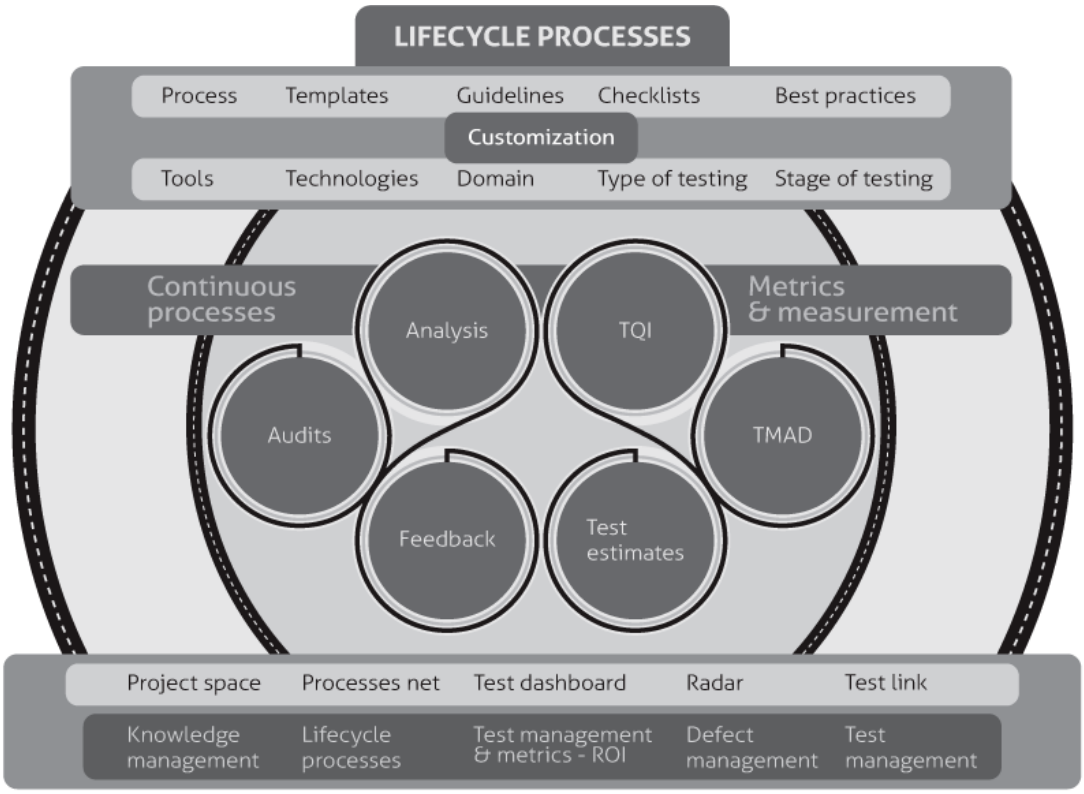
Mindtree
Welcome to possible

Integrated methodology for testing and quality management.

MindTest™ overview

MindTest™ is an integrated testing methodology that meshes all the components of a testing engagement, manages the quality of testing, and delivers measurable and predictable software quality. MindTest™ methodology collaborates in the areas of delivery, domain, technologies, tools, training, frameworks, quality, and knowledge management to meet the customer's objectives.

Mindtree's process evangelist team drives the MindTest™ methodology within the organization.



MindTest™ guidelines help tune standardized test processes to a specific testing engagement. These guidelines aim at improved productivity of the tester. They apply a 360-degree view to continuous improvement, resulting in greater efficiency, higher quality, and lower cost. Mindtree has defined this methodology based on best practices established through testing thousands of applications and systems in an environment of continuous process improvement.

The delivery team adapts MindTest™ and if required, customizes the frameworks based on the needs of the project. Throughout the lifecycle of project delivery, governance mechanisms are applied in the form of management reviews and metrics that drive these reviews.

Test process models

Defining test process models are important for the success of any test services organization. It helps in standardizing the process followed throughout the enterprise and thus creates predictability in the output / deliverable produced.

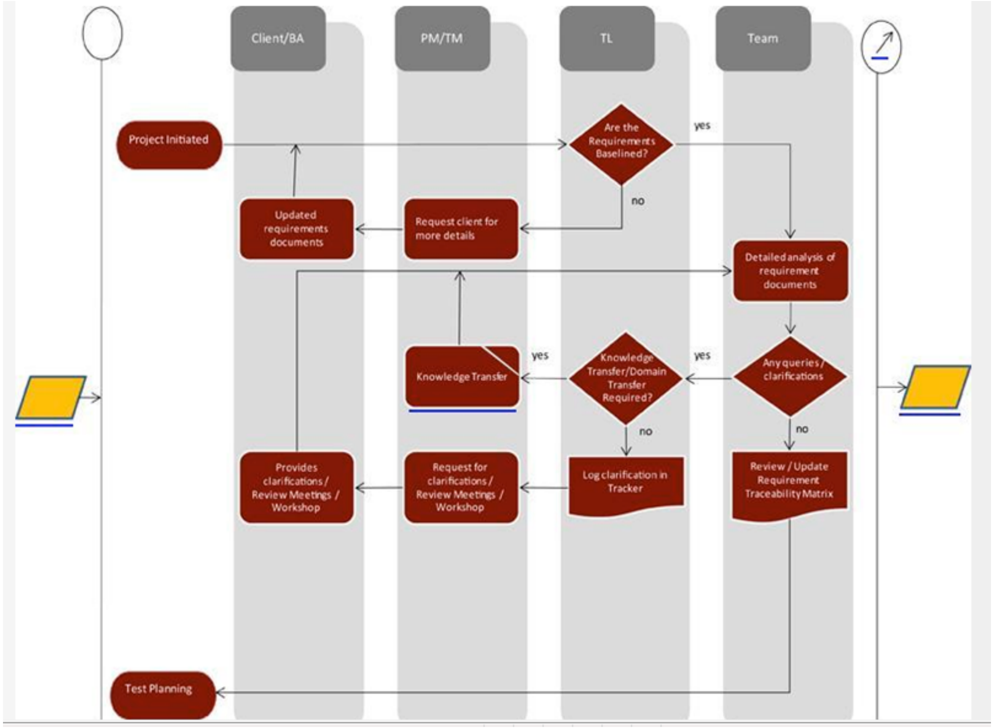
In MindTest™ we have defined the process that needs to be followed for most testing activities, a few of which are described here.

Software Testing Life Cycle processes (STLC)

MindTest™ Test Processes are defined as role-based swim lane diagrams for simplicity and clarity. These flow charts explain what is expected from all the different roles in the testing team during a specific testing phase. The phases of STLC have been explained right from the requirement analysis to test execution.

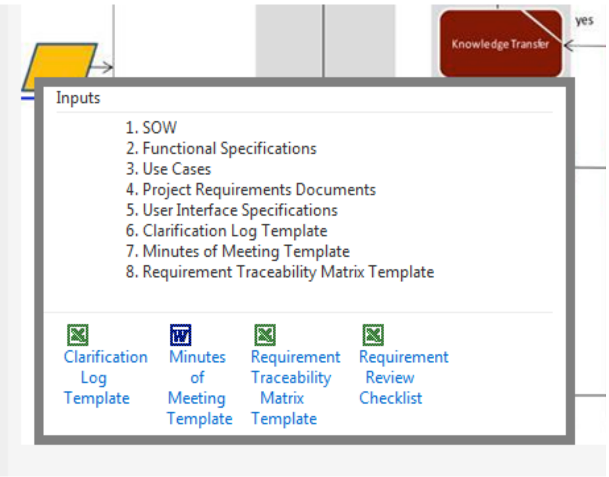
- The activities at each stage with the project team roles (test lead, project manager, etc.) have been explained
- Each stage has been clearly articulated with input metrics that can be collected, input documents required to begin the activities involved in that stage along with the output metrics and documents that come as output specific to that stage
- The templates and checklists relevant for each stage have been embedded with easy to download swim lane diagrams

Below is an example of the diagrammatic approach for requirement analysis:



Client/BA
PM/TM
TL
Team
Project Initiated
Are the Requirements Baselined?
yes
no
Updated requirements documents
Request client for more details
Detailed analysis of requirement documents
Knowledge Transfer
yes
Knowledge Transfer/Domain Transfer Required?
yes
Any queries/ clarifications
no
no
Provides clarifications / Review Meetings / Workshop
Request for clarifications / Review Meetings / Workshop
Log clarification in Tracker
Review / Update Requirement Traceability Matrix
Test Planning

The documents are embedded as shown below:



Inputs

1. SOW
2. Functional Specifications
3. Use Cases
4. Project Requirements Documents
5. User Interface Specifications
6. Clarification Log Template
7. Minutes of Meeting Template
8. Requirement Traceability Matrix Template

Clarification Log Template
Minutes of Meeting Template
Requirement Traceability Matrix Template
Requirement Review Checklist

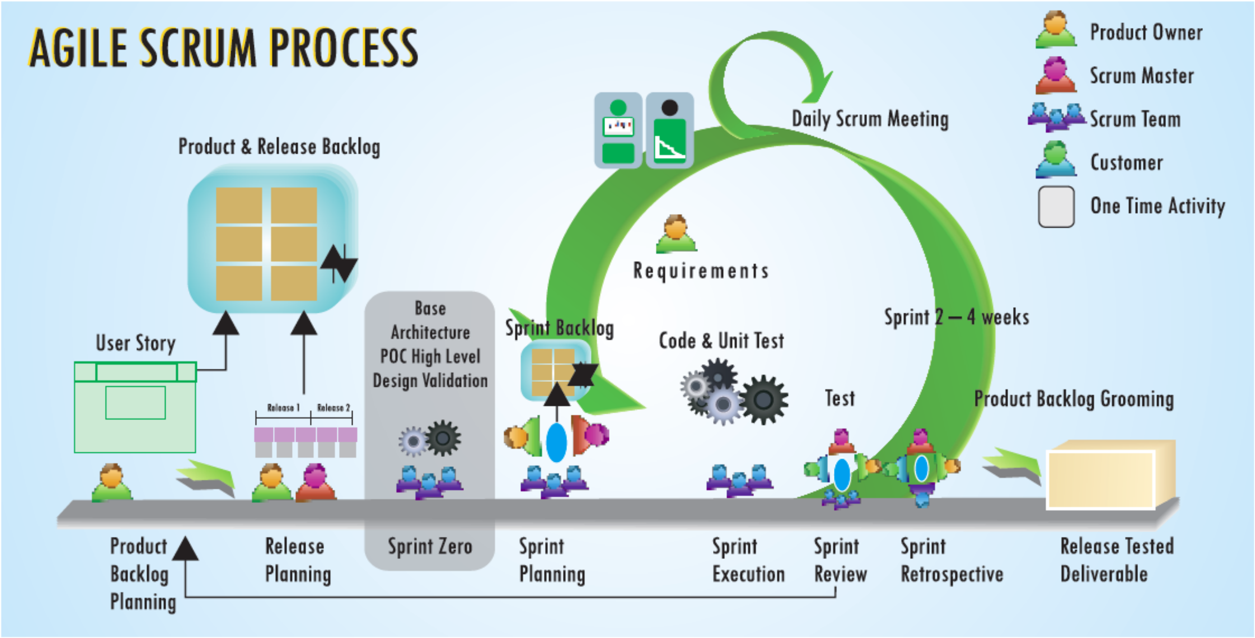
Agile testing methodology

Today, with most customers adopting agile, it is critical that enterprises standardize methodology that can be adopted a cross project teams.

In MindTest™ a generic agile iteration has been explained in detail.

- The testing objective of each phase is defined
- Testing activities expected at each stage are clearly articulated through Mindmap
- Quick tips are provided for activities which help the testing team
- Relevant templates and checklists are available for project teams to download

Agile guidelines: a glimpse



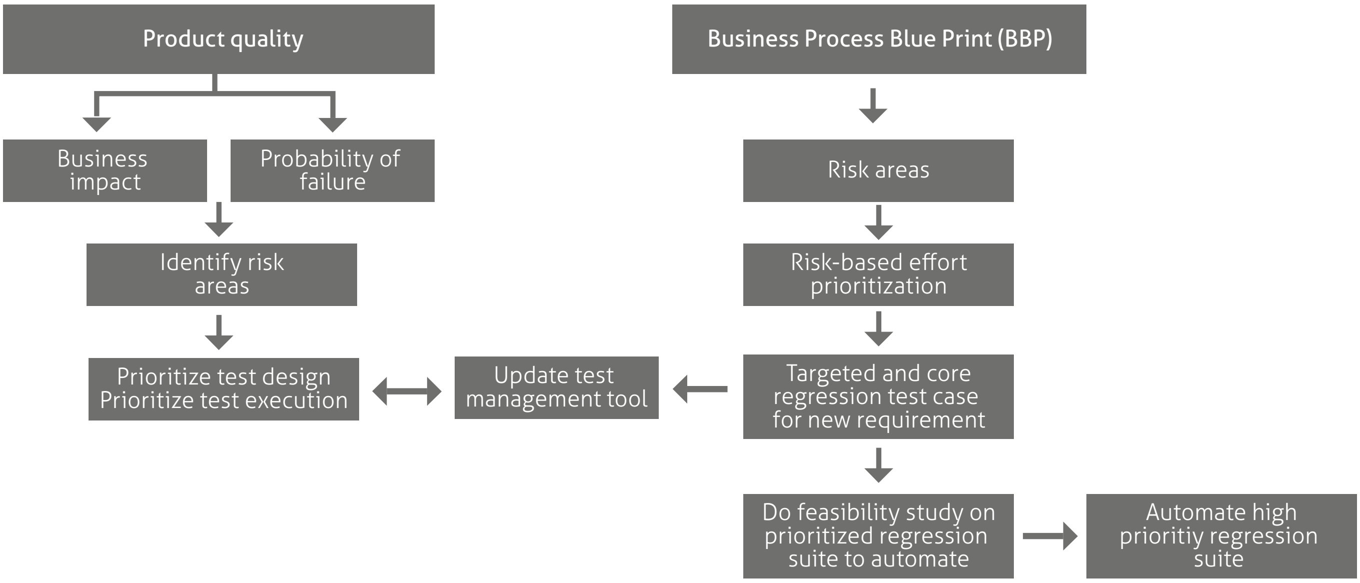
Release planning	Sprint planning	Sprint execution	Sprint review	Retrospective	Wait times
▪ Test planning ▪ Bring out clarity of user stories and scope of testing ▪ Define "Doneness" criteria ▪ Identify metrics to be monitored	▪ Expand the understanding of stories focusing on end user and non-functional stories ▪ Keep quality debt to zero	▪ Ensure "Doneness" criteria met ▪ Include both confirmatory and exploratory ▪ Unearth quality risk upfront ▪ Metrics collection	▪ Provide QA confidence score ▪ Bring-to-table and quality debt issue ▪ Discuss any defect backlog that requires immediate fix ▪ Test metrics analysis	▪ Identify testing related obstacles and how to overcome them ▪ Discuss and initiate any process improvements	▪ Proactively utilize time for next sprint ▪ KT and resource ramp-up ▪ Initiate process improvement, focus on one or two areas

Risk Based Testing (RBT)

The RBT approach to software testing prioritizes features and functions being tested based on their level of risk. It objectively determines how much to test by using a combination of business and technical requirements to focus and prioritize the testing

- Have articulated the approach to RBT for new requirements and regression testing
- Benefits
- Risk execution analysis, metrics and reporting
- Relevant templates and checklists available for download

The MindTest™ RBT approach can be implemented for both, testing new requirements and regression testing.


Product quality
Business impact
Probability of failure
Identify risk areas
Prioritize test design
Prioritize test execution
Update test management tool
Business Process Blue Print (BBP)
Risk areas
Risk-based effort prioritization
Targeted and core regression test case for new requirement
Do feasibility study on prioritized regression suite to automate
Automate high prioritiy regression suite

The business impact has to be provided, along with the requirements identified for the release by the business. BBP is the document where the core business process is mapped to the functional and further to the sub-functional area along with the business impact.

New requirement	Business process blue print	Regression test suite	Test execution
▪ Analysis of new requirement to implement RBT ▪ Set-up of guideline to implement RBT for new requirements	▪ Identifies the business risk ares at an enterprise level ▪ Provides the business impact to different risk areas	▪ Classification of test cases as per guidelines ▪ Identification and prioritization of regression suite as per RBT guidelines	▪ New requirement test execution as per RBT guidelines ▪ Regression suite test execution as per RBT guidelines

New requirements are prioritized based on their business impact and development complexity, which allows for effective test planning and effort distribution. For regression testing to be targeted and non-redundant, the identification of the regression suite should concentrate around the impacted functionalities. Thus, with this we can successfully build a robust regression suite which makes RBT implementation easier. The following is expected to be part of the regression suite update:

1. Addition of new test cases
2. Modification of existing test cases
3. Deletion of duplicate test cases from the existing set

Governance

In today's fast paced changing IT dimensions, governance is needed for every project to ensure quality deliverables to the customer. Governance mechanisms and continuous improvement initiatives are part of MindTest™. We govern our projects through different tools and mechanisms. Some of which are discussed below:

Test dashboard

Decision-makers depend on metrics to determine the level of quality in a software release. The degree of correctness of these all-important decisions is greatly dependent on objective, accurate and high quality data provided by test teams.

The Mindtree test dashboard is a tool developed in-house for capturing quantitative data on all projects where testing is performed. The test dashboard data is used as one of the inputs for project management / test management reviews by delivery managers. It is also used by test managers as a mechanism for seeking help and highlighting risks and issues for taking corrective actions.

Today we have an ensemble of 15 test metric on the test dashboard. Each metric is associated with a goal which drives the objective. The screen below provides sample metrics charts of the test dashboard.

Chart #15 - Test Development Productivity

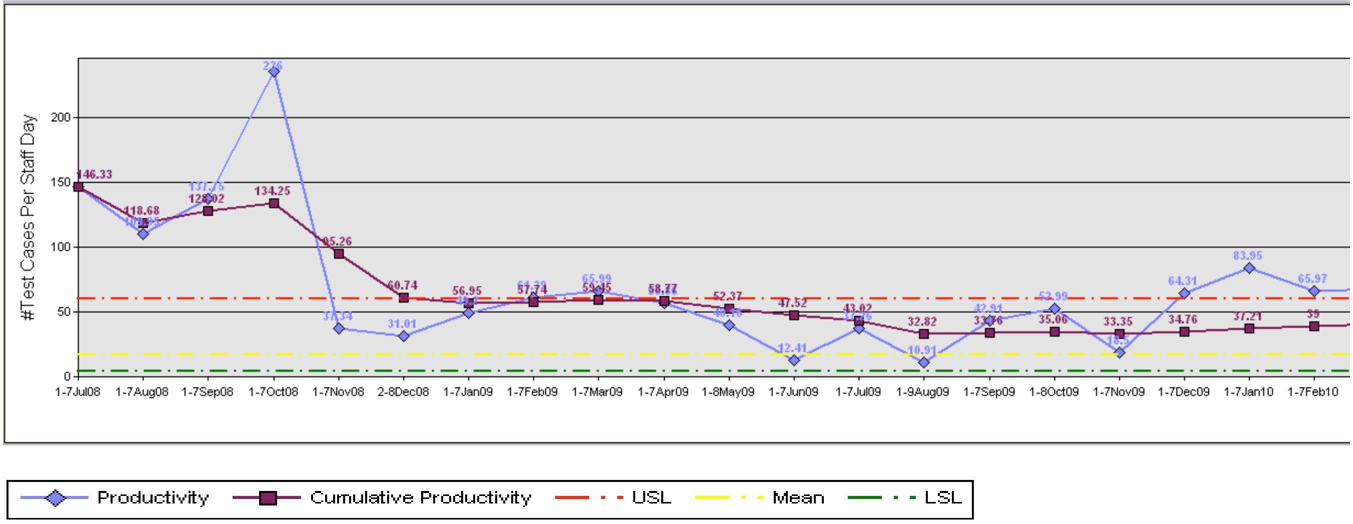


Chart #17 - Test Execution Productivity

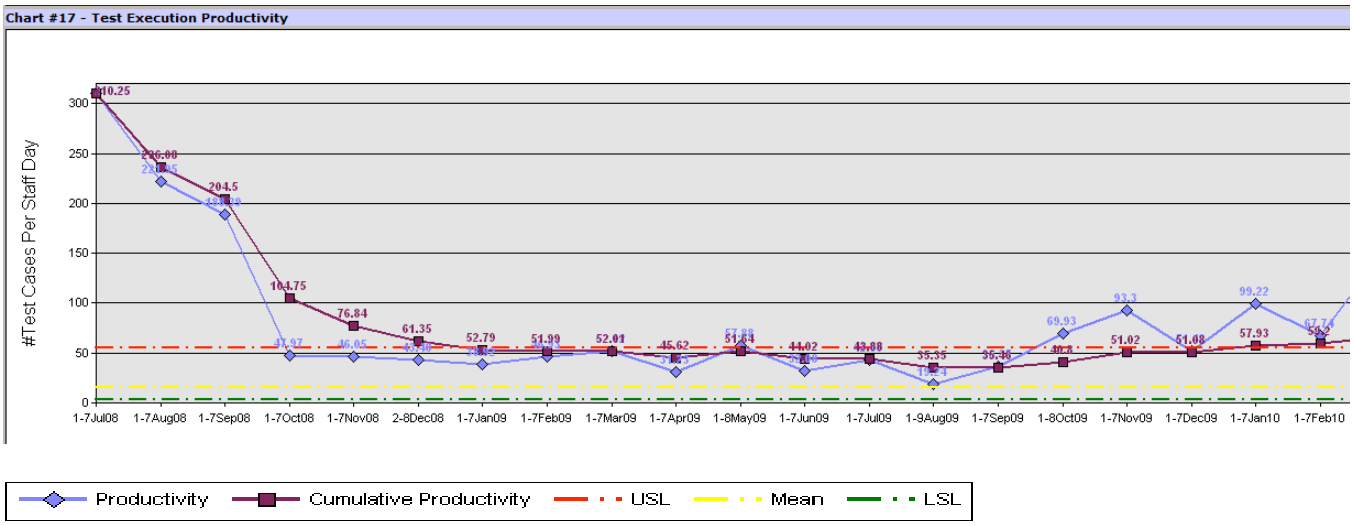
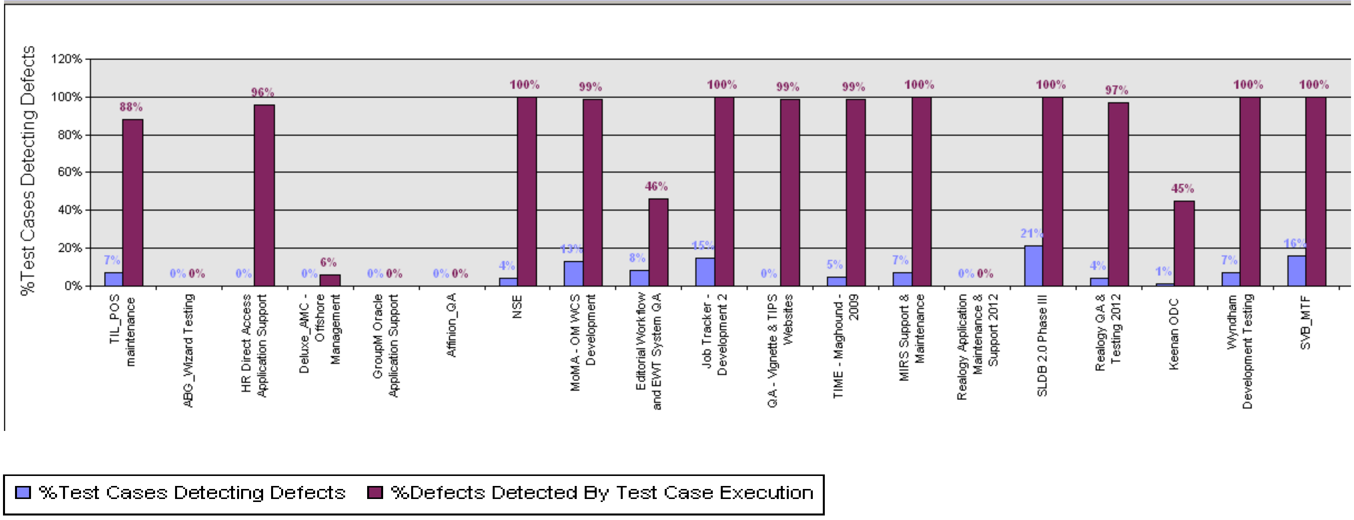


Chart #1 - Test Case Efficiency



The explanation for each metric is also provided for the user as shown in the sample below, pointer to the metric will provide the details.

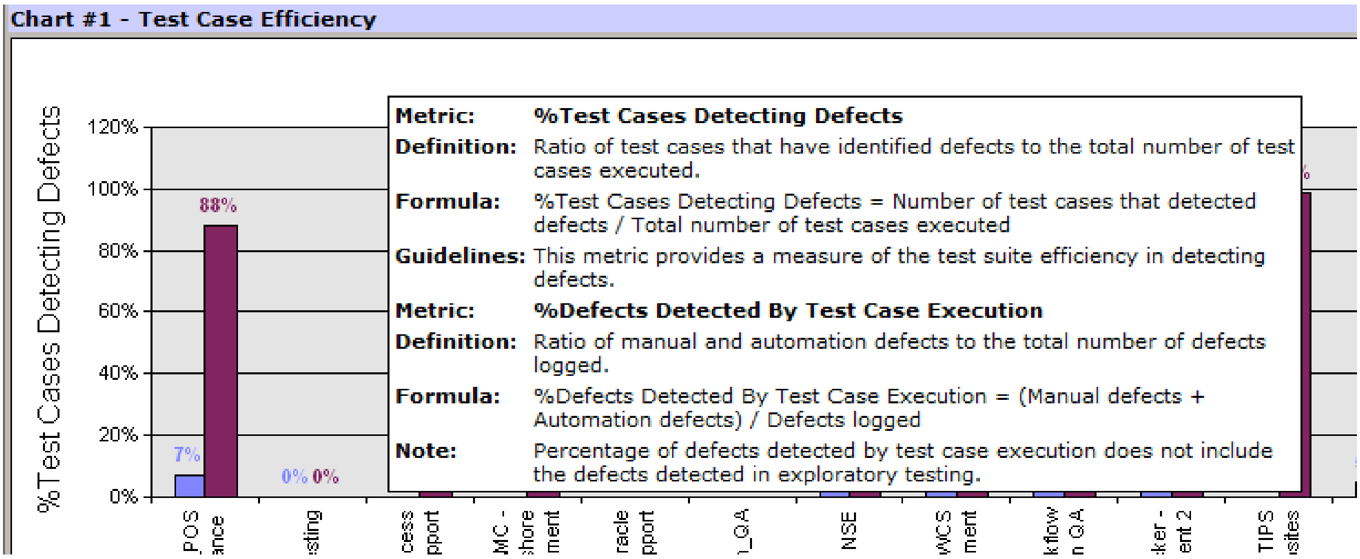


Chart #1 - Test Case Efficiency

Metric: **%Test Cases Detecting Defects**
Definition: Ratio of test cases that have identified defects to the total number of test cases executed.
Formula: %Test Cases Detecting Defects = Number of test cases that detected defects / Total number of test cases executed
Guidelines: This metric provides a measure of the test suite efficiency in detecting defects.
Metric: **%Defects Detected By Test Case Execution**
Definition: Ratio of manual and automation defects to the total number of defects logged.
Formula: %Defects Detected By Test Case Execution = (Manual defects + Automation defects) / Defects logged
Note: Percentage of defects detected by test case execution does not include the defects detected in exploratory testing.

TMAD model

Test Metrics Analysis and Decision (TMAD) model is a framework which enables project teams to choose the set of metrics aligned to the project's business objectives. This model helps teams define a metrics-based governance model. It also helps them in defining project specific metrics baselines and the corresponding corrective or preventive actions.

This model considers about twenty standard test metrics, some already present in the test dashboard. It then arrives at a common pattern and gives a flow to drive continuous improvement in test projects. The following is a sample set of attributes identified as patterns in the TMAD model, among others:

- Metrics goal patterns
- Associated patterns
- Undesired behavior patterns
- Decision support paths
- Corrective action paths
- Raw data requirements pattern

Since this model defines the formulae for arriving at metric results, the stakeholders can verify their data as and when in doubt about the authenticity of the graphs.

Benefits foreseen before defining this model:

- Aligning goals to the metrics associated patterns
- Ease of metrics analysis and decision making
- Introspection and retrospection on metrics data
- Prevention of recurring metric deviation
- Prevention of people biased decisions
- Identification and prevention of skewed data

Thus, with the help of a test dashboard and the TMAD model, we translate business goals into measurable quality metrics. A good mapping is established between the business goals agreed upon, objectives of the product, and quality metrics used to measure the excellence of the software. This linkage helps the test manager provide quality assessment of the product in a language that is better understood by all stakeholders.

INTEGRATED 360° VIEW

Product quality	Quality of testing	Test management	Test process	Test automation efficiency
▪ Defect trends, analysis, density ▪ Customer satisfaction ▪ DRE ▪ Release readiness ▪ Code coverage	▪ Proprietary Test Quality Index (PTQI) approach ▪ Test case efficiency ▪ Defect rejection ratio ▪ Review efficiency	▪ Schedule / effort variance ▪ Productivity ▪ Defect quality ▪ Maintainability ▪ Cost of testing	▪ Test coverage ▪ Exploratory testing % ▪ Automation testing % ▪ Issue and risks	▪ Savings in test time and effort ▪ Maintenance effort ▪ Test coverage ▪ ROI

While defining the metrics and processes, Mindtree clearly distinguishes between the processes and metrics designed to measure the quality of the software under test, and one which is designed to deliver efficient and effective testing.

As software products / applications and features become less differentiated, quality is likely to be the major differentiator. Therefore, access to this capability of software quality management of test efforts can become a strategic advantage for companies

Test Quality Index (TQI)

The TQI tool captures the subjective attributes of the quality of a testing service. TQI is a proprietary tool owned by Mindtree. It is a set of questionnaires with bivalent responses. The responses can be either a "Y" or "N", while "NA" is used as an exception. Generally the quality of testing is measured using metrics, like defect removal efficiency, test cases executed per hour, etc. However, these objective measures miss out on subjective attributes. For example, the quality of deployed resource, which includes domain knowledge, process knowledge, product knowledge, training, etc. These scores are projected to the respective leadership heads responsible for managing the delivery of the projects and services, and called Delivery Unit (DU) heads.

This tool is flexible and can be adopted across various projects as the metrics captured are generic, with an objective to measure the overall test quality. It gives you a comprehensive overall score for the entire project. The scoring is based on self-assessment by the project manager. It has different purposes to serve than the process compliance scores provided by maturity models like TMMI or CMMI; even though these scores are related in the bigger contexts.

TQI has the potential to impact and transform customer satisfaction and process maturity levels of software testing projects.

Test audits

Test audits are test technology audits where the objective is to identify the testing risks early in the life cycle of the project and help the team in creating contingency plans and mitigation plans.

These audits are carried out by the process evangelist team in collaboration with other test / project managers. The focus of the audit is to understand the project challenges. The audit team inspects all the documents available, interviews the audited team, and the process strives to uncover all the potential issues / risks. During the process, best practices from other projects are shared. Both the auditor team and the audited team are involved in identifying action items that need to be put in place to ensure the success of the project.

Account reviews

The objective of account reviews is for the test management team to have a detailed understanding of the different projects in each account and the types of testing performed in each of them. The test manager / test lead is expected to highlight the achievements, the issues, the training / certification requirements and various metrics they have been tracking for the project.

This discussion is a forum through which the test management team along with its three pillars of excellence (testing academy, test labs, MindTest™) identify the value additions that they could provide to the account to help them deliver better / faster / cheaper.

This is a face-to-face discussion where the business group head, the process evangelist team member, the operations team member, the people function team member and the testing academy team meets the rest of the project team.

Tools used as part of MindTest™

- At the organization level, we have defined a set of tools for different activities during the SDLC of a project. Below is the brief of the tools that will be used by the testing team.

Mindtree tools	Usage
MindTest™ portal	A repository that hosts the integrated processes, templates, checklists, guidelines and best practices to execute projects across all domains for testing.
Radar	Tracks, communicates, and analyzes defects in a software lifecycle, employing a review-based model that follows CMMI principles.
TestLink	TestLink is a web-based test management tool that is used to create, manage, and organize test cases into test plans. Helps in maintaining traceability.
Project space	Provides a persistent base for discus-sions, communications, documents and project specific knowledge.
Test estimate – estimation model for testing	Helps you arrive at the right test estimate for the testing effort.

Customer benefits

- Metrics to measure, track progress, assess maturity and improve predictability of the quality of testing and software
- Improved accuracy in decisions related to software quality assessment and release
- Measurable improvements in ROI with regards to time, cost, risk, and quality
- Greater visibility and control of software testing process and results
- Improved productivity and tester efficiencies through actual implantation of integrated processes and best practices implementation
- Enabling clients to establish a platform for continuous improvements in testing capabilities / processes

MindTest™ implementation for T&T customer

This customer is a leading global brand in the vehicle rental industry. It is a leading rental car supplier and primarily serves the premium commercial and leisure segments of the travel industry in the United States and other regions and as a premium brand in Canada, the Caribbean and other parts of the world.

Our customer provides a complete range of vehicle rental services through 10,000 rental locations in approximately 175 countries around the world. The company has three operating regions: North America, Europe, Middle East and Africa (EMEA) and Latin America / Asia Pacific.

Mindtree's engagement with this customer started in the year 2002. Mindtree is responsible for test planning, estimation, test case design and test execution, including functional testing and regression testing, end-to-end testing. Currently the Mindtree team is involved in multiple projects of the customer's testing portfolio for several testing segments:

- Web-based applications testing – manual and automation (Mindtree - SAF)
- Performance testing, security testing, mainframe testing, and database testing

Challenges

Business challenges:

- Communication for requirement and change requests in excel / mails started creating confusion resulting in schedule over-run created by late clarifications and rework
- Proper process implementation
- Difficulty in product quality measurement

Project challenges:

- Tracking requirement, defects and changing requests in excel / mails, and closing it became time consuming
- As defect repository was not available, team members had no clear visibility on status of defects
- Capturing quantitative data for different projects to track project health
- Identifying project risk was time consuming
- Capturing data like defect density, test execution productivity, effort over-run, review defect density was a cumbersome process

Proposed Mindtree solution

- Tool-based requirement / defects / change requests tracking (MindTest™ tool - radar)
- Adapting the MindTest™ governance mechanism to track project health and projects risk through test dashboard, test audits and account reviews

Implementation and project benefits

Radar usage implementation:

- **Requirement tracking**
  ABG uses the radar as a requirement repository. Requirements are logged in detail with the required attachments. These requirements are analyzed further and taken up as a tracker and brought to closure
- **Defect tracking**
  Unit testing defects: Code review and unit testing defects are logging in radar in the development phase
  QA Phase: QA defects are logged and assigned to respective developer, and brought to closure after the defect is fixed

Post implementation benefits:

- Easy-to-track, most affected module as defects get bucketed as per module
- As unit, code review and Quality Assurance (QA) defects are logged in the same application, it becomes easy to calculate DRE
- Easy-to-track defect ageing
- Email notification on any action helps the defect owner to take quick action, and reduces the turnaround time
- Requirement tracking in a common place helps to reduce rework
- Deferred or postponed, can be tracked over a period of time
- Requirement tracking in a common place helps to reduce rework
- Deferred or postponed, can be tracked over a period of time
- User-friendly advanced search, mass updates, and a structured reporting feature helps get a quick glance on the defect status / back logs, etc.

MindTest™ governance implementation:

- **Test dashboard**
  Started tracking complete testing metrics in several segments like – planned and actual, release information, schedule information, cumulative test execution data, and test design data. Analyze the comparative study and generating trend curves with phase / release-wise data and track the project health
- **Test audits**
  Well planned testing risks identification early in the life cycle of the project and help the team in creating contingency and mitigation plans
  Account audits
  Value additions identification that Mindtree can provide to the accounts / projects to help them in delivering better / faster / cheaper.

Post implementation benefits

- Easy-to-track metrics
- Quantitative approach helps measure project progress and decision making
- Tracking product health, risks in a planned way

Business benefit:

- Improved predictability in the quality of testing and the quality of product measuring metrics through the dashboard, track progress, assess maturity. Improved accuracy in decisions related to software quality assessment and release
- Measurable improvements in time, cost, risk, and quality
- Greater visibility and control of software testing process and results
- Enabling clients to establish a platform for continuous improvements in testing capabilities / processes

Conclusion

Integrated testing methodology is a must as the organization moves towards higher levels of test maturity. Various new processes and activities which are necessary for driving predictability in the quality of testing and hence the quality of software, are added. It is not enough to have a test process optimized just around finding defects before the final release, but it is essential to have an integrated approach to synchronize various aspects, such as:

- Ensuring that the test strategy and planning is consistent with the business objectives of the software being tested
- Domain, technology and tools aligned to requirement specifications and capability dimensions
- Setting-up processes and metrics that help track and measure the gap between existing quality and expected quality with an actionable plan
- Communicating the value and impact of testing on business goals through effective governance and tools