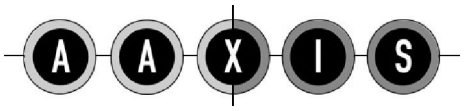
# Shared Services – QA

## Discussion & Approach




**AAXIS Group Corporation**

*1900 Avenue of the Stars, Suite 555*
*Los Angeles, CA 90067*
*Phone: 1-310-556-9700*
*Fax: 1-310-286-3100*
*Web:* www.aaxisgroup.com

# Table of Contents

Table of Contents ..... 2

1 Introduction ..... 3

2 Service Capability ..... 4

2.1 Technology Platforms ..... 4

2.2 Business Domains ..... 4

2.3 QA Functions ..... 5

2.4 Engagement Model ..... 7

2.5 Execution Mechanism ..... 8

2.6 Project Management ..... 9

3 Infrastructure ..... 11

3.1 Hosting & Data centers ..... 11

3.2 Software ..... 11

3.3 Administration ..... 12

4 Conclusion ..... 13

# 1 Introduction

Software Quality Assurance (QA) is a fundamental process area throughout the system lifecycle. During the development phase, QA ensures the verification of functionality, validation of system internals, and confidence to deploy the system for business use. The completion of development marks the transition to the maintenance and support phase; however, proper QA processes and resources remain critical till system phase out and disposal. In a large institution, the number of systems in the development or maintenance phase potentially number in the thousands. Each system may require its own testing environments, software tools, and QA engineers to provide adequate test coverage. This "silo" approach to QA may be the right mechanism in certain cases; however, the proliferation of this model across a global enterprise creates significant challenges, such as:

1. Inconsistent QA processes utilized across the organization
2. Dependence of individuals with intimate system knowledge
3. Procurement of excess software licensing and hardware
4. Ineffective work allocation to engineering resources
5. Requires a higher number of QA and support staff
6. Increased infrastructure management costs

A resolution path is to provide Quality Assurance capability via a Shared Services platform. Through the use of this mechanism, service centers across geographies can also be established to provide QA services to business units worldwide. The Shared Services approach aims to overcome the challenges of the "Silo" model, reduce overall Quality Assurance costs, and provide business units comprehensive QA of their systems. The strategy and implementation of QA Shared Services will require in-depth analysis and planning to realize the benefits after rollout. We at AAXIS believe we can provide clients assistance in establishing this Shared Services model. This document covers a number of areas which we believe are important and highlights dependencies and analysis needs.

# 2 Service Capability

Global corporations operate multiple software systems to service their business units around the world. These business units require a broad range of technologies from mainframes to client/server to distributed SOA. Providing a comprehensive Shared QA Services platform across an enterprise does have challenges due to the sheer extent of evolving business products and technology variation.

## 2.1 Technology Platforms

Critical to the success of Shared QA Services is to have technically competent QA/QC teams validating the functionality and operation of the system. Understanding of the underlying technologies used to implement the system is essential to deciphering whether the development team(s) used the right design approach or determining whether the delivered system will be capable of supporting its SLA requirements. With all the existing technologies which are currently in production at a multinational corporation, an analysis of what type of technology platforms can be or should be covered by the shared services is needed. Sunset systems which execute key business processes may not be suitable due to lack of experienced technical personnel or the inability of purchasing replica equipment which is no longer sold. This is not suggesting that legacy systems should be unsupported since there surely will exist cases where consolidation of certain legacy platforms to specialized QA centers will have clear benefits. The Shared Services model promotes the consolidation of scarce resources with specific technology skills that can be shared across business units. Furthermore, mainstream technologies, i.e. Microsoft .Net and Java Enterprise, to ERP systems, i.e. Oracle Financials, PeopleSoft, SAP, are platforms that should be supported at initial shared services rollout. This creates a larger candidate pool of systems that can potentially leverage the shared service capability.

## 2.2 Business Domains

In addition to technology platform knowledge, QA teams must understand the business domain of the systems which they are supporting. This is a challenge that is faced today as new QA resources are hired with general QA methodology and testing knowledge, but lack domain exposure. This domain gap is mitigated today since new members onboard to existing teams that already have a thorough understanding of the business and how the rules and processes manifest in the software system. During the initial implementation of the Shared QA

Service, the manner in forming teams and building both application and business knowledge should be considered. How should QA service centers become proficient in the domain? A common approach is relocating knowledgeable personnel to service centers to establish the knowledge base, or, the reverse, sending service center resources to the business unit. The success of this approach hinges of the effectiveness of the documentation of the person(s) which creates the knowledge base and the effective dissemination of the data. For some global institutions, business lines which are located around the world and in numerous time zones, it may be mandatory that domain knowledge of those specific business lines be present across multiple QA service centers - effective dissemination of the knowledge base is again paramount. A center of excellence for each business line can be established within the Shared Service and the knowledgebase/SMEs will be assessable across service centers to achieve a level of support suitable to the firm's global business.

## 2.3 QA Functions

The primary objective of a Shared QA Service platform is to provide the client with an effective mechanism of testing software systems to verifying their functionality and quality. A Quality Assurance methodology based on industry standards and best practices must be instituted early in the Shared Service rollout. This ensures process consistency across team execution, delivered QA artifacts, and an established system domain knowledge base. "Reinventing" a QA process within each business unit is unnecessary and often results in lower QA results. Furthermore, a systematic method of QA/QC incorporates required checkpoints and metrics which can be utilized to measure and audit projects to streamline and improve service processes and offerings.

Regardless of which engagement model is utilized for a given project, the ownership of ensuring system functionality and quality will be the responsibility of the QA team. This requires all project undertakings to establish the essential QA artifacts, such as, test strategy, test plan, test cases, test scripts, test procedures, etc., and receive approval and acceptance from business units. Test management and project scheduling of QA activities will depend on the engagement model, but execution of the system tests will be provided by shared services QA resources. Test execution can be broken into two core categories: functional and non-functional testing. Both categories of testing should be executed to effectively validate software systems. The tables below define the minimum set of testing capability that must be present.

**Functional Testing Capability**:

| System attribute | Description |
|---|---|
| 1. Black Box Testing | Verify system function points from the point of view of a business user and functional definitions detailed in system use cases |
| 2. Regression Testing | Verify that the existing functionality continues to work after introduction of new features or defect resolutions |
| 3. Integration Testing | Verify the functionality of the system when new software modules or external systems are integrated into the software solution |
| 4. Acceptance Testing | Verify the system functionality executes business process and rules as described within specification documents. Business unit sponsor must define the tests that the system must meet in order for it to be considered complete |

**Non-Functional (Technical) Testing Capability:**

| Type | Description |
|---|---|
| 1. Performance | Measure response times, transaction rates, and other time-sensitive requirements under expected production workload |
| 2. Stability | Evaluate the system's ability to continue to function properly under expected production workload for a long time period |
| 3. Capacity Planning/Deployment Sizing | Evaluate the system's ability to function on simulated production workload; and sizing the amount of system resources required to handle anticipated load. |
| 4. Fault-tolerance | Ensure that the system's backup resources are automatically allocated without interruption in the event of hardware/software failure. |
| 5. Security | Validation of security mechanism and frameworks |

## 2.4 Engagement Model

The Shared QA Services mechanism must be client focused by offering flexible engagement models that will accommodate clients' diverse business unit needs. Some possible engagement models may include: comprehensive "full" service, focused service, and infrastructure sharing.

In the comprehensive service model, the shared QA services team will engage business units with a "consulting" approach. The goal of the model is to own all QA related activities for a specific project or the entire business unit. Detailed analysis of software systems (new or existing) QA needs will be performed and matters, such as, cost, schedule, resources, and strategy, will be included in a proposed "outsourced" solution. Once the solution has been accepted by the business unit\project sponsor, the service provided by the comprehensive engagement model must be accountable for key performance metrics which include:

- Software quality
- On time delivery
- Customer satisfaction
- Team responsiveness
- Team capability

The next model – Focused Service – is to provide certain QA functions as standalone "ala carte" offerings. Business units may prefer to maintain ownership and overall responsibility of QA; however, they would like to augment or transfer certain activities to the shared service due to lack of expertise or manpower. There are a number of QA practice areas that lend themselves well to standalone offerings. Such services include:

- Functional Testing
- Performance Engineering
- Automation Engineering

Focused Service teams deployed for a client's business unit have the responsibility of completing their QA functions; however, the scoping of work, quality of the overall system, and ownership will remain with the business unit.

Shared service platform initiatives typically result in the establishment of enabling infrastructure managed from dedicated facilities and accessible across service centers worldwide. Physical servers, network applications, software licensing, system administration, help desk, etc. are provisioned and allocated to support project related work. Opening QA infrastructure directly to a business unit use establishes another engagement model and business value. Infrastructure sharing capability will enable business units to establish testing environments for their respective applications, utilize existing QA software suites, and reduce internal costs. Furthermore, a software system hosted in the shared service infrastructure may be readily accessible to QA shared services teams if the business unit decides, at a later point in time, to leverage technical capability and manpower within the shared services platform.

## 2.5 Execution Mechanism

Integral to the success of the QA Shared Service initiative is implementation of a reliable and repeatable execution mechanism. How should local IT staff interface with the shared service? Should service centers be specialized by business domains? How will work be allocated across globally distributed QA centers?

Communication bridges between QA Shared Service Centers and a firm's business units should be established to minimize potential project impacts due to different physical locations and time zones. Onsite staff may be present for larger projects/business units through the creation of local regional offices dedicated to facilitate and coordinate the work to be performed by service centers. Smaller projects or business units may not require daily onsite presence and may be adequately serviced through other communication means, such as, teleconference, video conferencing, web presentations, etc. In addition, software applications to streamline project execution and serve as information portals of project related artifacts and metrics may be purchased or custom developed. Fluid information flow between service centers and the client's business locations will allow remote centers to be transparent and as effective as a local onsite QA team.

Service centers must be formed to service the client's diverse business units and technology platforms. Segmenting centers into business domain coverage offers one strategy; however, certain business domains operate in only a small number of physical locations. It may be more effective to convert one location into a domain specific service center that will, in turn, service the other physical locations. For business lines that operate in a large number of locations, a domain specific service center must be able to service multiple global

locations under a variety of time zones. An alternative option is to create service centers dedicated to specific geographical regions, but this will require business domain cross-training and expertise within each center. Additional factors, such as, language requirements (Japanese, Chinese, French, etc.), specialized technologies (Cray, Teradata, legacy mainframes, etc.), and software packages (Oracle Financials, SAP, Cognos, TIBCO, etc.) may also affect how service centers are created and staffed. An analysis undertaking should be performed to formulate the approach to establishing service centers based on the factors above and how project work should be allocated to each center. Significant cost saving can be achieved by effectively allocating QA staff and prevent overstaffing on projects that have fluctuating work loads.

## 2.6 Project Management

A project management organization must take ownership for defining and maintaining the standards for the QA shared services initiative. This organization is the source of producing process documentation, providing project guidance, and establishing metrics to quantify the effectiveness of all projects. Best practices and industry standards should be embraced and provide the foundation for the processes adopted. The following key project process areas must be defined and actively managed by PMO:

- Project initiation
- Resource management
- Scope management
- Change management
- Cost management
- Risk Management
- Schedule development
- Communication Management
- Status Reporting
- Project Closure

In addition, the PMO should create a compliance group which enforces process adherence across the shared service projects and centers. This group monitors and audits ongoing and completed projects and establishes the performance and project related metrics. The data points collected provides executive insight into

effectiveness and value of the QA shared services initiative and identifies areas that can be improved though corrective measures, such as, process reengineering, employee training, or integration of additional support tools.

# 3 Infrastructure

## 3.1 Hosting & Data centers

In the creation of a shared services platform, a sufficient amount of infrastructure may be procured or transferred to support QA activities. A range of different physical servers, data sources, and network appliance must be hosted in select data centers which bridge QA service centers to business units around the world.

As additional business units utilize the shared services platform, cost-effective models for provisioning, for each software system, should be established and adopted. Virtualized server pools and physical server pools can be provisioned to meet QA needs without sacrificing effectiveness. For instance, QA activities which do not require high performance, such as, functional testing, can be provisioned through the use of virtualization technologies like VMware. Performance-oriented activities can draw from the physical server pool which can be allocated using a reservation model.

Consolidation of QA infrastructure to select data center will directly result in multiple project leveraging hardware resources concurrently. Strict security policies and controls must be in place to ensure data protection and restrict unauthorized access. Project personnel should only have access to project related resources and sufficient auditing and monitoring procedures must be in place to detect violations.

## 3.2 Software

In conjunction with physical hardware resources, software licenses must be purchased or transferred to facilitate QA activities. The follow table is an initial list of software categories:

1. Configuration management
    - Source repository
    - Defect tracking
2. Automation testing
    - Load testing
    - Functional testing

3. Data Sources
    - Message Bus
    - DBMS
4. Application Containers
    - Java Application Servers
    - Microsoft Windows Servers
5. QA tools
    - CPU/Memory profiling
    - Code static analyzers
6. Required product frameworks
    - Security products
    - Imaging products
    - Rules Engine products
7. ERP solutions

## 3.3 Administration

QA service centers will be located in a variety of countries and service business units around the world. This directly requires infrastructure administration and help desk support teams to be available. Shared Services platform should be able to utilize existing production support processes; however, it may require a separate dedicated support team to ensure knowledge of QA infrastructure resources and improve response times. The support team should be capable of performing the following:

1. Virtual/Physical server provisioning
2. Network maintenance/support
3. Physical server maintenance/support
4. Database maintenance/support
5. Software license allocation
6. Troubleshooting support
7. Security management and monitoring
8. Application deployment and configuration

# 4 Conclusion

In conclusion, enabling QA Shared Services requires strategies and approaches on a number of critical areas including: technology platform, business domain, engagement model, execution mechanism, project management, and infrastructure. There are clear limitations in the "Silo" model to Quality Assurance and instituting a Shared Services model allows clients to gain significant benefits and ROI, such as:

1. Standardized QA processes across the organization
2. Improved software quality and QA results
3. Consolidate domain and technical expertise
4. Reduced requirement of infrastructure and human resources
5. Efficiently allocate QA staff based on project work load
6. Substantial overall cost savings

AAXIS has successfully helped our clients design and implement dual-shore, shared QA service models to concurrently support multiple business units. Our clients were able to maximize the use of physical infrastructure, enable a 24x5 QA testing cycle, and effectively allocate QA resource based on project workload. With our experience in Outsourcing and Quality Assurance, AAXIS Group can assist in formulating and establishing a QA Shared Services platform for our clients.