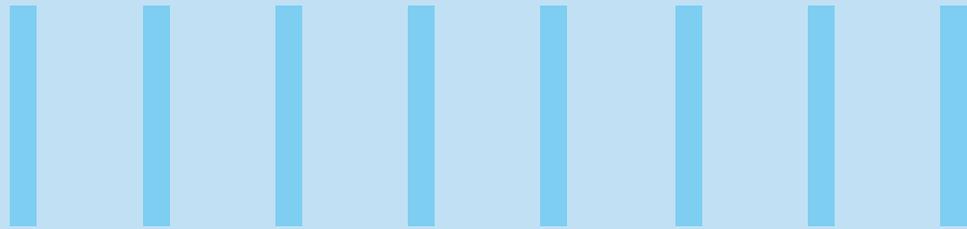




ENHANCING QUALITY ASSURANCE AND TESTING PROCEDURES

- Mayank Jain
Principal Consultant



Introduction

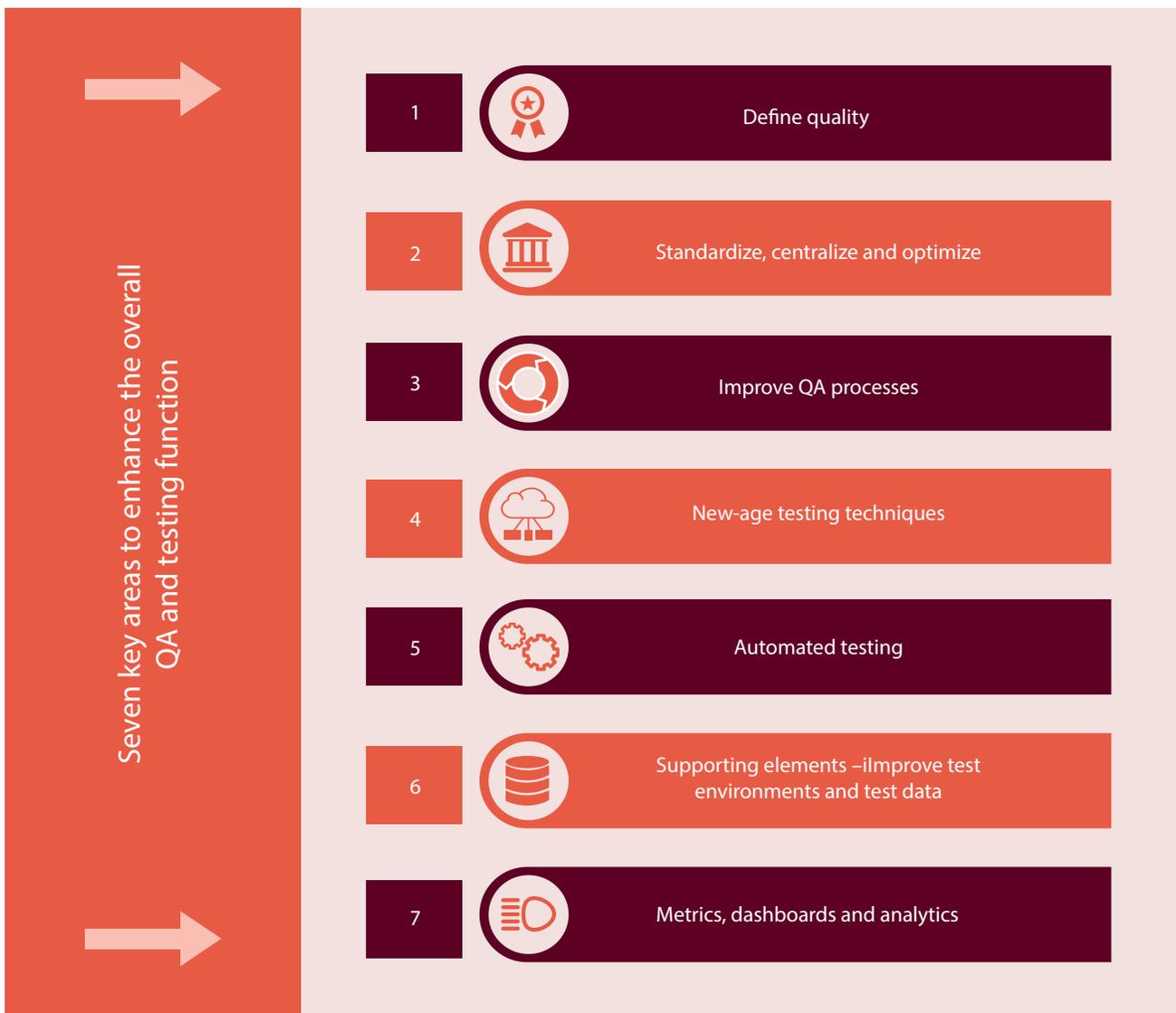
In today's world, most large and mid-size organizations have opted to centralize their software quality assurance (QA) and testing functions. If you are part of any such dedicated QA and testing team, looking to learn more about the latest QA trends, then, this paper is for you.

The Software World Quality Report, 2015-2016 indicates an average expenditure of 26% in 2014 and 35% in 2015 on QA

and testing. In fact, many organizations began allocating a yearly testing budget since about a decade ago, or even before. These budgets are allocated for the actual testing, the testing processes, procedures, tools, etc. But what happens if the defined processes are not implementable or the teams find them to be outdated and teams are unable to stick to these standards? The obvious outcome in such a scenario would be a decrease in QA effectiveness, increase in time taken and team frustration – all

leading to lower production quality. That's why testing processes need continuous review and enhancement, more so with newer technologies and shorter sprints (idea to production).

In this paper, I have outlined seven key areas that the QA and testing function must focus on to enhance their organizational maturity and bring innovation in their day-to-day work.



Defining quality

The ISO standard defines quality as “the totality of features and characteristics of a product or service that bears its ability to satisfy stated or implied needs.”

The important part of this definition is the conformance to requirements and the expectations from the quality assurance function. Quality, contextually, depends on the organizational setup, business demands, and the inherent nature of business competition in the era of mobile and increased social interaction.

In my opinion, the first step to improve quality should be to understand the expected level of quality. Accordingly, a decision can then be made whether to establish a dedicated testing function or simply follow a federated model. In both cases, a basic discipline to ensure the software quality processes and periodic enhancement to methodology and lifecycle, procedures, etc. should be instituted. Such discipline will ensure that products and services satisfy the stated or implied needs.



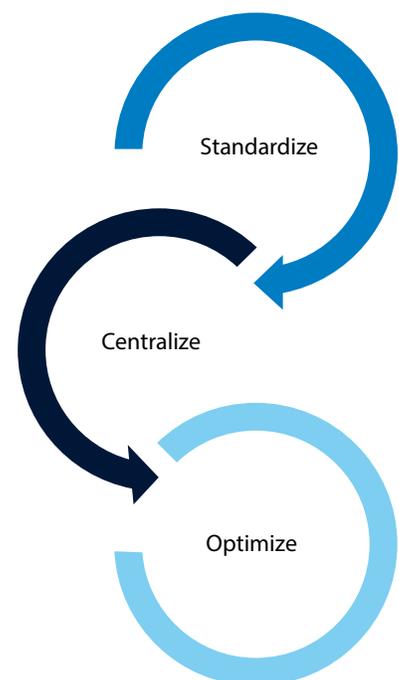
Standardize, centralize and optimize

As discussed, depending on the needs and expectations from the quality function, organizations can choose to either centralize their quality function or not. Let's discuss the challenges and the possible steps to address them in case the organization intends to centralize their QA and testing function in a Testing Center of Excellence (TCoE).

Challenges: TCoE processes are time consuming and expensive. In addition, often times, the application development team spends a large portion of their time explaining requirements or creating various builds for QA. These challenges are magnified when an organization has

multiple lines of businesses (LOBs) and lacks common grounds to leverage each LOB's capabilities and strengths.

Solution: Think of a scenario where each LOB follows similar processes! Would this not help integrate more easily? In my experience, it certainly would. So, the sequence of centralization should be to first standardize processes and tools for each LOB, then, proceed with centralization. These foundational steps will ensure optimization of idle resources, tools, tool licenses, and lower total cost of ownership. Dashboards can provide critical QA analytics here.



Improve QA processes

- **QA and test maturity assessment:** To baseline and improve the organizational QA capability, it's recommended to measure the maturity of existing processes and tools.
- **Test governance / clear policies:** Just like you cannot navigate to a new place without a map, QA teams need clear direction in term of the test methodology, how the testing lifecycle aligns with the development lifecycles, and the responsibilities of a tester, test lead and test manager.
- **Test management process (TMP):** TMP is an artifact that can be developed at the organizational level and individual lines of business or application areas can develop their specific test strategy. For example, the strategy can outline

its strategic direction on what areas would get automated, which customer facing applications would be piloted for security testing, and which applications would go mobile and be hosted on the cloud. These may be piloted with mobile- or cloud-based testing. The executive and operations committee, once instituted, should liaison between the business, application development, and the operations teams to align QA and testing methodologies with them.

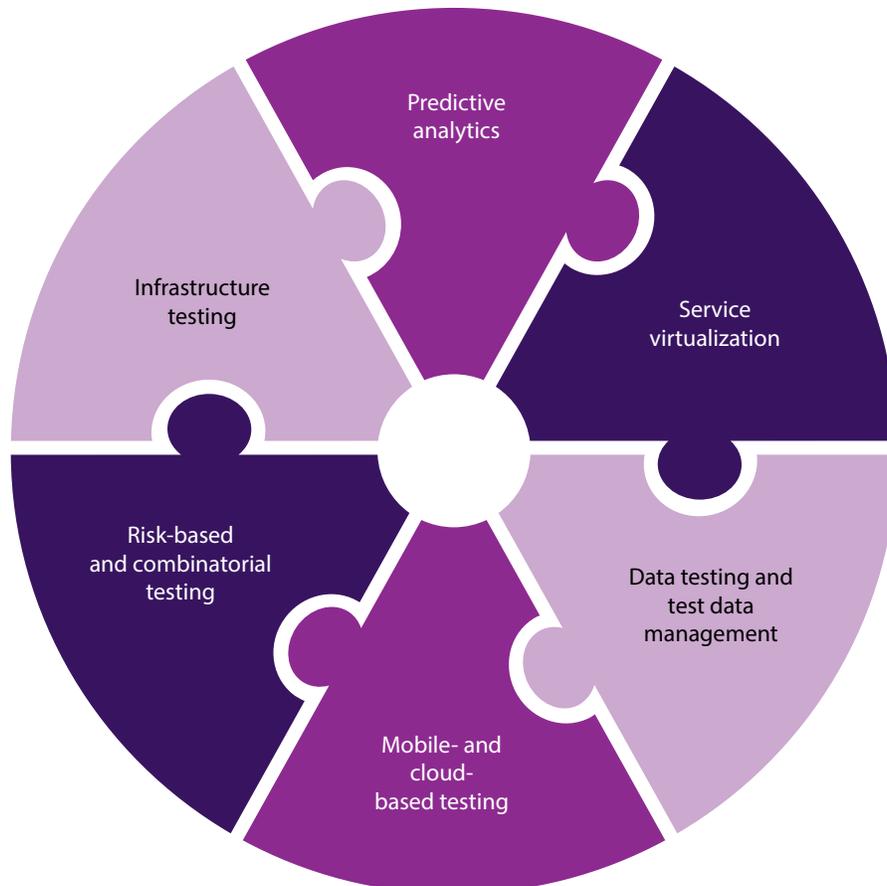
- **Shift left and get the requirements right:** It's proven that a shift left strategy in the software development lifecycle (SDLC) helps find issues earlier. The industry is moving towards using a single application lifecycle and finding ways by which different teams can increasingly collaborate and become agile to respond to each other's needs.

Reducing requirement volatility and developing agile teams can significantly improve strained dialogues between business and IT.

- **True vs. hybrid – Agile and DevOps:** Again, this depends on how we define quality. If the need of the organization fits well with a hybrid agile model, then, the advocacy of true agile processes would be immature. To achieve reduced cycle time and quicker time-to-market, continuous integration, continuous development and testing concepts are commonly being used.
- **Smoke test / build quality:** Approvals, UAT support, and metric-based focus area for regression helps break silos with the development teams and the business (top down and bottom up approach).

A new-age testing saga

Many IT professionals often face the question – What's next? I am sure you must have come across such situations too. To answer it, here are the latest trends in QA that organizations can leverage to reduce the risk to IT applications:



Predictive analytics

Like other industries, predictive analytics and machine learning concepts are now being leveraged in software QA and testing as well. Most QA organizations accumulate huge amount of data on defects, and test cases prepared and executed. Just like Facebook can predict and show what you may like and Netflix knows what type of movies you may like, QA teams can now predict the type of defects that may occur in production or the error prone areas of an application or the entire IT landscape based on the production or past QA defects and failed test case information.

Service virtualization

In today's world, different teams and multiple applications under the same or different programs, often, come to a point where one team cannot develop or test if the second application is not ready. In such situation, it is best to adopt service virtualization. This concept is mainly based on the fact that common scenarios can be simulated using a set of test data, allowing interdependent teams to proceed, without having to wait.

Data testing and test data management (TDM)

A majority of organizations have immense data issues, including data quality, availability, masking, etc. In fact, the system integration and testing (SIT) and user acceptance testing (UAT) teams can enhance the effectiveness of testing by leveraging various test data tools available. Apart from the tools, test data management is becoming an integrated part of the shared service organization. Many financial organizations across the globe have dedicated TDM functions to manage their test data as well as support various teams to create test data.

Mobile- and cloud-based testing

Mobile devices are ubiquitous these days. Today's mobile world is not just about smartphones or tablets, rather it is pervasive with handheld devices in retail stores, point of service (POS) systems, mobile

payment devices, Wi-Fi hotspot, etc. The list goes on! In the QA world, these pose unique challenges. For example, these devices and applications need to perform at speed and in various network conditions while using different browser, operating systems, and many more such conditions. Club this mobile challenge with applications and data hosted in a cloud environment such as Microsoft Azure, Amazon Web Services, to name few and you have magnified the testing teams challenges by manifold.

Since most organizations are not really equipped with mobile test labs, these are some areas where they can tie up with various vendors to perform mobile testing. Another trend that helps overcome these challenge is the adoption of newer methodologies such as agile Scrum, test-driven development, behavior-driven development, and DevOps. However, most of these methodologies demand progressive automation or model-based testing concepts where testers may need to be reskilled to wear multiple hats.

Risk-based testing (RBT) / algorithm and combinatorial testing

RBT is not a new concept and we all apply it in almost every project, in one way or the other. However, depending on the nature of the project or applications, RBT can be tricky and risky. QA and testing teams need tools that can generate various permutations and combination to optimally test and reduce the cost. For instance, in mobile testing, you may come across many operating systems and browsers, hence, many permutations and combinations are possible.

Combinatorial testing is another techniques that has gained fresh momentum in recent years and organizations can now use tools to derive an optimal set of combinations when attempting to test a huge number of possible scenarios.

Infrastructure testing

The recent Galaxy Note 7 debacle costed Samsung millions. And this is not a stray incident. In fact, the list is endless, making it important to thoroughly test the infrastructure. Many organizations now have dedicated infrastructure testing teams working in the shared service model. It is recommended to review the infrastructure testing needs and ensure that the services are well aligned with the IT Infrastructure teams who provision the internal and external hardware needs such as VDI, Windows patches, databases, etc.



Automated testing

As per the latest QA trends, automation testing is now a necessary testing type as against being optional, 5-7 years back. Many leaders still question the value of automation – what is the ROI? How is automation directly benefiting, etc.? In my view, the key is to do it right. For instance,

- Allocate automation funding for applications and not seek funding from projects to develop new automation

scripts and maintain the automation framework

- Automate regression testing and not functional testing
- Establish application-specific regression baseline
- Perform impact analysis using predictive analytics as described earlier and plan for automated testing at the release level instead of on project basis or simply, funded

- Track automation ROI and coverage metrics and showcase the value of automation as compared to manual regression
- Adopt and enable existing automation to take up new methodology and technology as it is related with Agile Scrum, Test and Behavior driven development, and DevOps model (as discussed in the new-age testing saga section)



Test environment, data and security

The test environment, test data, and the overall IT security challenges are the most agreed upon and accepted challenges. However, many organizations find it very difficult to build multiple QA environments and replicate their production processes. While conducting a process maturity assessment, I was surprised to hear that, a bank spent more than USD1 million but failed to implement a production-like environment.

Part of the reason is a lack of planning and budgeting itself, key to effective testing environment. And if budgeted but not approved, such initiatives take back seat and the organization continues to solve reactively instead of proactively.

In my experience, successful organizations typically promote centralization of these two functions and form test environment and test data management teams. These teams are responsible for ensuring the right data in the right environment at the right time. Both of these functions can

benefit from developing an operating and engagement model that would allow them to get funding, manage service requests, and obtain the necessary access to applications, jobs and data.

For security testing, there are many tools available in market these days. But it is recommended to look for tools that can integrate with application development and testing tools as well as support cloud and mobile infrastructure.

Metrics, dashboard and analytics

Metrics and dashboard concepts are not new but how the data is collected, retrieved, processed, displayed, and finally, analyzed to make informed decision has surely changed. There are many tools now in the market that can integrate with many technology platforms and drill down the capability in a very interactive manner. Some tools that are gaining popularity are Tableau, Quick View, etc. Many organizations develop in-house tools or leverage SharePoint as their metrics tool.

Whatever the choice of tool, below are some key consideration that QA managers and leaders would find beneficial:

- Capture and communicate the key performance indicators (KPIs) to senior management on production and QA defects, engagement feedback, cost avoidance, and application level defect density
- Define project level vs. aggregated view of the metrics
- For multiple departments or lines of business, apply consistent database scheme
- Define standard folder structure in the available QA or test management tools
- Develop integration for analytics tool
- Define key metrics to track and ensure data accuracy and quality
- Ensure automatic generation and analytical capability to assist in decision making
- Develop QA specific predictive analytics, for example, production and QA defects data can be used to predict potential areas for rigorous functional or regression testing, an upcoming trend





Conclusion

In summary, it's more beneficial to know "what" is happening at the macro level rather than "why" it is happening at the micro level. While it's important to measure perfectly but due to huge amount of QA data such as count of test cases prepared, executed, effort consumed etc., it's more beneficial to understand the trend at high level and not the detailed statistics. This will help define the Quality in the context of an individual organization as opposed to industry standard QA definition. Such definition can then guide organization specific QA metrics to collect, new age testing types and methodology to adopt as well as any consideration towards automation, improvement in QA processes and any supporting elements such as test data and test environments build up.

Author

Mayank Jain
Principal Consultant

For more information, contact askus@infosys.com

Infosys[®]
Navigate your next

© 2018 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.

Infosys.com | NYSE: INFY

Stay Connected     SlideShare