ENHANCING SOFTWARE QUALITY

INDIUM

RELIABLE RELEVANT METRICS TO THE RIGHT AUDIENCE

PREPARED BY : PREMALATHA

www.indiumsoft.com

TABLE OF CONTENT

1 Background

From the dawn of software we understood that one of the biggest challenges a software product faces is ensuring that its quality is sufficient enough, sufficient enough to allow the customer to use the product without any difficulties or setbacks and help meet his business needs. But what exactly is quality? Are bugs the only aspect of quality? Should the zero defect goal be the mark dictating whether our job is done? The answer to these questions is NO! While unexpected behaviour, manifesting itself in software bugs, plays a big role in product quality, it is not the only factor at play.

Analytics and quantitative measurement is gaining a lot of credence in the realm of quality management specifically around software testing.

“What cannot be measured cannot be managed” is the guiding philosophy behind testing metrics, a phenomenon that promises to deliver business efficiencies beyond just improving quality. Measurement helps with planning, tracking and managing the software project and enables organizations to objectively assess quality.

2 Software Metrics

Metrics derive information from raw data with a view to help in decision making.

Metrics, encompassing analysis of the process, product and the project, provide both quality managers and the management with dashboards to evaluate overall program health. An extensive collation of historical testing data in a defined framework, metrics provide intelligent insights for objective measurement of quality apart from improving efficiencies in estimation and scheduling.

Testing metrics are not a one-time compilation but an exercise that is dynamic and warrants constant validation, what with the rules of the game changing with every application and project.

The most important metric is value and few projects measure it. They tend to measure only cost and schedule. You should observe value and make sure that your process changes increase long-term overall value. Any other metric is just local optimization unless you have a theory or a preponderance of data to support otherwise.

Some of the areas that such information would shed light on are:

1. Relationship between the data points
2. Any cause and effect correlation between the observed data points and
3. Any pointers to how the data can be used for future planning and continuous improvements

Collecting and analyzing metrics involves effort and several steps.

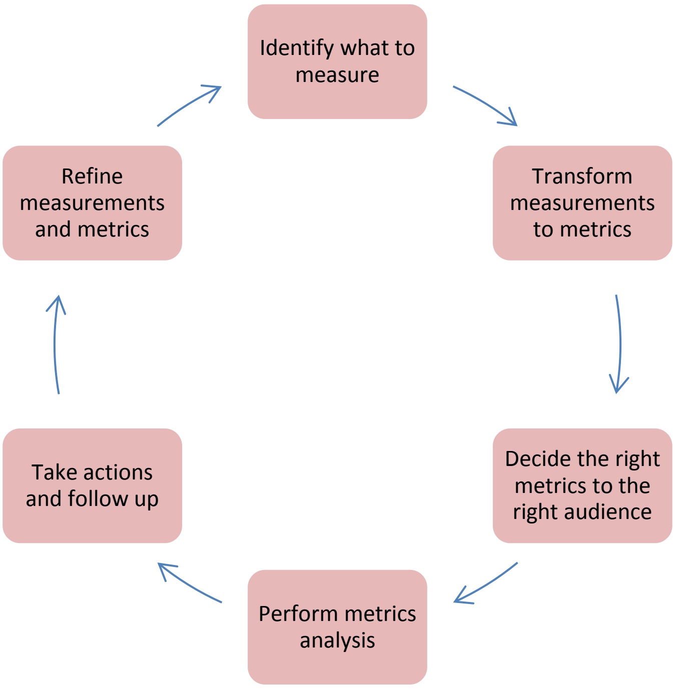
Identify what to measure
Transform measurements to metrics
Decide the right metrics to the right audience
Perform metrics analysis
Take actions and follow up
Refine measurements and metrics

Figure (i) - Metrics Life Cycle

The first step involved in a metrics program is to decide what measurements are important and collect data accordingly. The effort spent on testing, number of defects and number of test cases are some examples of measurements. Depending on what the data is used for the granularity of measurement will vary.

While deciding what to measure, the following aspects need to be kept in mind.

1. What is measured should be of relevance to what we are trying to achieve. For testing functions, we would obviously be interested in the effort spent on testing, number of defects reported from test cases and so on.

2. The entities measured should be natural and should not involve too many overheads for measurements. If there are too many overheads in making the measurements or if the measurements do not follow naturally from the actual work being done, then the people who supply the data may resist giving the measurement data (or even give wrong data).

3. What is measured should be at the right of granularity to satisfy the objective for which the measurement is being made. An approach involved in getting the granular details is called data drilling.

Regardless of the measurement scale, when the data are gathered we need to analyze them to extract meaningful information of data, more importantly what needs to be measured and to whom it adds value.

As the testing function/community matures, the test function needs to be aligned to the business and demonstrate contribution of the overall business. This implies assessing activities that go beyond release quality and process, like those related to profitability, business scalability etc.

3 Purpose

Metrics provide the following three basic functions:

- Control: To evaluate and control the performance of the resources.
- Communication: Metrics communicate performance not only to internal workers and managers for purposes of control, but to external stakeholders for other purposes as well. A well designed and communicated metrics provide the user a sense of knowing what needs to be done without necessarily requiring him/her to understand the intricacies of related processes.
- Analyze: Metrics collected are correlated and analyzed, to provide necessary information to the stakeholders.
- Improvement: Metrics identify gaps (between performance and expectation) that ideally point the way for intervention and improvement. The size of then gap and the direction of the gap (positive or negative) provide information and feedback that can be used to identify productive process adjustments or other actions**.**

4 Why classify metrics?

Let us look at the broader picture of the metric, what activities or phases one is interested in, what is particularly measured, depends on lot of other things, the type of project, complexity, availability of resources, the situation, customer requirement etc.

Metrics can be a good thing or a bad thing depending how well they align with delivering value to stakeholders. Let’s face it: building baselines, measuring results and continually raising the performance bar is not necessarily fun. It is time-consuming and arduous. The wrong metrics distract human capital from what is important. Entire organizations can be misled and substantial damage to the financial well-being of the company can result.

The value of the metrics gathered makes sense only if it serves the intended audience. For example, the cycle time, return in ROI will interest the management for making decisions as like the defect arrival rate, test efficiency; test effort inspires the test manager in making the release decisions.

Rightly said, the objective of this paper is to project the fact that relevant metrics to the relevant people makes metrics a real value addition than just collating and presenting the numbers.

Let us classify the stakeholders of the project as Sponsors, Delivery Organization and Continuous Process Improvement.

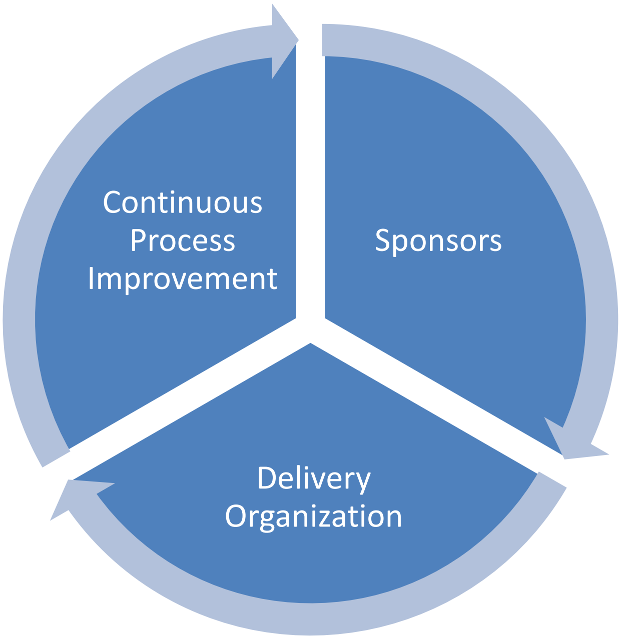

Figure (ii) - Stakeholders Categorization

5 Sponsors

Project sponsor is a person or an organization that is in charge of driving the project towards directions that will bring the project to successful realization of expected benefits. Project sponsor finances project initiatives and works, takes care of engagement processes, facilitates development of the initial scope and the project charter and participates in processes of project communications management.

The role of project sponsor is critical to ensuring the success of projects, especially IT projects. Successful initiation of a new project is always based on project sponsorship. If the project is well funded by project sponsors, it has the required financial foundation to carry out works, achieve objectives and produce deliverables.

A project sponsor should understand real ROI, plan market strategies, plan product releases and estimate new projects. The following metrics will interest the project sponsors based on the above said traits:

- Customer Satisfaction
- ROI

Metrics like Cycle time, Turnaround time indirectly helps in customer satisfaction, product and service quality and thereby profitability. Let us have a closer look at these relevant metrics for sponsors.

5.1 Customer Satisfaction Index

Customer Feedbacks can help take your business to the next level. Customer satisfaction is defined as "the number of customers, or percentage of total customers, whose reported experience with a firm, its products, or its services (ratings) exceeds specified satisfaction goals." In a survey of nearly 200 senior marketing managers, 71 percent responded that they found a customer satisfaction metric very useful in managing and monitoring their businesses.

It is seen as a key performance indicator within business and is often part of a Balanced Scorecard. In a competitive marketplace where businesses compete for customers, customer satisfaction is seen as a key differentiator and increasingly has become a key element of business strategy.

The usual measures of customer satisfaction involve a survey with a set of statement. The customer is asked to evaluate each statement and in term of their perception and expectation of performance of the organization being measured. Their satisfaction is generally measured on a five-point scale. The questionnaire prepared to get feedback on the following aspects:

- Quality of the service
- On time delivery
- Business requirement understanding
- Communication

Below picture gives the customer satisfaction index for a completed project. The survey is conducted in a five point scale and based on the calculation the percentages are obtained. The green line is the goal.

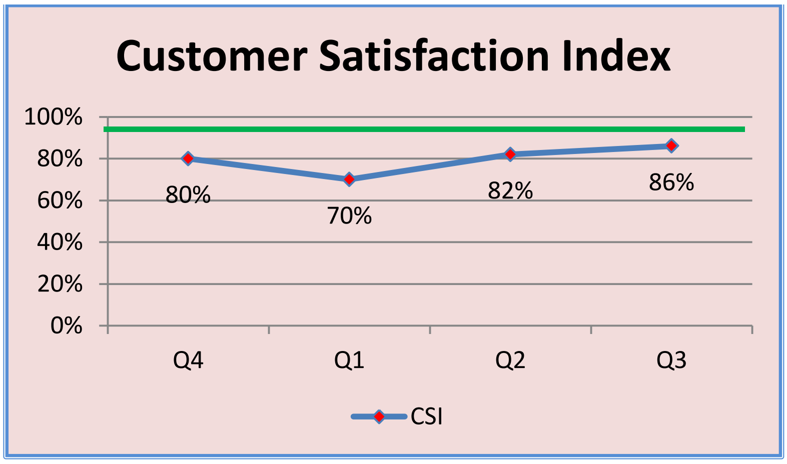

Figure (iii) - Customer Satisfaction Index

5.2 ROI

The very generic notion about testing is that it does not give any Return on the Investment made and all the investment done is to ensure the quality of the product to be delivered. So, the question is – Is there any ROI (Return on Investment) of software testing? Calculating ROI for testing engagement also become important as testing spending forecast is going to be increased as per the NelsonHall report.

Return on Investment is the ratio of money gained or lost on an investment relative to the amount of money invested. ROI shows the cost performance and the gain percentage. The amount of money gained or lost may be referred to as interest, profit/loss, gain/loss, or net income/loss. The money invested may be referred to as the asset, capital, principal or the cost basis of the investment.

ROI % = ((total earnings - total costs) / total costs) x 100

The outcome of the Cost Benefit Analysis would be identifying the improvement actions and roadmap to complete the same.

Given below is a case study on a test automation project.

Tool cost	$20,000
Training	$5,000
Machine	$3,000

Months	**ATDT**	**ATAT**	**ATMT**	**Investment**	**Gain**	**ROI**
	$	$	$	$	$	%
6	30000	5760	11520	75280	122400	62.59
12	30000	7200	14400	79600	153000	92.21
18	30000	9600	19200	86800	204000	135.02

Table 1: Return on Investment calculation

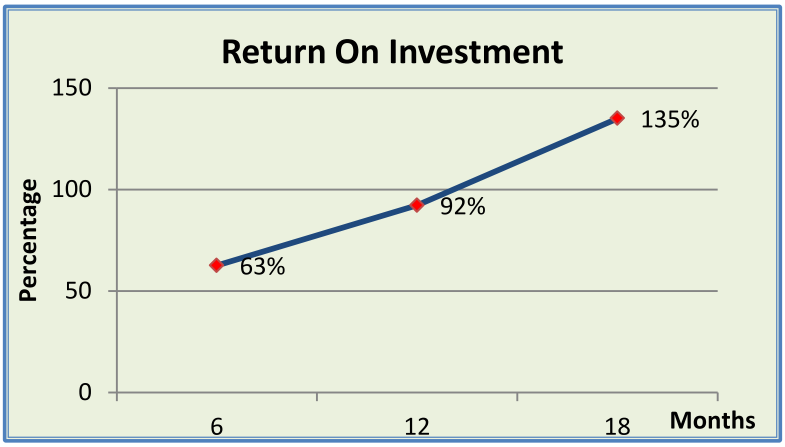

Figure (iv) - Return on Investment

6 Delivery Organization

The delivery team comprises the rest of the Project Management Organization and the Project Delivery Organization. Delivery Organization needs focus on the quality deliverables, on schedule and within the budget given by the stake holders. To perform these set of activities, the Delivery Organization needs the right set of metrics that would result in effectual work.

Today, an Organization's ability is to compete, prosper - even survive - depends on its skill to understand, leverage and integrate business and IT skills, processes and capabilities in the service of corporate strategies and operations. So, as an IT manager, we need to collect, monitor and act on information regarding the use and performance of IT assets and services and communicate that right set of information to the pertinent executives and managers across the organization.

Inline to this, there are several metrics which can be framed depending on the set of audiences. But on a wide range some of the common metrics that are used by the Delivery Organization across testing industry would be

- Defect Leakage
- Cycle Time

The list would go higher and higher depending on the demand of quality the relevant organization needs. One of primarily looked upon metrics by a testing organization on the delivery angle would be Defect Leakage.

"Customer is King" is a well said quote.

But, how is this related with Defect Leakage? Whenever there comes a conversation about customer satisfaction, the first phrase that would pop up in our mind, would be problems that occur at customers end. Later we think on how to help in reducing that number. One of the questions that come up in these situations is whether we're finding new things at customer sites or whether we're simply running into issues that we've already fixed.

The best way to figure that out is to look at our defect leakage rate.

6.1 Defect Leakage

Defect Leakage is the number of bugs that are found in the field that were not found internally. There are a few ways to express this:

- total number of leaked defects (a simple count)
- defects per customer: number of leaked defects divided by number of customers running that release
- % found in the field: number of leaked defects divided by number of total defects found in that release

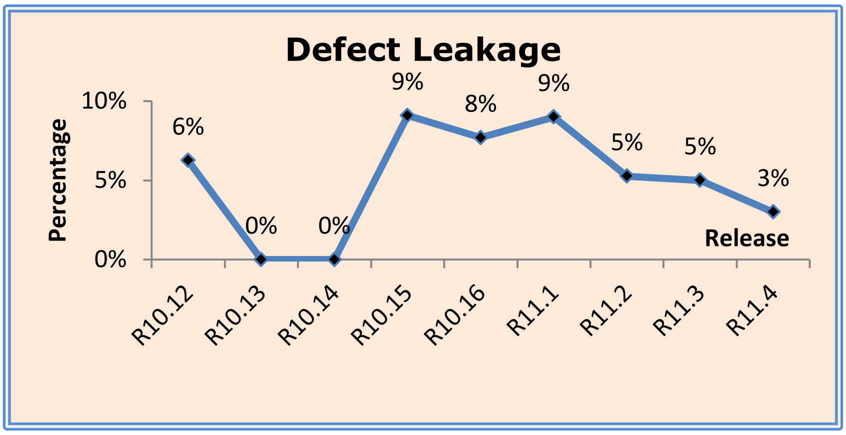

Figure (v) - Defect Leakage

The above given chart depicts the defects that were revealed by the client end. Now, the team needs to work why these defects were not checked out in the previous release or has this shown up due to certain patch fix that happened during the release. Similarly, in theory, this can be measured at any stage - number of defects leaked from dev into QA, number leaked from QA into beta certification, etc.

6.2 Cycle Time

Fast cycle time is the on-going ability to identify, satisfy and be paid for meeting customer needs faster than anyone else. A good implication is that the time devoted to cycle time analysis and improvement has to be considered work just as much as the time spent designing the product itself. By using cycle time as a central measure, highlight deficiencies that may be hidden between functions in a traditional organization.

Using cycle time measures also provides a quick and painless way to identify and measure defects. It is not compressing today's activities into a shorter time frame. But the only way to increase product quality and reduce cost while concurrently improving product development speed is to fundamentally change the development process itself.

In the following case study, the number of executions for each work package request, for each month is given. And the customer's request delivery timeline is 10 days. With proper planning the cycle time is reduced and maintained at 7 days.

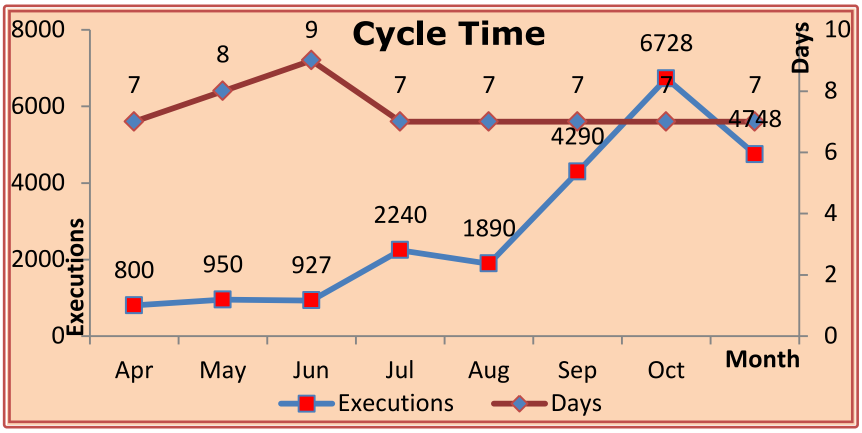

Figure (vi) - Cycle Time

Due to the reduced cycle time, the number of request of work packages from the customer has ramped up, thereby increasing the profitability.

6.3 Sprint Burndown chart

Next kind of metric that would impress any agile working team would be a Sprint Burn down. This Chart displays the remaining work in the sprint backlog. It's updated every day and it gives a simple view of the sprint progress. It also provides quick visualizations for reference.

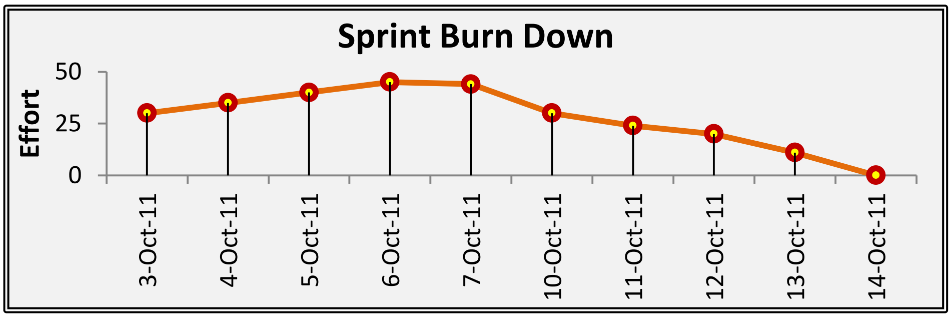

Figure (vii) - Sprint Burndown

Sprint Burndown chart is a place to see daily progress on the team's completion of work on an agile platform. In the above given chart we can check the progress of work completed by the team on each day of the Sprint cycle. Now, why do we need a Burndown chart? We use a Burndown because it

provides transparency into how we are doing over a given time frame. It allows quick answers to questions like, "Are we on track, are we ahead, or are we way behind?"
The release/Sprint Burndown provides exactly this piece of information towards a release goal. The inspection opportunity the release/Sprint Burndown allows provides the precious ability to adapt immediately when changes are realized. It gives the Scrum Team the chance to ask the hard questions early, instead of waiting until the week before the planned release date.

In general, the Delivery Organization needs to set few of these principles in mind before creating any metric to improve the organizational value. Metrics created for the Delivery Organization should emphasize on the:

- Visibility to help in aligning the investments with business priorities and requirements
- Ability to increase business process in manual/automation to drive effectiveness
- Capability to recognize the risks and potential for various levels and points of failure
- Facility to create a management discipline to control these risks
- Availability of real-time business impact information to enable you to apply a business focus to the delivery of IT services and enhance the business perception of the IT function.

7 Continuous Process Improvement

There are set of metrics that should be checked on periodic basis to improvise the overall quality of testing. These values needs to be figured based on the need for the set of people. Failing to meet expectations may be the result of poor performance, inadequate processes or controls, unachievable expectations or changes in expectations because of new business requirements.

A well-run IT Service Delivery Organization must constantly monitor performance and adjust to changing requirements or business priorities in order to achieve the appropriate balance between control and agility required to for long-term effectiveness. To obey this principle we need to have a check on few of these metrics on a periodic basis.

Some of the metrics that needs to check on a regular basis would be:

- Schedule variance
- Defect catch rate

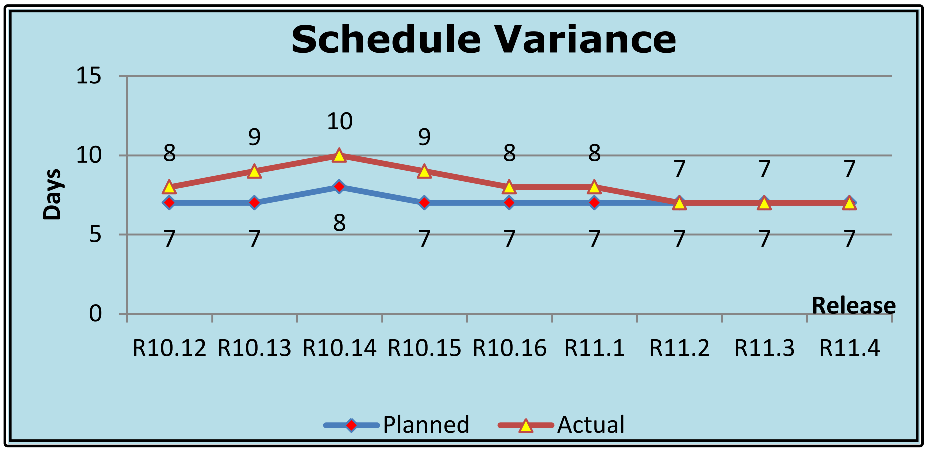

Figure (viii) - Schedule Variance

7.1 Schedule Variance

Schedule Variance % indicates how much ahead or behind schedule the project is in terms of percentage.

Schedule Variance % can be calculated as using the following formula:

SV % = Schedule Variance (SV) / Planned Value (PV)

Schedule variance highlights the following:

- Delayed start
- Application down time
- Error in planning test phase
- Logging more bugs

Some of the factors that need to be addressed before considering the importance of schedule would be:

- Reduce essential and accidental complexity
- *S*upport estimation
- Optimize processes
- Streamline software operations

7.2 Defect Catch Rate

Tester would love to know how many defects remain undetected when they deliver software to a customer or user. For capacity planning and in service management, knowing in advance how many people will be needed for application support would be welcome. Intuitively it seems impossible to predict the unknown future and how many defects customers and users are able to detect – and consequently have to remove – when starting to use the new application. However, statistical methods exist for predicting the probability of finding defects by calculating the expected defect density in requirements.

Defect catch rate is the percentage of defects captured with the test cases executed. It gives the efficiency of the test cases developed. Below is the defect catch rate for various releases.

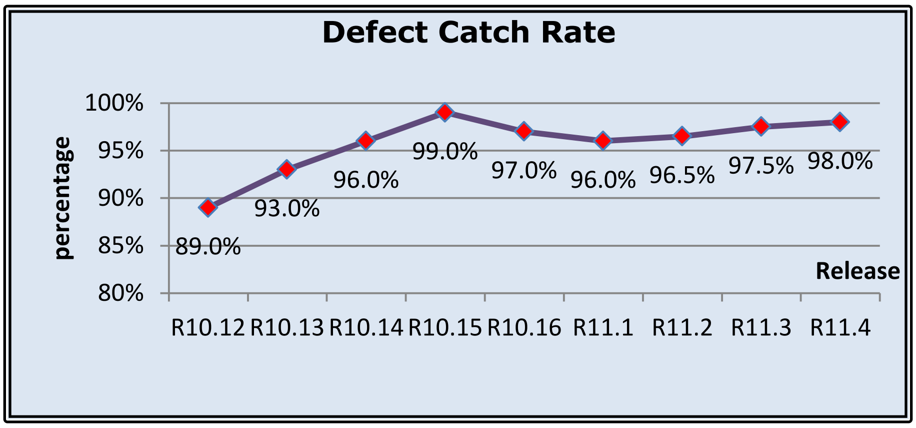

Figure (ix) - Defect Catch Rate

Before benchmarking the Continuous process improvement metrics, we need to think over on the need for generating the process improvement metrics. Some of the below pointed factors can be considered:

- The efficiency with which defects have been identified in different testing stages
- The overall confidence level in the testing efforts

8 Conclusion

The information gathered must have a specific purpose (or purposes). For example, the information will be used to determine the number of defects and time used for each testing phase, in order to determine the most cost effective ways to minimize errors.

The information to collect must be understandable and viewed as relevant to the collector or the information will not be collected. For example, to make the information in the previous example relevant, explain that the findings will highlight the testing methods that work and methods that don't work, so that employee effort is focused on productive activities.

Today it is fairly commonplace to collect metrics related to defects, coverage, risk, estimation, variance, test cycles, density and test cases that are sharply focused on evaluating the quality of software to help in making good release decisions. In addition to these some metrics like escapes, productivity, ROI allow us to evaluate aspects related to the test process and also some initial measure of business value.

Sponsors	Delivery Organization	Continuous Process Improvement
• Customer Satisfaction Index • ROI	• Defect Leakage • Cycle Time	• Schedule Variance • Defect Catch Rate

Rightly said, the objective of this paper to project the fact that relevant metrics to the relevant people makes metrics a real value addition than just collating and presenting the numbers.

- Analytics and quantitative measurement gains credence in the realm of quality management specifically around software testing
- The value of the metrics gathered makes sense only if it serves the intended audience
- Classification of metrics relevant to the Sponsors, Delivery Organization and Continuous Process Improvement team
- An extensive collation of historical testing data in a defined framework

About Indium:

Indium Software is exclusively focused independent software testing services firm since 1999. Over the years, Indium mastered objective methods that minimize the risk of failure of applications and software products. With a global headcount of over 300 employees, Indium works for a mix of marquee Enterprise and ISV clients spread across the globe. Indium is aggressively pursuing the Social, Mobile & Cloud agenda to make these the core of our next wave of service specialization.

Contact us:

USA - Indium Software Inc.

Corporate Office
California
Suite 210, 1250 Oakmead Parkway
Sunnyvale, CA - 94085.
Phone: +1(408) 501-8844
Fax: +1(408) 501-8808

Atlanta
Crown Office Suites,
1870 The Exchange, Suite 100,
Atlanta, Georgia 30339. USA
Phone: +1(770) 989-7302,
+1(678) 608-0809

Sales Inquiries
americas.sales@indiumsoft.com
apac.sales@indiumsoft.com
emea.sales@indiumsoft.com
india.sales@indiumsoft.com
sales@indiumsoft.com

INDIA - Indium Software (India) Limited

Global Delivery Headquarters
Chennai
VDS House
41, 2nd Floor
Cathedral Road
Chennai - 600 086.
Phone: +91-44-2811 6330
Fax: +91-44-4210 4033

Prince Arcade Building
22-A, 3rd floor
Cathedral Road
Chennai – 600 086.
Phone: +91-44-4347 7200

Steeple Reach,
Old No. 25, New No. 39,
Cathedral Road, Chennai –
600 086.
Phone: +91-44-4347 7122

General Enquiries
careers@indiumsoft.com
info@indiumsoft.com

Bengaluru
Salarpuria Adonis, 2nd Floor
Binnamangala, 3/1, Old Madras Road
Kadiranapalya, Indira Nagar
Bengaluru – 560 038.
Phone: +91-80-6784 7500

New Delhi
F-1/5 Okhla Industrial Area,
Phase-I,
New Delhi – 110 020.
Phone:+91-11-6613 0400