VIEW POINT

ON-PREMISES INFRASTRUCTURE UPGRADE AND MIGRATION TESTING APPROACH

Abstract

Digital transformation around cloud and its related services has changed the paradigm of testing. Most IT organizations have accelerated the implementation of niche technology elements. IT infrastructure and its management has moved from on-premises to cloud at an unprecedented pace. While cloud service providers recognize the need to ensure the quality of provisioned infrastructure, organizations are bound to continue with on-premises infrastructure due to challenges around compliance, regulations, data privacy, distribution, and security, to name a few. Hence testing of infrastructure changes in various layers like network, database, web, server, etc., has taken precedence over typical functionality testing.

This paper highlights the validation approach to perform on-premises infrastructure testing along with a four-layer priority matrix complemented with a three-tiered Scour-Rank-Classify model.

Infosys®
Navigate your next

Infrastructure: An Overview

The infrastructure of an organization can be quite diverse. Effort must be taken to understand and analyse the impact of either upgrading or migrating one or any of the diversified components. During the planning phase of such transformation programs, cross-collaboration among various teams is recommended. A hybrid (both functional and central) information gathering exercise for all surrounding systems will uncover any collateral impact. Some of the key areas prone to upgradation/migration initiatives are:

Understanding the Impact

An enterprise-wide application stack can range from tens to thousands of applications. These applications can have distinct objectives, and sometimes work in a coordinated/sequential and/or asynchronous manner to achieve business objectives. Some of these applications are exposed to customers (B2C) while others are limited, being used for business objectives (B2B). From a technology radar perspective, these applications can be implemented as web, desktop, middle-tier, application programming interface (API) services, etc.

All technical factors must be taken into consideration before planning the infrastructure program. Some key factors like environment downtime, availability of in-house infrastructure validation processes, adaptability of any infrastructure change across programs/projects, business impact (particularly for customer facing apps), tools, and technology expertise need to be studied thoroughly.

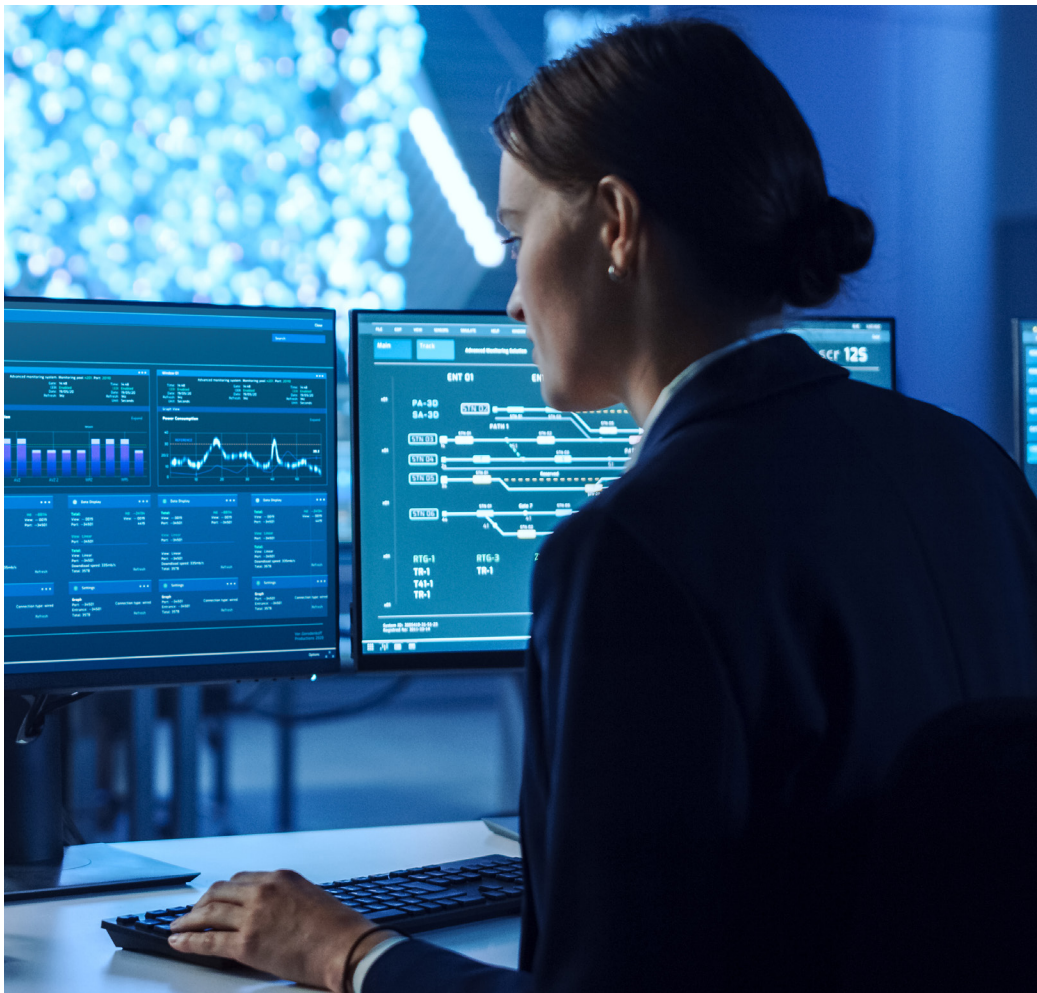
The Need to Test

A knowledge of 'why' infrastructure testing is needed will help in determining the 'what' and 'how' of it.

Infrastructure maintenance is necessary at the enterprise level to:

- Retain server certifications
- Ensure technology upgrades
- Bridge application-platform dependencies
- Support virtualization and migrations
- Guarantee scale-up and performance
- Satisfy business expectations
- Safeguard against quality and authentication vulnerabilities
- Maintain underlying system interactions

Based on the need, we recommend that the testing organization create a stack or roster of the necessary testing types. Some of the contextual types of testing are network connectivity test, installation verification test, system service checks, SQL server configuration checks, hardware and firmware checks, compliance tests, reliability tests, etc. Although, most cloud service providers (CSPs) will cater to the validation of some or most of these contextual features, it is imperative to set up a streamlined testing approach if the organization plans to keep the bulk of its infrastructure on-premises.



Setting up a Robust Testing Approach

Once the 'why' and 'what' of infrastructure is understood, a detailed approach is needed to meticulously include every application in the enterprise roster. It is possible for a minor change in the infrastructure stack to have a ripple effect on some or all of the enterprise applications.

A three-tiered **Scour-Rank-Classify** approach is recommended:

1. **Scour** – Collect, consolidate, and assess application inventory like third party applications, interfaces, protocol-based apps, in-house apps, COTS, etc.
2. **Rank** – Check the impact of application(s) based on parameters such as business, cost to maintain the application if infrastructure change does not happen, and the expected lifetime of the application. Once impact is analysed, applications are ranked ordinally.

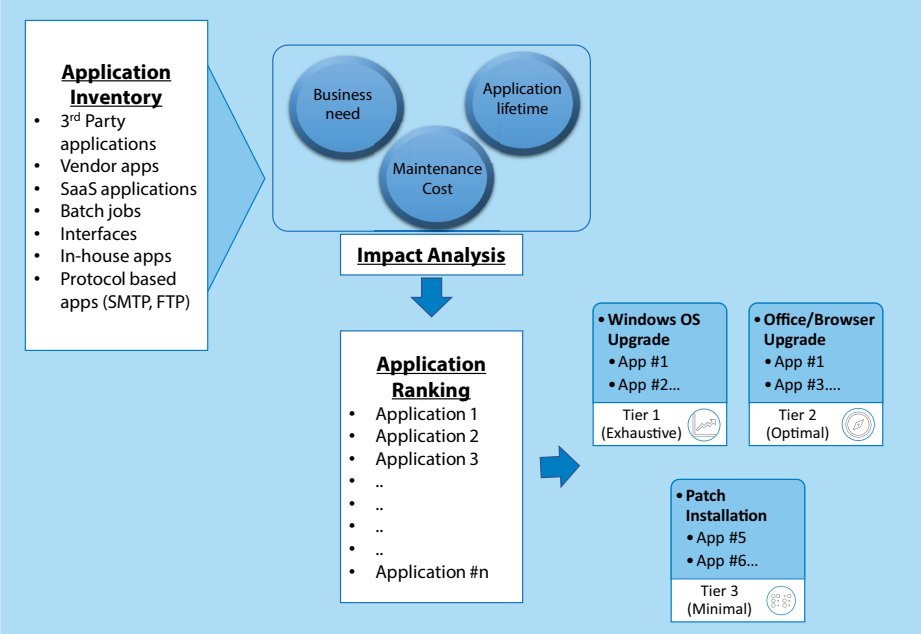


Figure 1: Application classification framework

3. **Classify** – Once the application inventory is collected and ranked based on the impact, these applications need to be classified into three tiers – exhaustive, optimal, and minimal

Devising a Four-layered Matrix

Once the tiers are classified, the appropriate testing services to be covered must be mapped. This can be done using a four-layered matrix comprising of must-have, should-have, could-have, and good-to-have testing services.

An illustrative matrix is depicted below. This will form the base of the infrastructure testing organization during the test planning and estimation phase.

	INFRASTRUCTURE STACK					
QA SERVICES	**Servers**	**O/S Network**	**Mid-Tier Systems**	**Discovery Services**	**Message Queues**	**Database**
Network Connectivity Test						
Installation Verification Test						
System Service Checks						
Monitoring Tool Test						
Remote Connectivity Test						
Database Availability Test						
Event Viewer Test						
Permission Test						
SQL. Server Config. Test						

LEGEND Must Should Could Good to have

Figure 2: Infrastructure – QA service testing matrix

Workflow Depicting Key Tasks

A transactional workflow view of this three-tiered solution is illustrated below. This acts as a flagstaff and guides the testing, programming, and infrastructure teams to arrive at the right levels of testing. It also improves the relevant test coverage during their migrations or transformations.

An end-to-end testing approach such as this will also effectively bridge any gaps in the infrastructure testing processes of the organization.

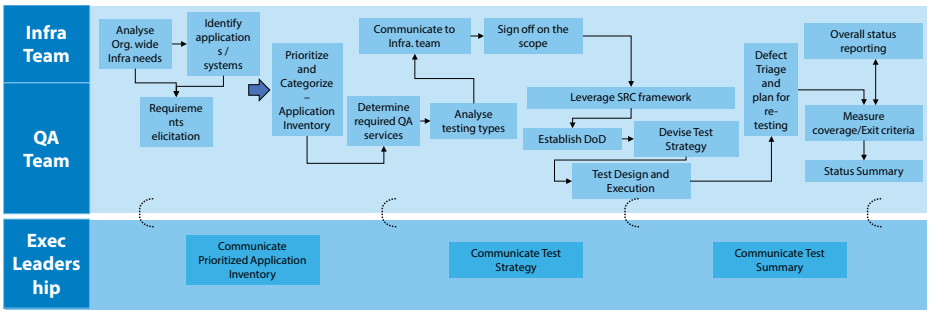


Figure 3: End-to-end testing approach

Conclusion

A thorough and comprehensive quality assurance strategy should encapsulate the infrastructure testing types and related variances mentioned above. Meticulous planning, understanding, and implementation of such a strategy will ensure emphasis on not just functional testing but also encourage organizations to look beyond traditional testing types. Furthermore, such testing practices can be combined with DevOps practices to carve out a niche for on-premises validations.

About the Authors

Venkatesh Jayaraman

Lead Consultant, Infosys Validation Solutions

Venkatesh is a Lead Consultant for Infosys Validation Solutions. His areas of interest include test process improvements, test automation assessments, and implementing best practices in test automation. He holds certifications from proficient IT bodies including PMI-ACP, PMI-PMP, ISTQB CTFL, CTAM, TMMi, AICPCU-AINS, CA DevTest, AWS-Cloud Architect, and Microsoft Azure along with other various relevant industry-vertical certifications.

Srinivas Chintala

Group Project Manager, Infosys Validation Solutions

Srinivas is a Group Project Manager with Infosys Validation Solutions. His key areas of expertise include account management, testing cost optimization, tool rationalization, and diversified QA service provisioning. He has strong domain knowledge of the banking industry, having handled several portfolios and anchored delivery of key programs for leading banks in Australia.

For more information, contact askus@infosys.com

Infosys®
Navigate your next