

FRTB Regulation: A P3 Framework for Centralized Testing

By Bhuma Sunderaraman

Contents

1. Business Case for the Testing CoE	3
2. P3 (Plan, Process, People) Framework.....	3
2.1 Plans.....	3
2.1.1 Test Scripts	4
2.2 Process.....	5
2.2.1 Key Performance Indicators - Benchmarking	5
2.2.2Technology.....	6
2.2.3 Feedback	7
2.3 People.....	8
2.3.1 Using a Whole Team Approach.....	8
2.3.2 Being proactive	8
3. Benefits of Centralized Testing CoE.....	8
3.1 Three Pillar Alignment.....	8
3.1.1Increasing Application quality.....	9
3.1.2Reduction in Testing costs.....	9
3.1.3Right/Up Skilling	9
3.1.4Reduction in Time-to-Release	9
3.2Consistency in Testing	9
3.2.1No Ad-Hoc Testing	9
3.2.2Re-usability of Test Cases.....	10
3.2.3One-View Approach	10
3.2.4Review Progress regularly.....	10
4. Conclusion: Agility of the Testing Function	10

1. Business Case for the Testing CoE

Regulatory Reporting is a big part of Banking and Financial Services Institutions as they strive to be compliant with the current laws and regulations in several countries.

Implementing a key regulation like Fundamental Review of the Trading Book (FRTB) requires increased data, additional analytics and reporting requirements. It specially poses challenges in terms of systems because of

- Lack of integration within risk/pricing systems of trading desks
- Different risk systems across different geographies leading to challenges in data integration
- Inconsistent methodologies within risk and pricing systems
- Intensive calculations not supported by current risk systems
- Lack of flexible and scalable data architecture

To conform to the different and challenging requirements and meet the regulatory deadlines, the financial services industry needs software systems and applications that are robust, reliable, re-adjustable to new functionalities and requirements and support complex and composite business processes. The need is speed and agility with no or little compromise on quality of data and reports produced.

The IT Development Team will come up with code changes, enhancements, and new releases etc which have to be thoroughly tested before going into production/market with very stringent timelines.

Traditional siloed QA/testing processes cannot succeed due to the following reasons:

- They take a huge amount of time to report defects/leakages because they are re-inventing the wheel for each project.
- Development Teams don't get feedback on time so that they can avoid writing defective code for future releases/enhancements
- Business Requirements keep changing per the regulatory guidelines. Applications have to keep up with the rapid pace and there is almost very little time for Testing before meeting the deadlines. .

Software Systems/applications having defects, and leakages could very well be compliance issues resulting in exorbitant fines for the Financial Institutions.

Business Units, the Developments and the Testing Teams need to align together to achieve the **common goal of the organization – "Compliance with FRTB"**.

The need of the hour is a centralized Testing Function, a Centre of Excellence (CoE), which can leverage people, processes and test plans, scripts across platforms to deliver quality results within the required timeframes and at a reasonable cost. The P3 framework lays down the roadmap for setting up a Centralized Testing CoE in the existing organization without causing a tumultuous upheaval.

2. P3 (Plan, Process, People) Framework

2.1 Plans

All the test plans and scripts need to be written according to the company policy. The Testing Strategy comprises of both Confirmatory Testing and Regression Testing. The automation of Regression Testing is important so that fewer defects fall through during various releases.

2.1.1 Test Scripts

Test Plans will be devised to confront challenges and also to crystallize thinking. Test Plans should include

- Generating Test Scripts
- Executing Test Scripts,
- Logging/Tracking Defects
- Defect Resolution
- Production Release

Each Project will have a Testing Approach that should be documented. The Testing Approach will include

- Test Scope
- Test Plan
- Test Environment
- Test Timelines and Activities
- Test RACI Matrix (R – Responsible, A- Accountable, C-Consulted, I-Informed)

Traceability to the Business Requirements will be key when making the test plans. Defects Logged will also be tied in with the Business Requirements. The Test Plan Draft will be submitted for peer review and feedback, so that it can be fine-tuned to address the needs of all the participants in the project.

During the Testing Phase, the defects will be logged by different testers using a shared site. The Defects will be given appropriate Severity Levels: High, Medium and Low, depending on how it would affect production. This will enhance knowledge levels and also help in prioritizing the Defects and resourcing the resolution. The Defects, to be fixed, will be assigned to the appropriate team/team member. Priorities will be given according to the severity level and this will be discussed in feedback and Project Review Meetings.

Test cases will be written based on the type of projects. The approach to writing the test cases can be driven by different needs like (a) Business Requirements, (b) Business process/Reporting (c) Data



2.2 Process

The goal of the Testing organization is to ensure that the software fulfills its requirements. This could be either the Vendor Software or Application Software developed in-house by the IT Team. There should be Project Review Meetings both during and after Project Completion. This will ensure effective communication and feedback on various improvements that could have been made in the project and as learnings for the next one. This will also lead to Best Practice sharing within the team.

2.2.1 Key Performance Indicators - Benchmarking

Key Performance Indicators (KPI) are measures that help decision makers define and evaluate progress towards defined goals. KPI metrics translate complex measures into a simple indicator that allows decision makers to assess the current situation and act quickly. Testing organization should strive to create metrics that benefit the whole IT organization.

Performance Measures will be both qualitative and quantitative. There should be more number of quantitative measures so that the performance is apparent to the individual and the organization as a whole.

#	Key Performance Indicator	Measure	Description
1	Test Coverage – Traceability to Requirements	# of Test Cases/# of Business Requirements	This is a quantitative measure. It measures the number of tests that have been scripted vis-à-vis the business requirements that have been gathered. The success of this measure depends on the product knowledge as well as on the granularity of the business requirements that have been gathered.
2	Test Rework	# of Tests Rework/# of Test Cases	This is a quantitative measure. It measures the number of tests that have been re-assigned to the tester vis-à-vis the total number of test cases. This means that if any of the test cases cannot be reproduced, they will have to be reworked. The success of this measure depends on Product Knowledge and the judgment of what the software can and can't do.
3	Defect Leakage – Post Production Migration	# of Defects in Production	This is a quantitative measure. It measures the number of defects that have fallen through the testing phase and have affected production. This also reflects the quality of testing. The number should be low for the success of the organization.
	Defect Leakage - % in terms of Severity a. High b. Medium c. Low	# Defects by Severity/Total Defects found	This is a quantitative measure, again a part of the above measure. The High Level Severity would affect and stop production till the defect is fixed. The Medium Level Severity will affect but not stop production. This defect also should be fixed within a

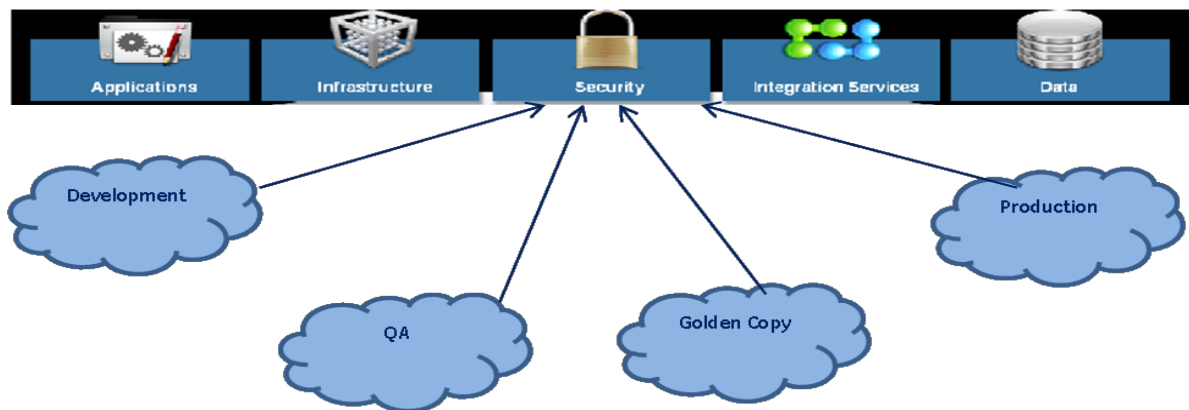
			small time frame. The Low Level Severity does not affect production at all. This defect can be fixed if needed within a much longer time frame.
4	Testing Cycle Time	# of days per Cycle	This is a quantitative measure. This measures the number of days taken per testing cycle. This would in turn depend on whether the testing is being done for a bug fix, new release etc. The success of this measure would depend on the automation of test cases and regression testing. This will enable in bringing new functionalities, technologies faster to the customer or business.
5	Customer Satisfaction	# of positive and # negative feedbacks received from the business	This is a qualitative measure. More number of positive feedbacks is recognition of the IT organization that they deliver what they promise at a reasonable amount of time.
6	Change Requests/Impact	# Change Requests/Impact should be classified as High,Medium,Low	This is a quantitative measure. Higher number of change requests shows poor quality of project design.

The Key Performance Indicators listed above, will be measured during each kind of Testing Cycle. This will help in evaluating the results of Testing and also measure test effectiveness in the Project.

2.2.2Technology

As part of Testing Logistics, building testing environments and having them ready is key to successful testing.

A Testing environment is a combination of hardware and software, on which the testing team is going to perform testing. It has to be completely different and not linked to production.



For successful release and deployments, one needs to have three sets of environments:

Development: Developers working on different product/application features work on the development environment. They can work on an environment set up on their local machines. The codes can then be merged making it ready for QA and Testing.

QA: Once the features are considered stable, they get merged to the staging branch and then get deployed to the QA Environment with a database refresh for QA and Testing. This will have real data, so that the testing can be done on near-production environment. Once testing is complete, and the bugs have been fixed with code changes, they are merged to be released into production.

Golden Copy: Environment will also be built where all the tested and defect free codes are merged with a recent database refresh. This is an important environment to maintain in case something gets messed up in Production and the Codes need to be rolled back.

Production: On the release date the development environment is merged into production and is deployed in the Production environment. The deployment will be made when the application is least active. The application will be monitored afterwards to see that it is stable and all the normal jobs are running fine. Make sure that the new features and fixes deployed work well in the production environment.

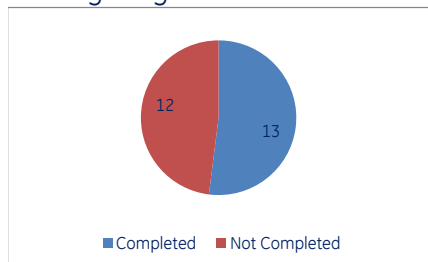
Sometimes, the deployment may not be successful and it has to be rolled back. The rollback should not break anything else and only the files deployed can be rolled back.

Permissions are also very important. While all developers can be given permissions to write/merge code in the staging environment, only a few senior developers should be given permissions in the production environment.

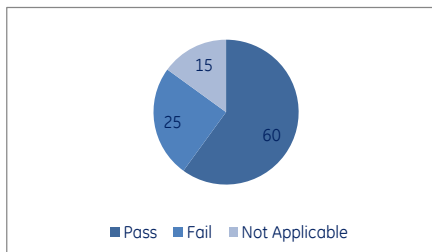
2.2.3 Feedback

A status dashboard should be created which will show the Progress of Testing Completion, Pass/Fail Percentages, Defect Status (Open, Closed, Resolved, Assigned, Deferred), Defect Severity(Critical, High, Medium, Low) Root cause Analysis (Environment issue, Bug) etc:

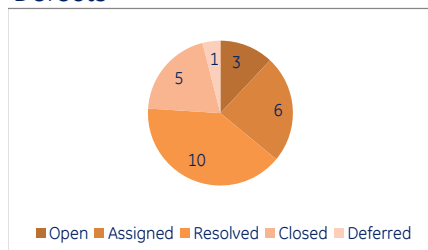
Testing Progress



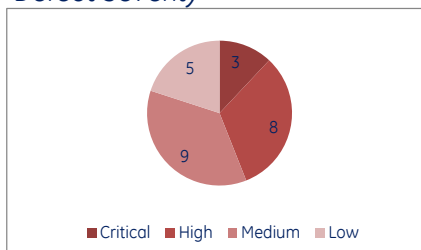
Test Cases



Defects



Defect Severity



2.3 People

There will be a list of people playing a key role in the development, testing and release teams for each of the projects that have been planned and need to be implemented. The Responsible, Accountable, Contributing and Informed Roles (RACI) for each of participants would be mentioned in the Testing Approach/Plan. The RACI Matrix for the Testing Activities will look like the following for each of the Testing Tasks:

Role	Name OF Person	Testing Cycle					
		Unit Test	QA	SIT	UAT	Parallel	Release
Application Owner		I	I	I	I	I	A
Project Lead(BU)		C	C	C	A	A	A
Technical Lead(IT)		R	R	R	C	C	R
Testing Lead(IT)		R	R	R	C	C	R
Data Architect(IT)		C	C	C	C	C	C
Testing BA(BU)		C	C	C	R	R	R
Subject Matter Expert(BU)		C	C	C	C	C	C

Key	
R	Responsible
A	Accountable
C	Consulted
I	Informed

2.3.1 Using a Whole Team Approach

The development and the testing team will work as a single entity and not at cross purposes. The ultimate aim should be to deliver results and not play a blame game or take unnecessary credit. When the entire team takes responsibility then we have a large pool of resources with variety of skill sets and experience to handle the different defects/errors that may arise. This also means working closely with customers and helps them to clarify their business requirements. This will in turn aid in testing and help in development as well. The “Three Pillar Alignment” involving a programmer, tester and a business expert is the foundation of the Centralized Testing CoE.

2.3.2 Being proactive

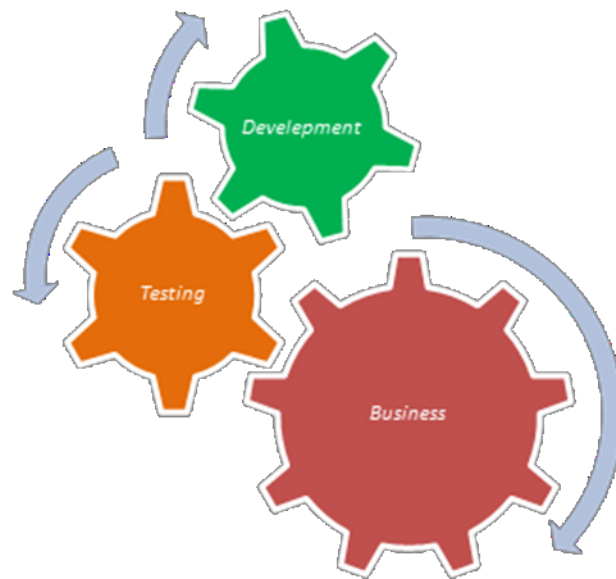
The people in the team will be proactive, creative and open to new ideas to complete their tasks. People should be willing to collaborate to help the team and the business succeed. There will be a drive to continually improve and find better ways to work. It will be good to also experiment with new tools methods and practices. No man is an island. Regular brainstorming with the team helps bringing people together to solve any issues or obstacles that may arise.

3. Benefits of Centralized Testing CoE

3.1 Three Pillar Alignment

The Testing function will align with the rest of the IT organization – the development and support teams. There will also be a strategic alignment with the other businesses/customers in the organization. This is the “Three

Pillar Alignment". The three teams will come together and finalize the **business requirements, design, build and testing phases**.



In the short run, this may be challenging each team may pulling at its own direction. This is a new approach at looking at things, and in the long run, will help in

3.1.1 Increasing Application quality

Business will frame detailed requirements and Developers will maintain adequate design documentation. The centralized testing team will leverage the common testing practices, test cases and tools instead of reinventing the wheel. This will increase the overall Application Quality.

3.1.2 Reduction in Testing costs

Testing costs will be significantly reduced as the resources will be shared across different project teams. There will be regular tollgate assessments. As the processes become more streamlined, there will be scope for improvement and further value addition to the Testing CoE.

3.1.3 Right/Up Skilling

The Testing team will work on different projects – right from 'keep the lights on' kind of work to complex applications. This will keep them motivated and they will make an effort to constantly right skill and up skill their knowledge.

3.1.4 Reduction in Time-to-Release

With the application of testing best practices and close alignment business and developers, the testing cycle will reduce. There will lower amount of defects and rework. This will reduce the time to release, projects can be implemented within the agreed timelines.

3.2 Consistency in Testing

Establishment of a Testing CoE will increase the consistency in Testing.

3.2.1 No Ad-Hoc Testing

The Testing Effort and rigor will be consistent with the Project and the Application Objectives thus eliminating slip testing and inconsistent test quality. There will be a formal framework for planning, execution of test scripts

and methodical defect tracking. Test Scripts will be part of the Standard Documentation thus sending the same message and understanding of what is being tested. This will increase the confidence and the predictability in testing. People will work as a team and not in Silos. The workflows, data flows and the interfaces will be known and so meaningful test cases can be framed for application expansion and enhancements.

3.2.2 Re-usability of Test Cases

The Test Cases will be lodged in a central library and indexed according to the Project type viz. Enhancement, Expansion, Data Quality, Sustainability etc. The test cases thus created, can be leveraged for other projects in the same application. There will be no re-inventing of the wheel for every enhancement/project. There will be a significant reduction in time taken for building the test cases for each project. Additionally, defect logging at a central library will help in having better test cases the next time round.

3.2.3 One-View Approach

Another major benefit of centralized testing is having 'One-View Approach' on the entire application. The system interfaces, data flows, data structures and data definitions are understood in the same manner across the Team. Hence there will be a better effort on the IT design, build and QA.

3.2.4 Review Progress regularly

The Testing Team needs to report and review the progress it is making. They need to execute test cases that are likely to find errors and also resemble actual usage. During the Testing Phase, there will be daily status calls at two levels:

- (a) Among Testers
- (b) Testing Lead and the Technical Lead.

The defects must be first discussed with the Testing Lead to see if they are really so. The defects thus found and fixed need to be logged at a centralized site, where it can be readily available for access at all times. This helps in enhancing application quality for subsequent development efforts and reducing the time to release as defects will reduce with the lessons learned from previous projects.

The Testing Status will also be communicated to the overall Project Lead and the Project Sponsor in Project Management Calls on a weekly basis. Key Risks should be discussed and Critical and High Severity defects should get attention and get resolved as soon as possible because they are the ones which could stop timely release.

4. Conclusion: Agility of the Testing Function

It is apparent that having a Testing CoE is vital for successful software project implementations in the Financial Services Industry. The P3 Framework ensures sharing best practices, resources and tools across the different projects for increased application quality, reduction in costs and faster release to the market. It will help in maintaining flexibility in supporting different technologies, developing various applications and bringing it to production in time to meet FRTB regulatory deadlines.

This framework is agnostic to the type of software or application. It's a methodology that can be applied to all Project/IT in Banking and Financial Services organizations without a major upheaval to the current state.

References

1. <https://www.uk.capgemini.com/resources/testing-centre-of-excellence-the-power-of-centralisation-and-standardisation>
2. <http://www.bellateq.com/whitepapers/07testingqualitycontrol.pdf>
3. <http://testobsessed.com/wp-content/uploads/2011/04/AgileTestingOverview.pdf>