Network Applications: Ensuring Performance and QoE with Network Emulation
# Table of Contents

- Introduction ........................................................................................................... 4
- Enterprise Applications ......................................................................................... 4
- Network Application Challenges ............................................................................ 6
- Meeting the Challenges ......................................................................................... 10
- Conclusion ............................................................................................................ 12
Introduction

Information management applications have progressed from an innovation to a competitive advantage to essential infrastructure for enterprises. As applications have matured, they have expanded from internal, LAN-based systems to distributed systems deployed across the wide area network (WAN) or cloud and used by employees, partners, and customers with varying degrees of access.

Networked applications are pervasive in business, and dependence on them keeps increasing. An outage, or even poor performance, can cripple day-to-day business, affecting revenues directly through lost sales and indirectly through decreased productivity.

With the proliferation of enterprise applications and increased complexity of systems and networks, it can be difficult to know how an application will perform on any given network, and to identify the actual cause of performance issues when they arise. Because enterprise applications are rarely tested under WAN conditions, expectations of performance are often at variance to actual performance.

Ixia’s Network Emulators are essential tools for effectively developing and deploying enterprise applications. They provide the ability to accurately and precisely emulate real-world network conditions prior to deployment, reducing development schedules and support cost, avoiding live troubleshooting and downtime, and consequently increasing revenue and competitive advantage.

Emulating production network enables companies to understand the behavior of an application before it goes live, to determine whether it will meet service level objectives (SLO), and to correctly identify factors affecting performance. Network or application changes are based on data rather than guesswork or conventional wisdom, assuring the performance, robustness, and scalability the organization needs in the initial rollout.

Enterprise Applications

Distributed information systems have expanded beyond simple accounting and inventory applications to address the wide range of issues facing the enterprise, such as customer relation management (CRM), supply chain management (SCM), business process management (BPM), enterprise resource planning (ERP), e-commerce, and other essential functions. Some organizations use outside solutions, some develop their own in-house, and others integrate multiple best-of-breed solutions.

The drivers for adopting and implementing enterprise applications are ultimately financial – reducing costs, increasing revenues, and improving competitive advantage. By consolidating functional silos at various locations, divisions, or companies into a centralized system, total cost of ownership for IT systems can be greatly reduced.
Architecture

Networked applications typically use a three-tier architecture consisting of the data tier, the application tier, and the client tier.

To maintain business continuity, data is protected through offsite backup or real-time replication at a remote location.

The *data tier* houses the files and databases containing the information the system manages. These are typically SQL-based databases or legacy data systems. The application tier communicates with the data tier through an application-programming interface, typically supporting CORBA, DCOM, RMI, RPC or a proprietary system. To maintain business continuity, the data is protected through offsite backup or real-time replication at a remote location.

The *application tier* performs several functions and can be divided into sub-layers that can be implemented on a single server or distributed among multiple servers. Web and application servers reside in this tier, interfacing with the two other tiers to accept a transaction request from the client, process the request, query the data tier, and return the results to the client tier. Business logic resides in the middle tier and includes implementation of workflows and business rules that distill policy such as pricing, logistics, or order processing into algorithms.

In keeping with the distributed nature of the architecture, servers implementing the application tier may or may not be co-located with data tier servers. In fact, servers implementing various functions of the application tier can be located remotely from each other.

The *client tier* is the user interface to the application. It can be a thin client (typically a browser-based application using HTML or Java that doesn't require installation on the client computer), a fat client (an application that must be installed on the client computer, typically implemented in C) or a hybrid.
Many vendors offer both fat and thin clients, with the momentum going toward thin. The client presents a screen used to gather information from, and present results to, the user. To reduce the amount of network traffic, the client often performs validation of input before sending transactions to the application tier. While a client may be local to the facility housing the application tier, most often clients are located remotely from the other tiers and may be connected via a low-bandwidth dialup line or high-bandwidth broadband link.

**Network Application Challenges**

**Application Management**

A single-enterprise application is a complex system. Given the reality that most organizations have multiple applications that may or may not be integrated with one another, it’s no surprise that the complexity can quickly become unmanageable.

Most applications have some level of management available, but a single system fault – a mis-configured server or hitting system limit – can generate multiple alerts. An application expert must sort through the alarms in the context of system-wide conditions to determine the underlying cause and propose a solution. Problems of this nature occur dozens of times each day in enterprise data centers, making troubleshooting and problem isolation a time-consuming and expensive task.

Application management software, whether built-in or provided by a third party, is designed to assist IT managers by providing an early-warning system to identify issues before they affect users. These systems are used to monitor performance metrics, automate routine maintenance tasks, and apply artificial intelligence to alarm situations to assist in diagnosing root causes and identifying solutions.

**Application Security**

Enterprise applications manage vast amounts of sensitive information, including proprietary designs, trade secrets, financial data, customer account data, and personnel records with Social Security Numbers (SSNs) and other data. Unauthorized access to such information can have serious implications for an organization, including lost revenue, damaged reputation, and possible litigation.

Firewalls, encryption, and user authentication are typically imposed at gateway points but not inside the network. As enterprise applications go online, functions that were once protected in back-office zones by physical access now have the potential to be exposed to threats from the outside world.

Conversations between the application and data layers are usually assumed to be secure, and applications therefore often have unrestricted access to the database. But the first level of security can be compromised through various methods, including dynamic SQL commands, password cracking, or configuration vulnerabilities such as open SQL server ports. Once the perimeter is breached, the intruder can use the application layer to gain high-level access to the database. For reliable security, firewalls must be installed not only at the gateway, but also between each tier of the architecture.

There is also risk from internal threats. A database user can be tricked into running malicious code through a Trojan horse or a seemingly benign utility. Rogue users or administrators can eavesdrop on the network to pick up passwords or data.

Given the reality that most organizations have multiple applications that may or may not be integrated with one another, it’s no surprise that the complexity can quickly become unmanageable.
For basic protection, strict attention to security-focused configuration and regular updates and patches for operating systems, server applications and database systems are essential. In addition, encryption should be utilized between all layers.

**Application Performance**

The distributed nature of enterprise applications can lead to performance problems. In addition, the solution to one problem, such as compression to reduce the number of bits transmitted, can create other performance issues:

- **Congestion.** The disparity between the bandwidth available on the LAN versus the WAN can lead to congestion when moving large amounts of data between locations, for example between the primary servers and off-site mirror or back-up servers for disaster recovery and business continuity.

- **Delay.** Distance introduces delay problems, especially for connection-oriented protocols like TCP, which require multiple exchanges to establish a session and acknowledgments after each transmission.

- **Processor-intensive functions.** Compression is often used to reduce congestion. Encryption is used to address security concerns. Both functions place high demands on the server CPU that can affect the number of transactions it can process.

Application performance problems can be addressed in two ways: by optimizing the application for WAN communications and by deploying a WAN accelerator. Application optimization is best done during development rather than as an afterthought. Algorithms are designed to avoid chatty behavior, for example by compiling multiple requests into a single message.

WAN acceleration solutions reduce costs by increasing the effective throughput of the existing network. Most solutions achieve a 10X to 40X improvement in response time or transaction processing rate using a wide array of techniques.

**Compression.** Typical compression techniques replace repeating bit patterns with a short label before sending the data across the WAN. On the receiving side the label is removed and the original pattern is inserted back into the bit stream, reducing the number of bits transmitted by up to 70% (see Figure 1).

![Figure 1: Compression (A - No compression, B - With compression)](image)
Transport Control Protocol (TCP) maintains the end-to-end connection by requiring periodic acknowledgments from the destination. A sliding window (the number of packets sent before an acknowledgement is required) controls congestion (see Figure 2). WAN acceleration increases throughput by sending acknowledgments locally rather than across the WAN (see Figure 3.)

**TCP Optimization.**

![Diagram showing TCP optimization](image)

**Note:** each transmit consists of one or more IP packets, depending on the TCP window size. Each acknowledgement is one IP packet.

**Application performance can be addressed in two ways:** by optimizing applications for WAN communications, or deploying a WAN accelerator.

**Figure 2:** TCP Optimization – A) Normal transmit and acknowledgement results in increased window size, B) Lost packet, no acknowledgement causes retransmit and reduced window size, C) Excessive delay cause acknowledgement to arrive after timeout, resulting in re-transmit and reduced window size

**Note:** each transmit consists of one or more IP packets, depending on the TCP window size. Each acknowledgement is one IP packet.

![Diagram showing optimized TCP connection](image)

**Figure 3:** Acknowledgements do not travel across the WAN and therefore do not experience long delays. More efficient connection protocols are used between the accelerator appliances, which are not pictured on the ladder diagram.

**Note:** Each transmit consists of one or more IP packets, depending on the TCP window size. Each acknowledgement is one IP packet.
**Caching and Mirroring.** Client-side WAN accelerators store copies of web pages and other data as it passes through so that subsequent requests can be answered locally by the accelerator rather than requiring retransmission from the remote server (see Figure 4).

*Figure 4: Caching and Mirroring – A) Caching, B) No caching*

**Traffic Shaping through QoS.** Delay-sensitive traffic such as video or voice is prioritized and guaranteed delivery. Traditional data traffic such as email or web page requests are delivered if available bandwidth permits, or dropped if higher-priority traffic has consumed the available bandwidth.

**Off-loading CPU-Intensive Tasks.** By offloading CPU-intensive tasks such as compression, encryption, and XML processing to an appliance, organizations can not only improve WAN performance but also handle increased traffic without the expense of licensing additional processors (see Figure 5).

*Figure 5: Off-loading CPU-Intensive Task*

**Forward Error Correction (FEC).** Some WAN acceleration solutions use FEC to reduce the need for retransmissions on loss-prone networks. FEC sends recovery information in-band with data for indexing to allow for reconstruction of lost packets.
Too often, application performance isn’t evaluated until after deployment, exposing the IT department to unwanted attention when response time doesn’t match expectations. Troubleshooting becomes frantic. Meetings may become emotional and politically charged. Users may accuse the IT department of using them as an ad hoc QA department. Pressure to provide a quick fix may short circuit efforts to identify root causes and effective solutions.

### Meeting the Challenges

Performance testing and network emulation are key to successful enterprise application deployment:

- Performance testing simulates the demands placed on the system by generating realistic traffic/transaction loads and modeling realistic user behavior
- Network emulation creates the expected environment by reproducing realistic network conditions and behavior

Both are critical to predicting application performance before deployment. Evaluating realistic traffic loads on a perfect network can create a false sense of confidence that the application will perform to expectations after deployment.

Testing provides concrete data to answer important questions about application readiness and to set performance expectations, which can then be communicated to management and users.

<table>
<thead>
<tr>
<th>Questions</th>
<th>Example Answers</th>
</tr>
</thead>
<tbody>
<tr>
<td>What are the application performance thresholds in terms of network metrics</td>
<td>Response time falls off drastically when packet loss exceeds 8%.</td>
</tr>
<tr>
<td>(delay, packet loss), number of transactions or number of users?</td>
<td></td>
</tr>
<tr>
<td>Which part of the system is the weak link – web server, application server,</td>
<td>Delay above 200 ms creates TCP timeouts and retries, resulting in response times</td>
</tr>
<tr>
<td>database, and network – and what are the remedial action items?</td>
<td>beyond SLO. A WAN accelerator can eliminate TCP connection issues for remote</td>
</tr>
<tr>
<td></td>
<td>offices, but doesn’t address issues for mobile users.</td>
</tr>
<tr>
<td>Can the application meet SLO under all expected profiles, such as remote</td>
<td>Users can expect 3 second response times when connecting via DSL.</td>
</tr>
<tr>
<td>office, home office or business travel access?</td>
<td></td>
</tr>
<tr>
<td>Does the new version solve performance issues or introduce limitations not</td>
<td>Regression testing against our reference network emulation profiles demonstrates</td>
</tr>
<tr>
<td>in the earlier version?</td>
<td>a 12% improvement in response time for remote users.</td>
</tr>
<tr>
<td>Will the proposed fix actually improve performance?</td>
<td>The developers argue that the network, not the application, is the source of</td>
</tr>
<tr>
<td></td>
<td>performance issues. Bandwidth limitation testing shows no significant performance</td>
</tr>
<tr>
<td></td>
<td>enhancement when WAN bandwidth is increased.</td>
</tr>
</tbody>
</table>
In A Perfect World: Establishing the Baseline

As the first step in producing a realistic performance prediction, the baseline establishes how the application performs in a perfect world, or at least on a perfect network. The QA team studies the organization to assess periodic and event-driven usage and behavior changes, such as time-of-day variations, end-of-month/quarter/year demands, and factors that may be unique to the organization or industry. The results form the basis of a set of traffic-load and user-behavior profiles that reflect the range of expected usage patterns.

These profiles should include the login storms that occur in the morning and after lunch, typical usage with full attendance, and peak usage during crunch times, such as end-of-period reconciliation and reporting. The profiles will be used during baseline testing and network emulation testing.

The application is first tested with the load/behavior profiles under optimum network conditions. The infrastructure of the test lab provides an environment with no packet loss, no packet reorder, and minimal delay. The metrics reported from each profile establish the application performance baseline, which is compared to SLO. An application that fails or barely passes SLO in an optimum network will not have acceptable performance across the WAN.

Next, failure thresholds are established. Threshold testing doesn’t use typical load/behavior profiles. Instead, traffic loads are ramped up in steps and failure points (the level at which the performance metrics violate SLO) due to the limits of the application are identified. Threshold testing helps predict scalability limits and allows management to plan for growth, identifying points where investment in infrastructure will be required.

Real-World Testing: The Reality Check

Once the application achieves acceptable performance and margins, each of the tests performed during baselining are performed again under more realistic conditions.

The network team provides a set of network profiles that reflect the characteristics of each type of user:

- An on-site profile will have LAN speeds and minimal impairment
- A remote office profile will have WAN speeds and more impairment
- A home office profile might have broadband access speeds and delays for VPN security
- A mobile user may have a variety of connection options, including dialup, either directly into the company network or to a provider and then via VPN, or broadband with VPN
The key to a successful application rollout is robust testing under realistic traffic and network conditions.

Conclusion

Enterprise applications are essential elements of the business infrastructure. Poor application performance can cripple day-to-day business and affect productivity. The complexity of enterprise applications and networks makes it difficult to predict application performance and identify the root cause of performance issues.

The key to a successful application rollout is robust testing under realistic traffic and network conditions. The two-tier testing model of establishing baseline and network emulation metrics and determining load and impairment thresholds empowers you to compile the data you need to make informed decisions and communicate realistic expectations vertically and horizontally through your organization.

Ixia Network Emulators are essential tools for effectively developing and deploying enterprise applications. They provide the ability to:

- Accurately and precisely emulate real-world network conditions prior to deployment
- Efficiently test different network user profiles
- Reduce development schedules and support cost
- Avoid live troubleshooting and downtime
- Increase revenue and competitive advantage

Network emulation testing can mean the difference between a smooth deployment and a crisis. Make sure you have the tools required for a successful rollout.
This material is for informational purposes only and subject to change without notice. It describes Ixia's present plans to develop and make available to its customers certain products, features, and functionality. Ixia is only obligated to provide those deliverables specifically included in a written agreement between Ixia and the customer.