The Business Case for Test Environment Management Services

Application test environments managed by temporary resources can cause spikes in release costs, significantly impact quality/test team productivity, high rates of defects that leak into production systems and consequent business losses. Test environment services delivered via a dedicated managed services approach represents a superior alternative that delivers greater monetary benefits and fosters enhanced reputation as well as brand loyalty.

Executive Summary

This much about testing is incontrovertible: Poor test environment configuration and utilization, lack of automation and comprehensive tooling, and the unavailability of test environments can impact the quality of testing and the application delivered. This is caused primarily by the lack of ownership and management of test environments.

The reasons are manifold. For starters, most IT organizations lack a dedicated testing team that ensures code quality from top to bottom. In fact, many organizations temporarily divert employees from development and other responsibilities to manage test environments. Such a setup leads to a lack of accountability and creates issues that often impact the quality of testing, which in turn compromises the quality of software applications. Our experience suggests that the cost of QA project/activity is increased by at least 20% to 25% due to diversion of efforts by testers and developers.

Managing environment-related issues and defects consumes nearly 40% of the effort involved in the software development lifecycle.1 Apart from delaying testing cycles and creating additional costs, the test management problems noted above force managers to unnecessarily rush through the testing process, conduct tests in environments that are far from optimal (i.e., environments that do not approximate production scenarios) and result in risk-based sign-off (i.e., acknowledging the risks involved in releasing a product that is not completely tested).

Against this backdrop, test environment management services (TEMS) provides an accountable model, thus making it a superior alternative. The TEMS framework helps organizations to address common test environment issues and achieve efficiencies in demand fulfillment, capacity utilization and environment availability. Delivered as an end-to-end managed service, TEMS includes creating a centralized, single
ownership for environment management, monitoring and maintenance along with cloud-based infrastructure provisioning.

TEMS is designed to:

- Support in-flight projects, planned projects, maintenance projects and new application releases.
- Manage existing test environments, build new test environments and provide additional capacity on demand via the cloud.
- Help organizations establish the processes and controls that are required to place an application environment in the cloud, provided the application is properly cloud-enabled.

The monetary and reputational benefits of switching to TEMS are significant as its per-project costs are lower than test projects without managed services. However, organizations need to be aware that the TEMS model has its own limitations. They must also exercise due diligence in choosing the right provider in order to achieve desired results.

Why Managed Test Environment Services?
Jeff and his team are excited about their first big assignment — testing a new business application to be released shortly. They have left no stone unturned in their preparation.

However, on the day of testing, Jeff faces a few setbacks. First, he discovers that the test environment was not built to the project specifications due to lack of capacity and configurations. Second, by the time he gets the configuration right, he is staring at an unscheduled infrastructure maintenance activity. He is forced to hurry through the testing process. As a result, defects surface immediately after the application is rolled out.

Such incidents are common in testing centers where lack of ownership and poor management of testing impact quality, allowing defects to pass through to production.

Issues in Test Environment Management
The growing criticality of business applications needs no substantiation. The increasing complexity of today’s applications has expanded the scope of testing, creating unforeseen challenges for testing teams. To deal with the complexity and the pressure to reduce release cycle times, organizations must ensure that test environments closely mimic real-time situations and are highly available. Reduced release cycle time does not provide an allowance for defect leakage into production, and regardless of the project time frame all testing steps involved until release sign-off must be rigorously followed.

Test environments have a vital role to play in delivering fully tested code and in ensuring confidence in the successful release of the application. Our experience with organizations across industries suggest that poorly built test environments often limit the testing team’s ability to test applications for various scenarios and increases required testing cycles, which in turn results in unnecessary costs and allows defects to pass through to the production environments.

Poor software quality costs organizations worldwide $500 billion annually, according to a recent survey by Caper Jones & Associates LLC. Defects seep in at various stages of an application’s development lifecycle; importantly, finding and fixing these bugs in the early stages costs less than remediation during the later stages. According to Gartner, “The cost of fixing defects ranges from a low of approximately $70 (cost to fix a defect at the requirements phase) to a high of $14,000 (cost to fix a defect in production).”

Testing is considered an important but not very business-critical activity; hence, test environments within many organizations receive a low priority. Test environments are typically managed not by a dedicated unit but by a team of employees temporarily diverted from development and production support activities. With no unit exclusively responsible for testing, there is a lack of ownership and accountability. Further, such ad hoc testing teams often have limited capabilities in building and managing test environments, leading to unnecessary delays in creating test environments and other issues (see sidebar, page 4) that significantly impact the effectiveness of testing and ultimately the business. It is therefore not surprising that in an Aberdeen Group study, only 42% of over 200 organizations surveyed were happy with the performance of their business-critical applications.

The need for efficient test environments is endorsed by an old but still relevant National
Institute of Standards and Technology (NIST) study. According to the study, improving test infrastructure can save U.S. organizations one-third of their annual QA costs.5

**TEMS to the Rescue**

By applying a combination of consulting and managed services, the TEMS framework addresses common issues that plague test environment management. In most deployments, a dedicated TEMS team works closely with an organization’s IT department to create a centralized resource that owns, monitors, and controls test environments. A real-time project dashboard typically enables day-to-day tracking of project milestones and metrics, while an integrated delivery model synthesizes the efforts of multiple project teams.

One of TEMS’ key attributes is that it can be offered modularly as well as an end-to-end service (see Figure 1). It can also be delivered in multi-vendor scenarios and with minimum disruption to daily business activities in a rapid and robust fashion by following best practices for change management. Organizations can evaluate their current test environment management setup and choose the most suitable service.

TEMS provides a single contact point for resolution of issues, queries, changes and all other requirements, thus creating greater accountability for losses due to test environment issues. Broadly, TEMS’ end-to-end service includes the following:

- Setting up new environment: service on-boarding and build-out.
- Environment monitoring and management: hardware, networking, systems software and applications.
- Infrastructure monitoring: servers, databases, mainframes, etc.
- Service request management: issue resolution and coordination.
- Release management: systematic updates, terminations and reports.

Further, TEMS offered via the cloud can generate additional benefits by eliminating the need for costly upfront Cap-Ex investments and providing on-demand access to test environments.

**TEMS: A Better Alternative**

In the traditional model, the cost of test environment management is borne by various teams. When organizations switch to the TEMS model, those costs become visible since under TEMS a separate group is accountable for providing testing services through a direct cost allocation model. While TEMS merely makes the relevant costs visible, organizations may mistakenly believe it necessitates additional costs. On the contrary, the monetary losses due to environment downtime—paying testers for the idle time and for the additional time they work to finish the task—are far greater than the cost of creating a separate team that minimizes the incidence of environment issues.

---

**TEMS Suite**

<table>
<thead>
<tr>
<th>Module</th>
<th>Services</th>
</tr>
</thead>
</table>
| Environment Management               | • Manage/monitor SLAs.  
• Plan/analyze environment requests.  
• Facilitate design workshops.  
• Liaise with testing/infrastructure teams.  
• Create/update service reports.  
• Identify improvement areas. |
| Environment Monitoring               | • Additional responsibility for monitoring servers and notification. |
| Environment Maintenance              | • Handle infrastructure support or any combination of technology support levels.  
• Incident management, problem management, change management.  
• Build and smoke-test environment. |
| Cloud Infrastructure Provisioning/Management | • Provide hosting services (using partner vendor).  
• Provision virtual servers and test environments. |
In this regard, TEMS offers a superior alternative to the traditional approach, creating new efficiencies in managing test environments and delivering greater monetary benefits (as described below). Such advantages arise out of the fact that ownership and accountability within TEMS rests with an exclusive team as opposed to the traditional model where a heterogeneous team is accountable for test environment management activities. Such a centralized, dedicated TEMS team therefore frees up ad hoc resources drawn together from multiple groups and instead allows them to focus on their core activities.

TEMS’ major benefits include:

• **Automation for increased efficiency:** Cost-effective automation option and significant experience gained from similar engagements along with the ability to integrate automation tools effectively across enterprise using Run Book Automation, Operations management, Autonomics and Cloud orchestration will improve the efficiency and effectiveness of test environments.

• **Improved build efficiencies:** With a dedicated team of specialists running the show, new test environments can be built and deployed rapidly. This minimizes the defects that arise due to inaccurate test environment configuration (the rate of which is 30% according to studies) and achieve build efficiencies of more than 80%.

• **Reuse prevents waste:** As a “tool-agnostic” framework, TEMS enables reuse rates of more than 80% of customer-invested tools and technologies, thereby lowering the total cost of ownership (TCO).

• **Increased environment availability:** Through proactive monitoring, TEMS ensures the health of the environment as well as its availability. In some cases, average uptime of more than 90% has been achieved from just 68%. Higher availability allows testers to perform extensive and exhaustive testing which vastly improves result efficacy.

• **Proper scheduling replaces chaos:** Improper scheduling of tests when the environment is shared by multiple testing teams often delays testing projects. This forces test managers to conduct non-functional testing in scaled-down environments.

---

**Quick Take**

**Drawbacks of Traditional Test Environment Management**

• Test environments differ from production environments in terms of operating systems, patch levels, software versions, configuration, etc. The wider the gap between the test and production environments, the greater the chance of an application failing after being deployed or a defect leaking into live systems. Further, configuration changes made in response to errors often go untracked, causing errors when applications are moved into production.

• Loosely managed asset control necessitates undue investment in infrastructure. Inadequate access control reduces confidence in the state of the environment and its configuration control. Together, this introduces additional risks and adds unnecessary costs to the project.

• Simulating the production issue in the test environment to identify root cause issues is challenging due to a lack of complete alignment between the test environment and production versions across system component layers.

• Testing teams often clone production databases or extract data by writing scripts to create test data. This approach takes a lot of effort, is prone to errors and may not meet data protection policies. Often, this activity is not change-controlled and is non-auditable.

• The conventional approach of manually creating in-house testing environments that fully mirror the complexities and multiplicities of real-time environment consumes a lot of resources. This has prompted application developers to adopt the risky approach of conducting testing in production environments. However, the Sarbanes-Oxley Act and other regulations mandates restricted access to production systems. This has forced organizations across industries to conduct testing in separate environments. Further, it is not uncommon for organizations that cannot afford large-scale test environments to use sizing or extrapolation techniques during performance testing to understand how an application behaves in the real world. This approach, however, leaves too much room for potential problems after implementation.
environments and/or functional testing that relies on stubbed environments, resulting in projects being signed-off with numerous caveats and application problems in production. In contrast, by continuously providing environment availability metrics through daily standardized reports, TEMS allows teams to book the environment in advance. Further, through proper scheduling with set start and end dates and institutionalization of the release management process, TEMS helps deliver fully tested projects on time.

- **Better utilization of infrastructure:** In the traditional model, approximately 50% to 70% of the technology infrastructure earmarked for testing is underutilized, according to both anecdotal and published reports. Through effective capacity planning, TEMS ensures that testing projects receive the right amount of resources and test environments are optimally utilized, prompting improved returns on existing environments. For instance, incidents such as over-provisioning, which result in additional costs, can be prevented.

- **Communication replaces commotion:** In the absence of ownership, test environment downtime and changes are often not controlled and communicated properly. This poses challenges in managing testing projects and achieving KPIs. TEMS creates an effective communication mechanism among the key stakeholders in the event of environment issues by deploying same-day text alerts, etc.

With a host of other benefits such as targeted problem management and root cause analysis; controlled access management; maintaining up-to-date inventory of environments and infrastructure components; compliance with security and standards; etc., TEMS offers organizations a one-stop solution to their test environment issues.

### TEMS Efficiencies

Figure 2 illustrates the QA project efficiencies that TEMS can generate, with the dark red and dark green bands representing the lowest (pre-TEMS) and best-in-class (post-TEMS) efficiency levels respectively. In our experience, most of the current test environment efficiencies fall in the D and E bands, with an average efficiency of 57%. Deploying TEMS has

<table>
<thead>
<tr>
<th>QA Environment Availability</th>
<th>Commonly seen Pre-TEMS (%)</th>
<th>TEMS Partly Implemented (%)</th>
<th>SLA and OLA in place (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Post-TEMS: Highly available managed test environments reduces project costs.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(91-100) A</td>
<td></td>
<td></td>
<td>95</td>
</tr>
<tr>
<td>(86-90) B</td>
<td></td>
<td></td>
<td>85</td>
</tr>
<tr>
<td>(81-85) C</td>
<td></td>
<td></td>
<td>57</td>
</tr>
<tr>
<td>(60-80) D</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(41-60) E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(21-40) F</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(1-20) G</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Pre-TEMS: Poorly managed test environments raises project costs.

Note:
1. Bands G to D indicate improvements to QA environment can be rapid as low availability indicator means the environment is unstable or poorly supported.
2. Bands C and B represent environments that get timely support and are stable, but require more processes and controls to improve availability.
3. Band A represents environments that are stable and their availability is high as committed SLA/OLAs are in place.

*Source: Cognizant Research Center*

Figure 2
increased the average efficiency levels to 75%, and has the potential to increase it to 95%.

However, it must be noted that the TEMS framework does not establish rigorous processes or controls and has lower SLA commitments when compared with production environments. Additionally, its scope is limited to the staging environment and does not include the change controls at the production environment and release management process. These limitations are by design of the service and separate TEMS from production environment service management. They add value by lowering the cost of operation and enabling the test environment to align itself with release management objectives.

Moving Forward: Embracing TEMS
A stable, reliable and flexible test environment is important for companies to cope with today’s testing demands. Poor test environment management affects delivery schedules and results in increased risk of subsequent software failures as well as additional business expenditure. TEMS has much to offer organizations seeking to overcome environment-related challenges in an easy and convenient fashion.

Organizations planning to deploy TEMS must find a partner that understands business challenges and has prior experience in offering managed testing services. Reputable consulting firms that have rich domain experience and the ability to work in multi-vendor scenarios should be considered to ensure a smooth transformation, create new process efficiencies and accelerate payback, culminating in instilling greater discipline in managing test environments. This development will help ensure that managers enjoy their testing assignments and assure application code quality and business continuity across the enterprise.

Footnotes

Bibliography
About Cognizant

Cognizant (NASDAQ: CTSH) is a leading provider of information technology, consulting, and business process outsourcing services, dedicated to helping the world’s leading companies build stronger businesses. Headquartered in Teaneck, New Jersey (U.S.), Cognizant combines a passion for client satisfaction, technology innovation, deep industry and business process expertise, and a global, collaborative workforce that embodies the future of work. With over 50 delivery centers worldwide and approximately 166,400 employees as of September 30, 2013, Cognizant is a member of the NASDAQ-100, the S&P 500, the Forbes Global 2000, and the Fortune 500 and is ranked among the top performing and fastest growing companies in the world.

Visit us online at www.cognizant.com or follow us on Twitter: Cognizant.