, processors, leading software providers, banks and credit unions. Visit Paragon Application Systems at www.paragonedge.com.

About the Authors

Bob Collins, VP of Technology and Services at Paragon, has 20 years of experience implementing and testing ePayment systems worldwide and an in-depth understanding of the financial industry. As one of Paragon's founders, Bob has hands-on experience developing and using Paragon's solutions to test a wide variety of financial message formats, and has set the standard for providing exemplary technical support to Paragon's clients. Bob earned his BS in Computer Science (concentration in Business Administration) from East Carolina University.

Harold Pruitt, a Solutions Architect at Paragon, has conducted several stress and performance test engagements for Paragon customers. Harold also provides hands-on training and mentoring to customers who are implementing Paragon's testing solutions. He has extensive experience in stress and performance testing, and has benchmarked host systems at various hardware labs around the globe. Harold is a graduate of the University of Houston with a degree in Philosophy.

Eric Bergemann, a Lead Software Engineer on the Paragon Development team, is responsible for software development for several of Paragon's products, including FASTress (Paragon's performance, load, and stress testing solution). He has assisted several customers in setting up and conducting their performance and stress testing. Eric has a BS in Computer Science from Campbell University.