Watch Your STEP !

Prabu Chelladurai

Senior Project Manager, Polaris Application Certification Enterprise (PACE)
Polaris Software Lab Canada Inc
5090 Explorer Drive, Suite 401, Mississauga, ON L4W 4M8, Canada
prabu.chelladurai@polaris.co.in

Abstract

Timely delivery of high quality software within budget is no more a nicety; but a necessity. Software testing, with a significant stake in enabling it, compels enterprises to focus on its improvement. However, the volatile nature of today's enterprises triggered by various factors like recession, attrition, technology change, competition and diverse culture impede any improvement effort and make the course and post-implementation of improvement feel more fragile than agile

This paper details a framework named STEP (Software Test Enhancement Paradigm) that provides ample and adoptable guidance towards improving and fortifying software testing. This proven framework has been enriched with best practices from industry standard frameworks like CMMi, TMM, TMMi, and TPI. This paper also outlines nine proven cost effective solutions based on technology and software process [with a case study] to remove fragility and induce agility in the journey of software test improvement. These aspects blended with STEP's unique ability to accommodate 2 flavors of test process improvement: Staged & Continuous and granularly calibrated measurement of maturity enable easier adoption, focused improvement, quicker realization of ROI thereby guaranteed Delivery of Quality !

Biography

Prabu Chelladurai is a Senior Project Manager at Polaris Software Lab Canada Inc and is involved in Software Test Management and Process Improvement for Polaris and its clients. He has managed large testing initiatives; calibrated software processes practiced by Polaris' clients and implemented STEP/TMM practices for the company's testing division named Polaris Application Certification Enterprise (PACE). He is a certified practitioner of CMMI, ITIL and Function Points (CFPS). Prabu has a Masters degree in Software Engineering from Carnegie Mellon University (CMU, Pittsburgh). While at CMU, he has researched and implemented the best practices of Extreme Programming and Scrum in various software projects.

1. STEP Overview

Software Testing holds a significant stake in ensuring high quality. Awareness of the same prompts enterprises around the world to invest a lot in maturing and improving the test processes to ensure their business objectives are met. But are they comfortable in achieving and sustaining high test maturity? Is there adequate focus on all the relevant facets/dimensions of testing? Do they know where their test maturity stands? More often than not, the answer is 'NO'.

STEP (Software Test Enhancement Paradigm) Framework was designed at Polaris Application Certification Enterprise (the testing arm of Polaris Software lab Ltd) to overcome these challenges. STEP Framework abbreviated for Software Test Enhancement Paradigm is business goal driven test process improvement framework. Its features include: focus on five comprehensive test dimensions, granularly calibrated maturity measurement, "Feel Agile. Not Fragile" tools/concepts and the flexibility of two distinct representations. STEP can be implemented as a standalone improvement framework. It can also be implemented in tandem with the likes of Capability Maturity Model Integrated (CMMi), Testing Maturity Model (TMM) and Testing Maturity model Integrated (TMMi) thereby amplifying the benefits. The STEP framework has been represented in figure 1 below:

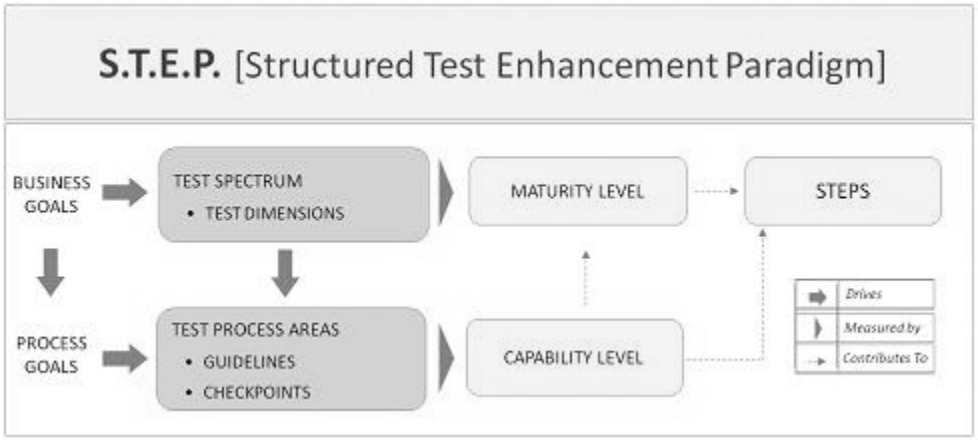

Figure 1, STEP Framework

a. Business Goals: Business goals are the key drivers of the STEP framework. They enable scoping of test spectrum and arriving at various process goals. An example business goal can be: Reduce Cost of Quality by 30%

b. Process Goals: Process goals are usually both qualitative and quantitative and are derived from the business goals. Example: Reduce SIT (System Integration Test) defect leakage by 25%

c. Test Spectrum: Test spectrum comprises of various test dimensions that are considered in scope for the STEP.

- Test Dimensions: Test dimensions are given below (examples of each dimension is given in the brackets '[ ]'):
    i. Type [Functional, Non-Functional]
    ii. Phase [Unit Test, System Test, SIT(System Integration Test), UAT(User Acceptance Test), PVT(Production Verification Test)]
    iii. Technique [Black Box, Grey Box, White Box]
    iv. Mode [Manual, Automated]
    v. Extent [Release Specific, Regression]

d. Test Process Areas: These are the individual units of the STEP Framework

- Guidelines: This is a sub entity of each test process area and provides the various guidelines, suggestions and recommendations for successful implementation of the respective process area. These guidelines can be customized by referring to successful best practices in the industry: CMMi, TMM, TMMi and past lessons learned
- Checkpoints: Checkpoints are sub entities of each test process area which assist application teams in ensuring satisfaction of the process capability requirements

e. Capability Level: This is used to measure the capability of a test process area. Capability level is on a scale of A to D based on the extent to which the particular Process Area is implemented

f. Maturity Level: This is used depict the maturity of the test spectrum pertaining to the Staged flavor of the STEP framework. Maturity level is on a scale of 1 to 5 based on the process areas that have been implemented

g. STEPS: 'STEPS' is a combination of Capability and Maturity Levels. In has a Max Value = 10.00

STEPS = Maturity Level STEPS + Capability Level STEPS
Where,
Maturity Level STEPS = Maturity Level of Testing [Maximum of 5.0 Points]
Capability Level STEPS = Capability Level Points [Normalized to a Maximum of 5.0 Points]

2. STEP Process Areas

The STEP framework comprises of 17 Process Areas that span across all facets of software testing. These process areas were primarily inspired from the Testing Maturity model (TMM) developed by the Illinois Institute of Technology [Illene Burnstein 2002]

1. Test Strategizing
2. Test Specification
3. Test Execution
4. Test Planning
5. Test Monitoring & Control
6. Test Environment
7. Defect Management
8. Test Organization
9. Test Training Program
10. Test Ware Management
11. Test Lifecycle & Integration
12. Verification
13. Metrics Program
14. Knowledge Management
15. Test Process Management
16. Defect Prevention
17. Test Optimization

Each of the above mentioned process areas have their own objective to fulfill. This paper does not detail each of them as the focus is on the overall STEP Framework and its applicability.

3. STEP Flavors

One of the shortcomings in most of the test process frameworks is the lack of flexibility in terms of the options for adoption. STEP Framework targets addressing the same with two flavors: 1. Staged Representation; 2. Continuous Representation. These two representations have been inspired from the widely successful CMMi framework

3.1 Staged Representation

Staged Representation flavor of STEP enables test enhancement in a staged fashion. 17 test process areas are spread across five levels (Maturity Levels). Each of the test process areas identified across the various levels in the framework, have a defined path for staged improvement. There are up to five

maturity levels [Figure 2] available: 1, 2, 3, 4 and 5 with maturity increasing in that order. The staged representation is available in TMM and TMMi

1) Level 1: Initial

This maturity level as the name indicates is where any test organization would start. There are no specific process areas targeted for this level

2) Level 2: Defined & Managed

This level targets at standardizing key testing components that cause an impact at project level. The process areas addressed include: Test Strategizing, Test Specification, Test Execution, Test Planning, Test Monitoring and Control, Test Environment, Defect Management

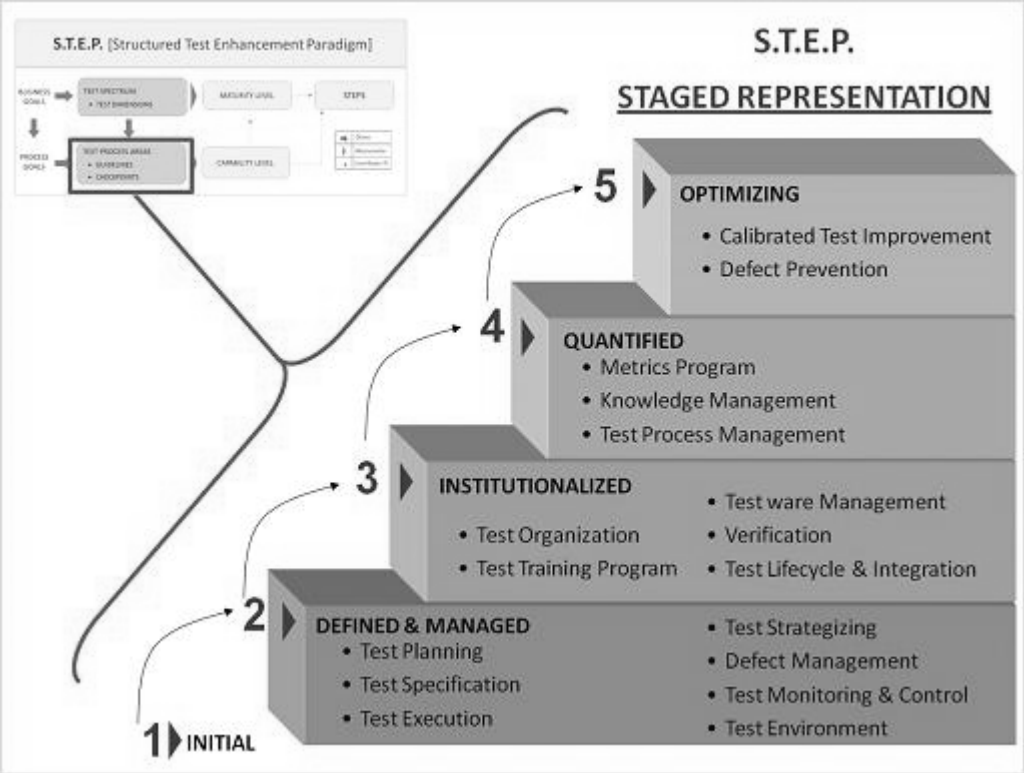

Figure 2, STEP Staged Representation

3) Level 3: Institutionalized

Maturity Level 3 targets institutionalizing the testing practice at an organization level and improving certain engineering aspects of Software Testing. The process areas targeted institutionalizing include: Test Organization, Test Training Program, Test-ware Management, Verification and Test Lifecycle and Integration

4) Level 4: Quantified

The crux of this level is the Metrics program. It also focuses on other process areas like Knowledge Management and Test Process Management

5) Level 5: Optimizing

This is the highest level of maturity focusing on continuous improvement. Process areas like Calibrated Test Improvement and Defect Prevention enable the objective of this maturity level

3.2 Continuous Representation

The test process areas are grouped in to 3 categories to enable flexibility for an organization. Each of the test process areas identified in the STEP Framework, have a defined path for continuous improvement. Each milestone in this path for continuous improvement is called as a Capability Level. This representation that is unique to the STEP Framework allows flexibility in picking and choosing the process areas.
There are up to 4 capability levels [Figure 3] available for each process area: A, B, C, and D with capability increasing in that order. The general criteria to achieve each of the levels are as follows:

- A: Beginners with quality investment (effort, time and money) to achieve goals
- B: Failed to achieve goals with agreeable variance
- C: Goals Achieved with compromises & unknown/new challenges
- D: Goals achieved without compromises, unknown challenges and is repeatable i.e. Feeling Agile!

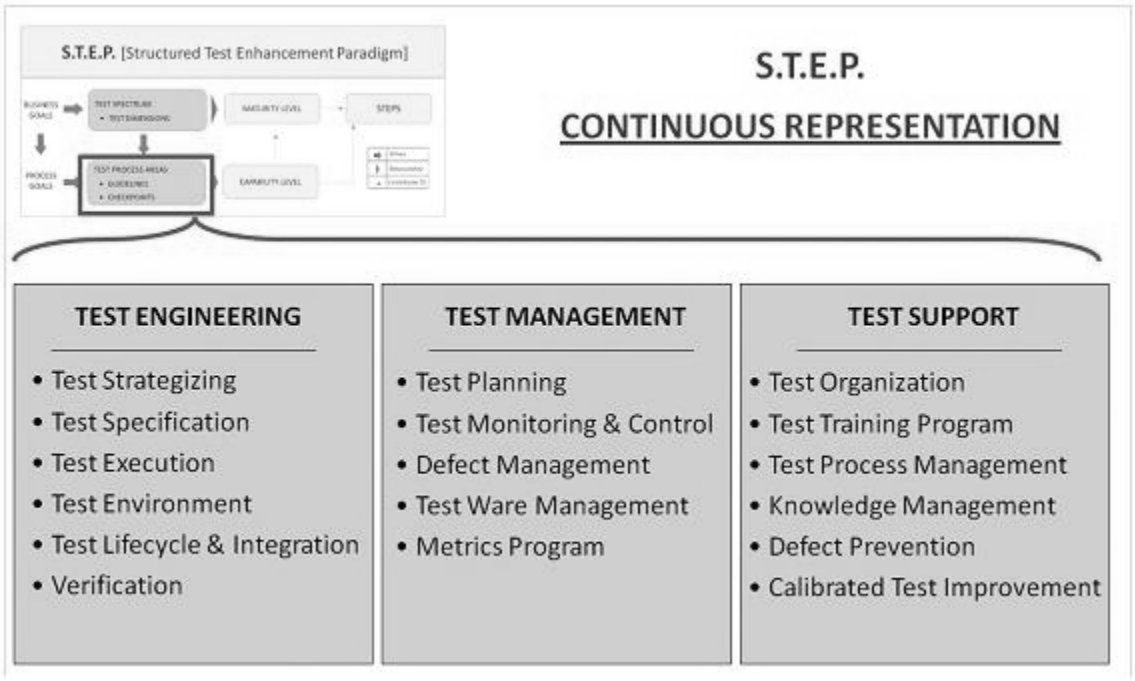

Figure 3, STEP Continuous Representation

1) Test Engineering
This group includes the process areas that focus on the engineering aspects of software testing. The process areas include: Test Strategizing, Test Specification, Test Execution, Test Environment, Test lifecycle & Integration and Verification

2) Test Management
This group targets the managerial aspects of software testing. The included process areas are: Test Planning, Test Monitoring and Control, Defect Management, Test-ware Management and Metrics Program

3) Test Support:
This group constitutes process areas that do not deal with core testing but definitely play a role in enhancing them. The process areas covered are: Test Organization, Test Training Program, Test Process Management, Knowledge Management, Defect Prevention, and Calibrated Test Improvement

The 17 process areas distributed across maturity levels and process Groups in Staged and Continuous representation line up as follows: [Figure 4]

CONTINUOUS / STAGED	TEST ENGINEERING	TEST MANAGEMENT	TEST SUPPORT
LEVEL 1 – INITIAL	-	-	-
LEVEL 2 – DEFINED	• Test Strategizing • Test Specification • Test Execution • Test Environment	• Test Planning • Test Monitoring and Control • Defect Management	-
LEVEL 3 – INSTITUTIONALIZED	• Test lifecycle & Integration • Verification	• Test-ware Management	• Test Organization • Test Training Program
LEVEL 4 – QUANTIFIED	-	Metrics Program	• Test Process Management • Knowledge Management
LEVEL 5 - OPTIMIZING	-	-	• Defect Prevention, • Calibrated Test Improvement

Figure 4, STEP Staged & Continuous Representation

4. Feel Agile, Not Fragile !

The volatile nature of today's organizations is triggered by various factors like recession, attrition, technology change, competition and diverse culture. These factors challenge the ease of achieving and sustaining high process maturity and subsequently hinder the intended objectives of the same. As a result the course of implementation and post implementation of process improvement feel more fragile than agile. This situation would be no different for STEP. Below are 9 practical & proven cost effective ways to remove fragility and induce agility in the journey of software test process improvement.

4.1 'Spectrum'ized Test Strategy

Usually when a strategy is designed with a blank mind there is a lot of room for gaps to creep in. Defining the test spectrum upfront by considering all possible dimensions will definitely help strategize in a structured and quick way. The possible dimensions that apply to most test strategies are as follows (These are the same dimensions that are used by the STEP Framework for configuring the Test Process Areas):

i. Type [Functional, Non-Functional]
ii. Phase [Unit, System, SIT, UAT, PVT]
iii. Technique [Black Box, Grey Box, White Box]

iv. Mode [Manual, Automated]
v. Extent [Release Specific, Regression]

For example, let us assume a test strategy is being developed for an application with the following characteristics.

- The Application has not existed before
- It is internal to the organization
- There will be no more releases or enhancements to this application
- The Application interfaces with many internal applications

The test dimensions to be addressed by the strategy will be as follows:

TYPE	*FUNCTIONAL TEST*		~~Non-Functional Test~~		
PHASE	*UNIT TEST*	~~System Test~~	*SIT*	*UAT*	~~PVT~~
TECHNIQUE	*WHITE BOX*	~~Grey Box~~		*BLACK BOX*	
MODE	*MANUAL*		~~Automated~~		
EXTENT	*RELEASE SPECIFIC TESTING*			~~Regression Testing~~	
Legend >>>	*FOCUSED IN TEST STRATEGY*			~~Not Focused in Test Strategy~~	

Process Areas Addressed:

- Test Strategizing (Maturity Level 2)

Key Benefit(s):

- Structured & Comprehensive Test Strategy with ease

4.2 Verifiable Test Estimation

A Close to accurate estimate is the foundation of successful Test Planning. But achieving it is not an easy task. One way to ensure accuracy of test estimates is to have a secondary estimation technique to verify the estimates arrived at using the primary estimation technique. The common techniques used for Test Estimation are

- *Function Point Analysis*
- *Test Point Analysis*
- *Use Case Point Analysis*
- *Program Complexity*
- *Activity based Estimation*

Depending on the form of requirements at hand it would be prudent to choose a primary and secondary estimation technique from the above list. Creation and use of simple software tools that implement these estimation techniques would simplify the task further

Process Areas Addressed:

- Test Planning (Maturity Level 2)

Key Benefit(s):

- Higher level of confidence in the estimates
- Accurate Test Estimates therefore less chances of budget or schedule overrun
- Quicker and Easier way to compute test estimates

4.3 In-Line Test Automation

Software companies invest in Test Automation. The investment bothers most because of the time it takes to realize the ROI. This often influences siding with Manual Testing alone. But if the Software Lifecycle is

slightly tweaked (to deliver code in iterations), then the ROI on Test Automation can be realized immediately.
Normal Scenario: Test Automation Scripts are prepared for Release 'N' and the same can be executed no sooner than Release 'N+1'
Agile Scenario: If the Code for Release 'N' is released in iterations then, the Automation scripts can be executed as soon as the next iteration in Release 'N'

Process Areas Addressed:

- Test Execution (Maturity Level 2)

Key Benefit(s):

- Quicker realization of ROI on Test Automation
- Significant reduction in Test Cycle time and proportional reduction in Time to Market

4.4 Zero-Effort Test Reporting

Test Reporting is always a critical and time consuming activity. The element of quantitative reporting just complicates things. One of the solutions to feel agile while still fulfill the reporting needs is by automating the process of test reporting. This is not as complex as it sounds. Simple macros that can function independent of the Test/Defect Management Tool are sufficient to achieve this. The diagram below [Figure 5] highlights a proven design that reduced the effort spent in Test Reporting to 'Nil'. *[Ref Appendix 1 for details]*

Process Areas Addressed:

- Test Monitoring and Control (Maturity Level 2)
- Defect Management (Maturity Level 2)
- Metrics Program (Maturity Level 4)

Benefit(s):

- Zero Reporting Overhead
- Better control of the Software Testing Activities
- Easier implementation of Metrics Program

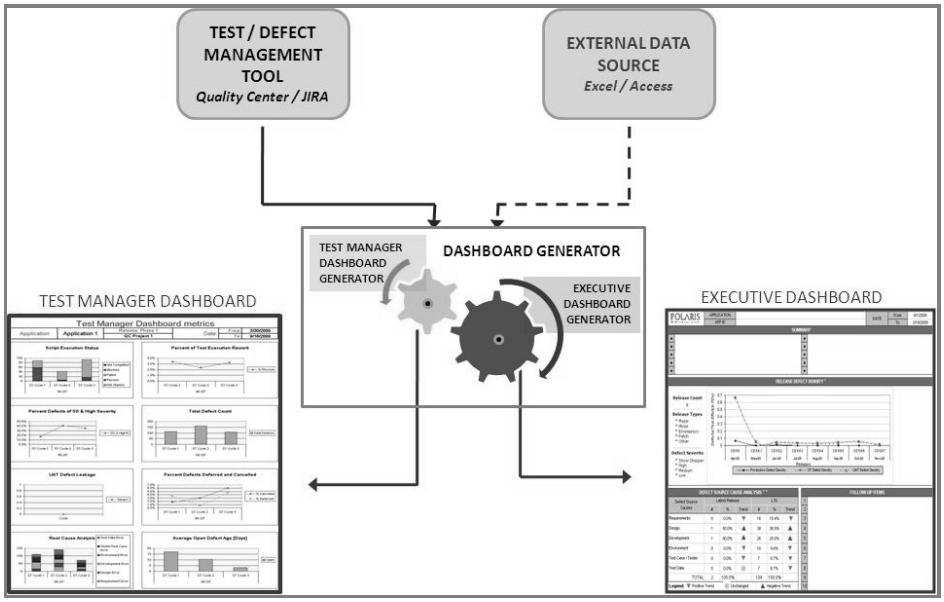

Figure 5, Test Dashboard Generator for Zero Effort Test Reporting

4.5 Sell Internally

If proper awareness is not created, efforts to improve processes are usually seen as more documentation. Hence it is a must to look at innovative ways to inculcate the importance of the process improvement initiatives. In other words, the initiative must be sold to the employees who are the practitioners and not just decided at the Executive level.
For Example, 'Review the Requirements Document'. Employees are usually reluctant to indulge in this. If you can say 10 Requirements Defects have the potential to cost at least USD 50,000, then the seriousness would be understood.

Process Areas Addressed:

- All process Areas (Maturity Levels 2 to 5)

Key Benefit(s):

- Process Improvement will not be looked at as Documentation overhead
- Process will quickly transition to habit

4.6 Test Early, Test Often

When to start testing? Usually testing gets pushed to the end of SDLC. But an ideal solution (in line with V-Model) is to test early and test often. Though this sounds impractical due to the team structure/dynamics, it will really help if 2 goals are pursued: Do More in parallel with development; Thorough Review of Requirements and Design.
Fact: Average Cost to fix one defect [Roger S Pressman 2003] rises significantly through the SDLC

Design Phase	: 1 Hour, USD 82
Construction Phase	: 6 Hours, USD 537
Testing Phase	: 15 hours, USD 1240
Production	: 60 Hours, USD 4959

Process Areas Addressed:

- Test Lifecycle and Integration (Maturity Level 3)

Key Benefit(s):

- Significant Cost and Effort Savings
- Reduced Time to Market and Defect Leakage

4.7 Maximize Coverage <> Maximize Documentation

This practice will definitely help remove the perception that process improvement is about more documentation. One of the ways to accomplish is to reuse test artifacts across different testing phases. This will enable better coverage in later phases. Usually the testing responsibilities of various phases lie with different group of stakeholders. Each group prepares test artifacts from scratch and misses on the coverage aspects instead end up with loads of duplicate test artifacts. If a small contractual change is made to make the test artifacts evolve through phases, then the degree of reuse and coverage automatically increase as well
Process Areas Addressed:

- Test Specification (Maturity Level 2)

Key Benefit(s):

- Effort Savings by Reuse of Test Artifacts across phases
- Enhanced Test Coverage and Reduced Defect Leakage

4.8 Increase Environment Availability

Test Environment best practices usually demand that the test environment must be a mirror of production environment. While this thought cannot be discarded, it is highly important to look at innovative ways to improve the availability and testability of the environments. Availability can be improved by adopting virtualization solutions or by developing harnesses to simulate the application components that are not available.

Process Areas Addressed:

- Test Environment (Maturity Level 2)
- Test Execution (Maturity Level 2)
- Test Lifecycle & Integration (Maturity Level 3)

Key Benefit(s):

- Reduction of impact caused by Test Environment Non-Availability

4.9 Calibrated Test Process Improvement

On a general performance scale, improving from 35% to 50% is much easier than improving from 95% to 97%. Achieving this can be eased if the improvement initiatives at a very high level of maturity are calibrated. One proven solution that can be adopted is the Process Calibration Framework [Figure 6].
The Process Calibration Framework [Muthu Gopal & Prabu Chelladurai 2007] quantitatively analyzes & improves the various aspects of process in lines of factors such as Performance, Stability, Compliance, & Capability using two components: Process Meter and Calibration Tree. *[Ref Appendix 2 for details]*

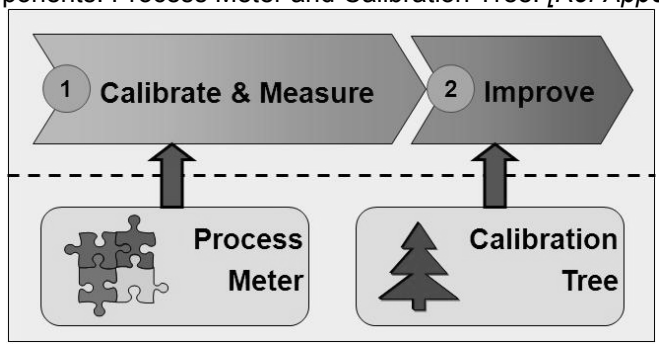

Figure 6, Calibration Framework

Process Areas Addressed:

- Calibrated Test Improvement (Maturity Level 5)

Key Benefit(s):

- Focused and Prioritized Improvement
- Quicker realization of ROI

5. STEP Vs Other Popular Frameworks

STEP framework compares with other popular process improvement framework across many attributes. ***Differences:*** Below is a table [Figure 7] that highlights comparison with some of the key attributes

Attribute	CMMi	TMM	TMMi	TPI	STEP
Exclusive focus on Software Testing	-	X	X	X	X
Flexible Representations for Adoptions	X	-	-	-	X
Continuous (Focus on Process Areas of Interest)	X	-	-	X	X
Staged (well defined evolutionary plateau towards continuous improvement	X	X	X	-	X
Enhanced Coverage with a Robust Multi-Test Dimensional Focus	-	-	-	-	X
Granularly Calibrated Test Maturity Measurement Framework	-	-	-	-	X

Figure 7, STEP Vs Other Popular Frameworks - Differences

Inspirations: Though the above table cites a few differences between STEP and other popular test frameworks used in the industry there are attributes in STEP which have been inspired from these frameworks and refined slightly (as needed). Some of the inspirations from the other frameworks are as follows:

- The concept of Staged and Continuous have been inspired from CMMi
- The 17 Process Areas are primarily inspired from TMM/TMMi. But following changes have been made based on experience in implementing them:

#	Process Area in TMM/TMMi	Equivalent Process Area in STEP	Change Type and Reason
1	Test Design and Execution	Test Specification Test Execution	**[MODIFICATION]** Process Areas have been separated as they have the capability to influence different aspects of Testing
2	-	Defect Management	**[ADDITION]** Process Area has been added as a Process Area as defects directly relate to quality
3	Peer Reviews	Verification	**[MODIFICATION]** Peer Review is a type of Verification and there are many other verification techniques that can be adopted. for example, Inspection, Delphi technique are commonly used review techniques
4	Software Quality Evaluation	-	**[DELETION]** Process Area objectives have been covered as a part of Verification [Level 3]
5	-	Knowledge Management	**[ADDITION]** managing knowlwedge is important for a level 4 company
6	Quality Control	Test Process Management	**[MODIFICATION]** The process area has been renamed to avoid confusion with CMMi Quality Control. Also, the process area has been moved from level 5 to level 4
7	Test Process Optimization	Calibrated Test Improvement	**[MODIFCATION]** The process area has been renamed to place emphasis on the need for calibration

Figure 8, STEP Process Areas Vs TMMi Process Areas

Though STEP can be a standalone test framework, the above inspirations enable it to compliment other popular frameworks and amplify the benefits

6. Case Study

6.1 Institution Overview

The institution under consideration is a division of a global bank that offers transactional services Given below are some facts about the division of the bank:

- Over $276 billion in average liability balances; Over $12.8 trillion in assets under custody; Over $3+ trillion in worldwide transactions daily
- Serving 96% of the world's Fortune 500 companies
- 10 regional processing centers worldwide
- Multiple Domains: Cash Management | Trade Services | Securities & Fund Services
- 93 Software Applications across multiple technologies supporting the above business

6.2 Problem Statement

The IT group of the division under consideration was facing the following problems with respect to Software Testing & Application Quality:

- Increase in Production Incidents
- Increase in support and maintenance costs
- Increased unplanned releases and patches
- Ineffective & Ad hoc Testing

Ultimately resulting in,

- Increased Cost of Quality
- Increased Time to Market
- Decreased Customer Satisfaction

6.3 STEP Solution

Upon an analysis of the problems and consideration of the business and process goals it was decided to adopt ***'Continuous Representation' of the STEP Framework with focus on Test Engineering and Test Management***. In lines with the decision, following process areas were chosen for improvement:

PROBLEM AREAS / PROCESS AREAS	Increase in Production Incidents	Increase in support and maintenance costs	Increased unplanned releases and patches	Ineffective & Ad hoc Testing	Increased Test Turnaround Time
Test Strategizing	X	X	X	X	X
Test Planning					X
Test Monitoring and Control					X
Test Specification	X			X	
Test Execution	X		X		
Test Environment		X	X		
Defect Management				X	
Metrics Program		X	X		
Test Training Program	X			X	
X - Indicates the which Problem Area(s) were primarily addressed by the Process Area					

Figure 8, Problems addressed by Process Areas

Maximum STEPS possible is 10.00. Considering only a few processes were chosen, the Maximum STEPS for the identified processes is 4.65. The number of STEPS in the AS-IS Process was found to be 1.66. Details of the analysis are as follows:

Maximum STEPS = 10.00

Max STEPS for Process Areas considered = 4.65
- *Capability Level STEPS* = *2.65*
- *Maturity Level STEPS* = *2.00*

STEPS in AS-IS Process = 1.66
- *Capability Level STEPS* = *0.66*
- *Maturity Level STEPS* = *1.00*

Following are the key activities performed in order to achieve the Target STEPS:

- Identify goals for the process areas identified for improvements.
- Clear set of guidelines and checklists were defined based on best practices from CMMI, TMM, TPI and past learning.
- Following 'feel Agile, not fragile' concepts were implemented targeting some of the process areas considered in scope:
  - 'Spectrum'ized Test Strategy
  - Verifiable Test Estimation
  - Zero Effort Test Reporting
  - In-Line Test Automation
  - Sell Internally
  - Maximize Test Coverage not Documentation

After 6 months of pilot implementation and 12 months of a steady-state implementation, the Target STEPS (4.50) were achieved except for a couple of process areas. Details of the calculation are as follows [Figure 9]:

STEP CALCULATION		AS-IS SCORE		Target Score		Actual Score	
Capability Level & Capability Level Points	Test Strategizing	A	1	D	4	D	4
	Test Specification	A	1	D	4	D	4
	Test Execution	B	2	D	4	D	4
	Test Planning	B	2	D	4	D	4
	Test Monitoring & Control	A	1	D	4	D	4
	Test Environment	A	1	D	4	D	4
	Defect Management	A	1	D	4	D	4
	Metrics Program	-	0	D	4	C	3
	Test Training Program	-	0	D	4	C	3
CAPABILITY STEPS		0.66		2.65		2.50	
MATURITY STEPS		1.00		2.00		2.00	
STEPS		1.66		4.65		4.50	

6.4 Benefits

Figure 9, STEP Calculation for Improved Process

The benefits reaped speak volumes for success of the implementation

- 70% Reduction in Sev 1 Outages [Figure 9] which enabled reducing support costs by 21%
- Additional potential annual savings of USD 500,000 via Automation of Test Metrics (for 75 Applications)
- 34% reduction in unplanned releases
- Streamlined & Robust Test & Defect Management Processes
- Around 15% Reduction in Maintenance & Support Costs

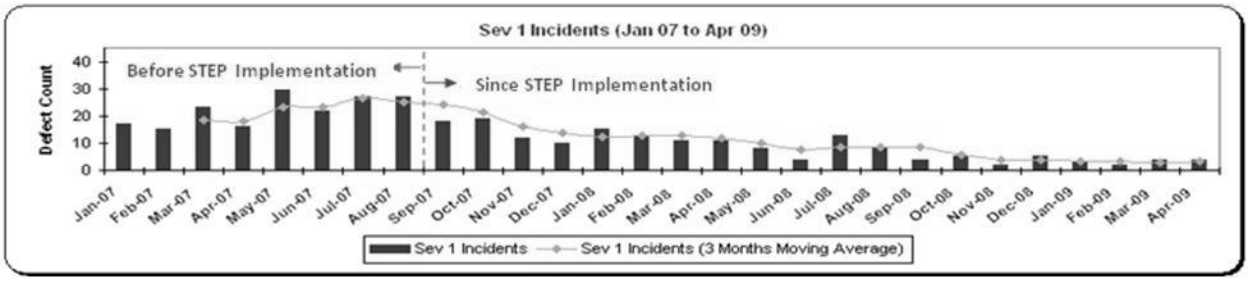

Figure 9, Sev 1 Defects from Jan '07 to Apr '09

Appendix

Appendix 1: Test Dashboard Generator

The Test Dashboard Generator advocated as a part of STEP includes the following features

- Comprehensive set of Test Metrics
  - Script Execution Status
  - Test Execution Rework
  - Percent of Severe Defects
  - Total Defect Counts
  - Defect Leakage
  - Percent Defects Deferred
  - Percent Defects Cancelled
  - Root Cause Analysis & Trend
  - Defect Age
  - Defect Density
- Dashboards for Senior Executives & Test Managers/Leads
- 100% End-to-End Automation
- No Separate License Costs
- Blends well with 'Any' Test Management Tool
- Extensively Customizable

Given below is a sample Test Manager Dashboard generated using the Test Dashboard Generator:

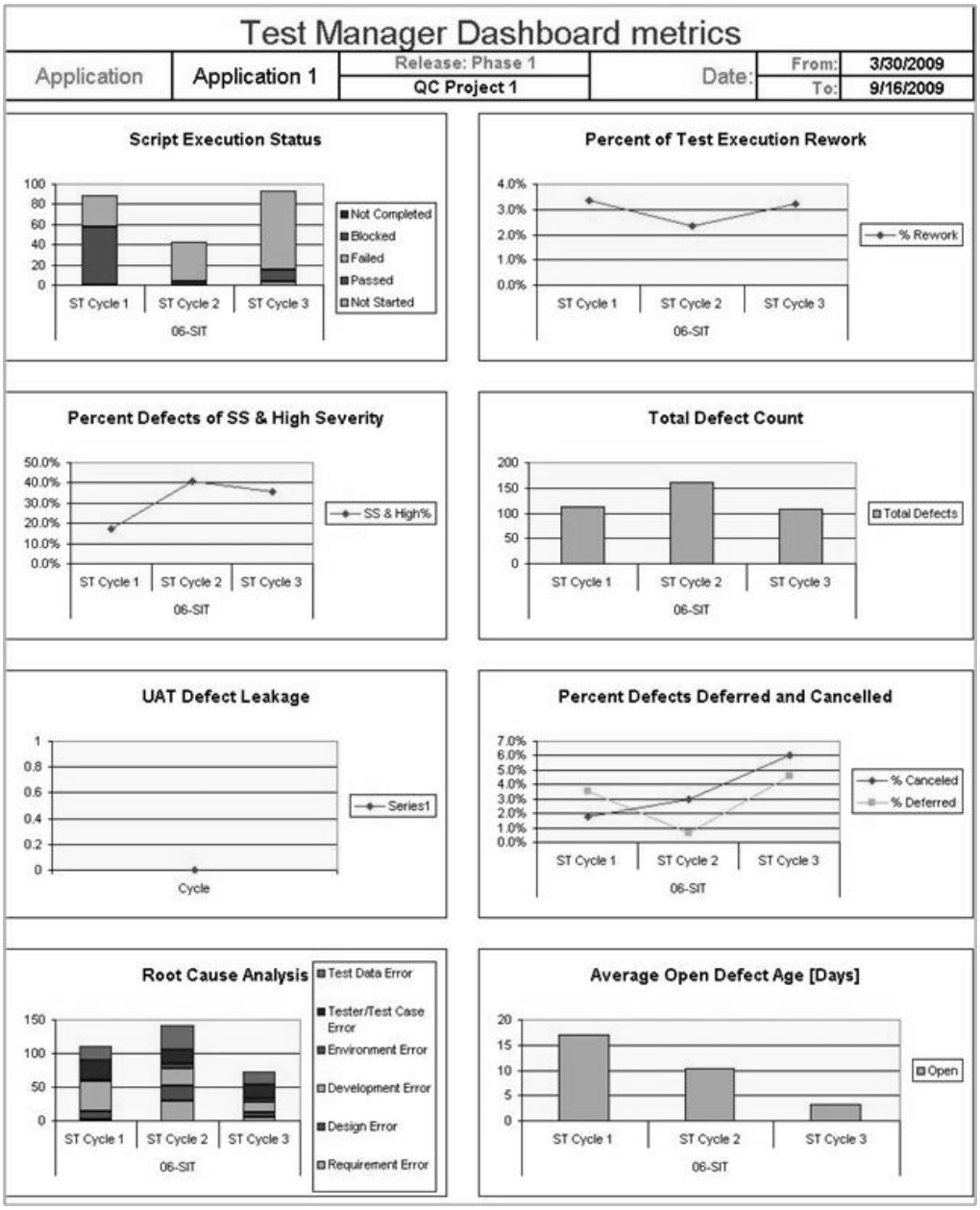

Figure 10, Sample Test Manager's Dashboard

Appendix 2: Process Calibration Framework

What is Calibration? - Calibration in general is associated with instruments and is defined as the act of aligning the instruments to a measurement unit(s) to enable it to respond in terms of the unit(s). For example calibrating a thermometer to Fahrenheit scale means the thermometer can be used to read temperature in terms of Fahrenheit. **What is Process Calibration?** Extending the definition of Calibration, Process Calibration can be defined as aligning the process to certain calibration factor(s) so that the health of the process can be measured with respect to the calibrated factors(s). **The Process Calibration Framework** quantitatively analyzes & improves the various aspects of process in lines of factors such as Performance, Stability, Compliance, & Capability using two components: Process Meter and Calibration Tree.

The process of calibration happens in 2 phases:

1. Calibrate and Measure [using Process Meter]

   This Phase involves the following Steps:

   - Identify Sub-Processes
   - Identify Calibration Factors. Some common factors include
     - *Performance* [The degree to which quantitative goals (quality, quantity, cost, time) of the process are met]
     - *Compliance* [The measure of cross conformance between the process and the requirements]
     - *Stability* [The consistency of the measurable attributes (quantitative goals) of a process]
     - *Capability* [The ability of the process to cater to the requirements]
   - Associate suitable Test Metrics with the identified calibration factors
   - Assign Ranks [0 is the minimum rank and 100 is the maximum rank]
   - Compute Process Health [0 is the minimum rank and 100 is the maximum rank]

2. Improve [using Calibration Tree]

   - The calibration tree has the score in the process meter as its root. Any score that needs to be analyzed forms the root of the calibration tree.
   - The score branches out into the attributes, which were responsible for the score on the process meter.
   - The reason behind the value of the attribute(s) is analyzed in lines of the 3As: Actors, Artifacts & Activities. This analysis helps interpret the strengths and areas of improvement in the process

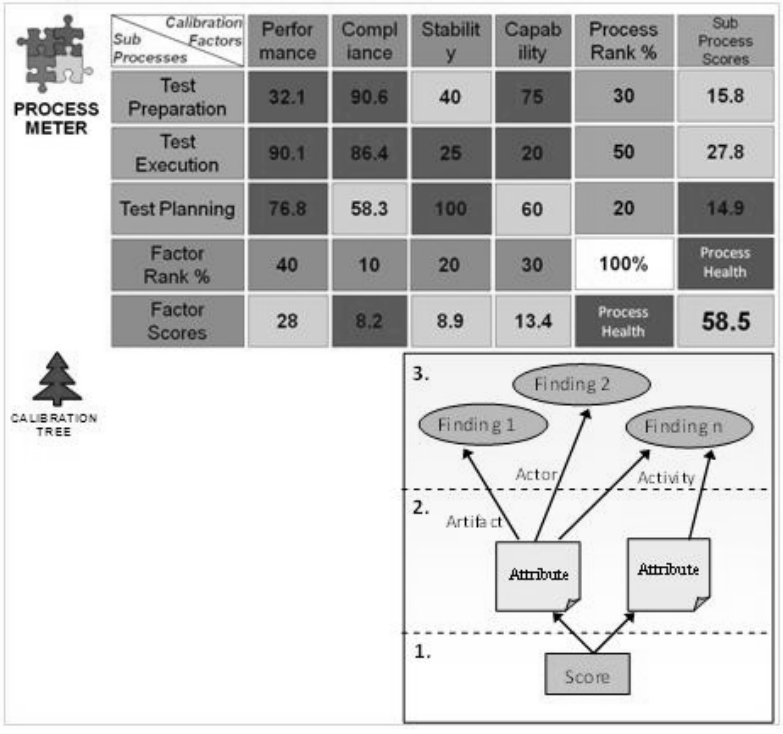

Figure 11, Process Meter and Calibration Tree

The above diagram (figure11) shows the calibration analysis performed on 3 testing process areas. The overall health is 58.5 on a scale of 100. This score is generally considered low in the calibration scale and hence can be improved a lot. In order to improve the overall score, we will have to target improvement of the weaker areas of the process. A more careful analysis of the process meter will help identify these weak areas. In this example, the performance of the test preparation process, the stability and capability of the test execution process are the key reasons for the low overall score. Now these identified areas have to be passed up the Calibration Tree to narrow down on the exact improvement effort

References

[1] Mary Beth Chrissis, Mike Konrad, Sandy Shrum. CMMi®: Guidelines for Process Integration and Product Improvement – 2nd Edition. Addison-Wesley Professional, 2006.

[2] Burnstein, Ilene. Practical Software Testing: A Process-Oriented Approach. Springer Professional Computing,2002.

[3] Muthu Gopal, Prabu Chelladurai. Calibrate Your Estimation Process presented. 2nd Annual International Software Estimation Colloquium by QAI, 2007

[4] Roger.S.Pressman. Software Engineering, A Practitioner's Approach - 5th Edition. McGraw Hill, 2003

[5] Eric Van Veenandal. Test Maturity Model Integration. www.tmmifoundation.org, 2010.