



Technical White Paper

**Productivity Improvement and
Measurement Mechanism
in Maintenance Projects**

**Project Management Leadership
Conference 2007
Organized by
QAI**

C. Padma

Head, Application Services
Business

STATEMENT OF CONFIDENTIALITY

Information included in this document, in its entirety, is considered both confidential and proprietary to Sonata Software and may not be copied or disclosed to any other party without its prior written consent.

Abstract

This paper describes the measurement process that is capable of predicting costs and schedules, used for Maintenance Projects. It also provides an overview of measurement of Comparative Productivity, which in turn helps in analyzing Cost-Benefits and calculating the ROI for an outsourced operation.

About the Author

Padma heads one of the four development groups at Sonata.

With experience in handling large project-management, multiple accounts-management and customer interfacing, Padma brings to the team an extensive expertise.

Apart of being in-charge of project delivery in her group, she also participates in other activities like pre-sales, organization-wide quality initiatives and resource management.

Table of Contents

- 1. **Abstract1**
- 2. **Typical Maintenance Process1**
- 3. **Lean Manufacturing Principle2**
- 4. **ROI Measurement Approach4**
- 5. **Case Study – Maintenance Project6**
- 6. **Application of the Principle of Lean Manufacturing6**
- 7. **Conclusion13**

1. Abstract

It is a well-established fact that 50% to 60% of software outsourcing is done in the area of Maintenance Processes. It is also true that a major chunk of IT budget is allocated to maintain the existing applications to avoid breakdown of operations. This simultaneously proves the importance of Maintenance Operations and highlights the challenge of keeping it as low-cost as possible. This is a continuous challenge to all companies and Project Managers.

The work-pattern analysis of a typical Maintenance Project reveals that it comprises of either Platform Migration or Maintenance Processes, or both. However, the issue is that the knowledge of an application is extremely critical for its maintenance; even though Knowledge Transfer (KT) is not an easy job, especially for applications that have evolved over decades and have multiple customizations.

The objective of this white paper is to present a structured method for an effective KT of an application to remote teams. This is being referred to as the 'Lean Manufacturing Approach', as the developer is not burdened with too many CRs at any point of time.

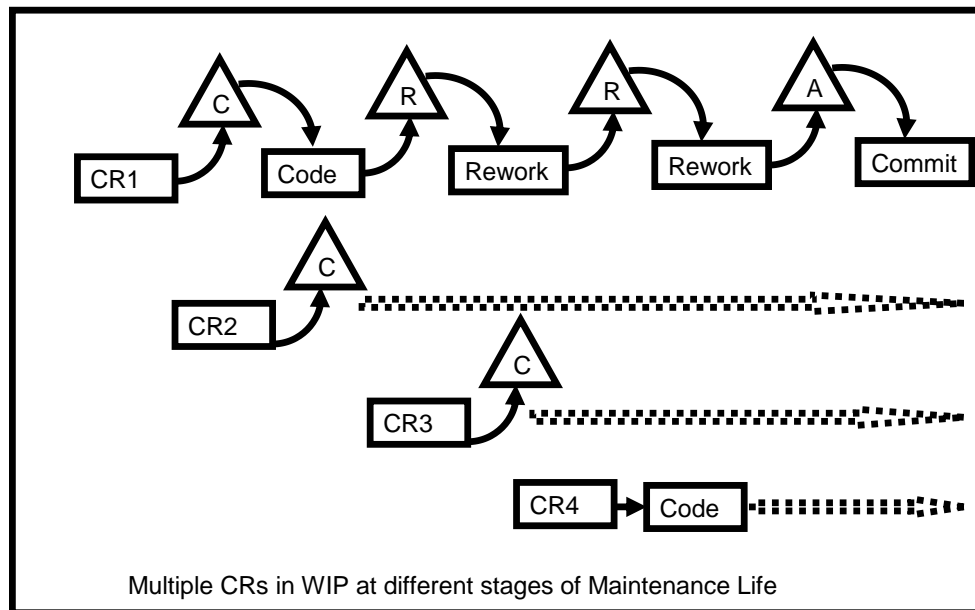
Maintenance of Projects involves the management of highly dynamic tasks as the list of bugs to be tracked and eliminated keeps changing on the basis of the requirements of its end-users. Hence, the measurement process for Maintenance Projects has to be different from that for project-based assignments, which have definitive estimates, tasks and milestones to be achieved.

This paper describes the measurement process, capable of predicting costs and schedules, used for Maintenance Projects. It also gives an overview of the measurement of Comparative Productivity, which in turn helps in the Cost-Benefit Analysis and ROI calculations of an outsourced operation.

2. Typical Maintenance Process

The following diagram depicts the flow of a typical Maintenance Process.

Change Request (CR) for maintenance is assigned to the developer for fixing. The developer analyses it and performs the Detail Design. While doing this, he might have some questions, which need to be raised and allayed.



'C' in triangle indicates the Clarification stage.

'R' in triangle indicates the Review stage.

'A' in triangle indicates the Approval stage.

As the developer waits for Clarification for CR1, he may pick up CR2, the next CR, for Analysis. During the Analysis of CR2, he may receive the Clarification for CR1. Then, he completes the Coding for CR1 and puts it for Review.

Similarly, while waiting for Clarification for CR2, he may pick up CR3 for Analysis.

This way, at any given time, there are multiple CRs in different stages, due to which it gets too complex for the developer to switch between them and for the Project Manager to track their status.

This affects the throughput adversely and delays the Release Cycle.

3. Lean Manufacturing Principle

Individual allocation of CRs for Clarification gives rise to enormous wait-times and also reduces throughput. Taking a new CR for Analysis during each wait-time without any plan also results in chaos because of too many CRs in the WIP. It also adds to the delay in terms of the time required to switch between CRs.

Typical 'time-traps' are observed in the following areas:

- Analysis of a defect is highly dependent on application functionality at a detailed level. But application documentation or Detail Design is generally not done at that level of detail. Hence the developer has many questions and consequently, there is a time-lag for Clarifications.

- Due to lack of complete knowledge of application functionality and Detail Design which is done in a hurry, many reworks come up after delivery. Multiple rounds of rework create confusion, as much time is wasted in analyzing the impact of each change on the application.
- Generally, there is a lag between Coding and Review of a CR. Review is done just before Code Freeze, expecting nominal rework. However, sometimes a lot of rework is required, which pushes the Release Date further.
- A suitable batch size of CRs can help reduce these delays.

Principle of Lean Manufacturing:

The activities that cause critical-to-quality issues from customers and the longest delays in any process offer the greatest opportunity for improvement in cost, quality and cycle time.

Hence, the key improvement areas are:

- **Optimized batch size & Time-Boxing:** Select a batch size of CRs in such a way that there is no wait-time for any individual; at the same time, not too many CRs in the WIP. Decide the optimal queue size. For example, a cycle time of one week and a batch size of four CRs per individual for a Time-Box of three weeks is ideal.
- **Functionality-based allocation:** In order to reduce the number of interaction cycles during Analysis and rework cycles after delivery, CRs are segregated on the basis of features/modules and all the related CRs are given to one individual.
- **Analysis of all CRs in the first Time-Box:** The developer studies all the CRs belonging to a particular functional area and comes up with Detail Design and estimation for them, which are reviewed by the customer representative. Then, the customer representative gives his feedback, according to which changes are incorporated in the CRs.

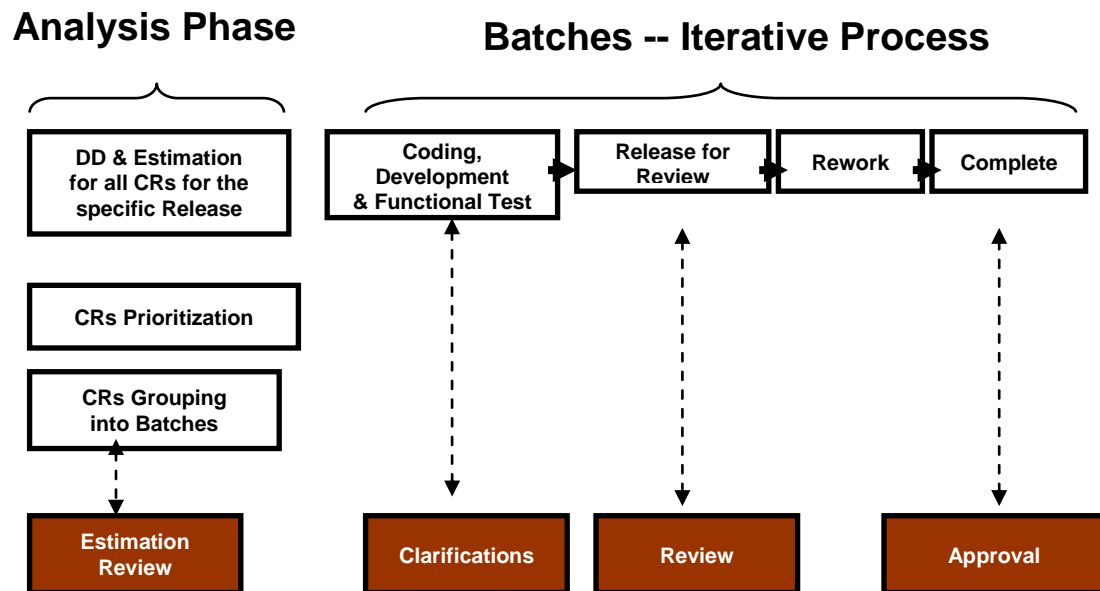
This helps the developer in:

- Understanding the application functionality in a broader perspective than a single CR.
- Performing an Upfront Analysis of 3 to 4 batches of CRs, which helps in prioritizing them based on dependencies.
- Having precise estimation to track the project costs and schedule.
- **In-time customer review:** Every batch release of CRs is reviewed by the customer. This helps avoid too many reworks pushed to the end.

This way of organizing the work had the following benefits:

- Customer centricity: Knowledge of the customer's requirements for CRs at a module-/feature-level gave a bigger picture and improved understanding.
- Improved quality: The number of rework cycles reduced.
- Reduced time-traps and optimal WIP.
- Precise estimation to track the project costs and schedule.

- Overall, increased Return on Capital Employed (ROCE).



4. ROI Measurement Approach

Key factors to be considered for measuring ROI are given below:

4.1. Team Efficiency

4.1.1. Work Delivered

- Efficiency of the team translates to cost-effectiveness:
 - Comparative Productivity.
- Utilization:
 - Percentage of resource utilization or idle time.
- Value-Adds:
 - Additional requests for maintenance/delivery of projects because of staff augmentation for offshore operations.
- Revenue generating efficiency or capacity utilization:
 - End-user is happier than before by ...%. A method to be evolved by customer groups to measure the Key Result Areas (KRAs) for the end-users.

4.1.2. Performance efficiency (This feeds into Comparative Productivity parameter mentioned above).

- Estimation method: The following steps are taken for executing a Maintenance Project:
 - Analysis of the assigned maintenance CR.
 - Review and Approval of the analysis document by the customer.
 - Agreement on schedule and efforts between offshore resource and customer resource for completion of the project.

Method of arriving at the agreement:

- Offshore resource provides the estimation and schedule for executing the project on the basis of Task Breakdown.
- Customer resource reveals the effort and time he would take to complete the same job, on the basis of the analysis document mutually agreed upon.
- The estimation and schedules provided by the offshore resource and the customer resource are compared with each other and an agreement is reached for completion of the work by the offshore resource.
- Performance tracking:
 - Offshore resource executes the work and delivers it.
 - The trend of customer resource estimate vs. offshore resource estimate and the actual time taken for completion of the project is plotted over a period of time. It is then used for measuring offshore efficiency.
- Technical performance parameters:
 - Effort variance.
 - Schedule variance.
 - Number of reworks per task.

4.2. Customer's Time Required

For execution of a Maintenance Project, a customer team is required to devote time to:

- Technical clarifications, reviews, testing – in excess of what is normally done by a customer team member.
- Project management.

4.3. Operational Losses

The following factors lead to operational losses during the execution of a Maintenance Project:

- Network down-time.
- Equipment down-time.

- Time difference between requisition and receipt of equipment/resources.
- Time difference between requisition and receipt of support services.

The ROI trend can be plotted by monitoring these three parameters for measurement of ROI over multiple quarters.

The next section presents a case study which illustrates the implementation of the principles of Lean Manufacturing which is used to set the work execution process. It also showcases the Productivity Tracking Mechanism to measure the productivity of a Maintenance Project.

5. Case Study – Maintenance Project

This case study relates to an enterprise company, for which the maintenance of all back-end applications was executed offshore. The main objectives of this exercise were to:

- Improve the quality of service and reduce the time to service for business groups.
- Have the internal teams focus on strategic initiatives.
- Reduce the maintenance budget and increase ROI.
- Reduce the customer Leads' involvement with the offshore team over a period of time.

In this project, the “**Principle of Lean Manufacturing**” was adapted for the optimal utilization of resources. Productivity was tracked for the mutually agreed SLAs between offshore resource and customer resource.

6. Application of the Principle of Lean Manufacturing

6.1. Analysis of the First Release Cycle of the Project

After the completion of the first release cycle of the project, the following conclusions were drawn:

CR-wise Analysis and Development: For each CR allocated to developers, development was expected to happen immediately after the Analysis. It was presumed that there would not be too many Clarifications or that they would be received immediately. However, this did not happen. As the effort allocated was clubbed with Coding, the coder moved quickly to Coding. This led to defect leakage of Analysis.

Dynamic CR Allocation: On the basis of completion of previous CRs or depending on the time when they got stuck with issues, allocation of CRs to developers happened dynamically. , This gave rise to a lot of interaction among developers each day, leading to frequent revision of plans.

Idle Time: Sometimes developers had to wait for Clarifications. There were instances when despite having waited, they did not receive the Clarifications-and did not intimate the Lead and Manager. Non-escalation of such issues by developers to the PM led to idle time.

Crowding of CRs: All CRs would be addressed to certain team members at the same time which led to a situation where a single developer would work simultaneously on 4-5 CRs.

Reduced predictability: Analysis of CRs was done by all the developers. This resulted in low predictability of delivery. Hence, planning, scheduling and tracking was done regularly. .

Excessive communication with offshore Lead/non-accountability: Two queues – Issue List and Review Queues were with the reviewer. This led to a clash of priority between Issues and Reviews. Similar dilemma existed for the offshore Lead as well and hence the non-progress could not be accounted for.

Effort over-run: Effort over-run due to rework, idle time and limited options of reusability.

6.1.1. Changes Adopted

After this detailed analysis of the first release cycle of the project, a conscious decision was taken to adopt the “Principle of Lean Manufacturing” and “Productivity Tracking” for the Maintenance Project. Certain changes were made to the delivery process to ensure that all the issues discussed above were addressed.

The customer and the vendor mutually agreed to shift from the model for “Managing Resources” to the one for “Managing Results”. In keeping with this decision, the vendor proposed an SLA-based model for project execution.

This document elaborates on the process of engaging in such a model and lists down some terms of the SLA, against which performance of the project will be monitored.

6.1.2. Analysis Phase

For each release cycle CRs were grouped together and an Analysis Phase was planned to be executed before development. This saved the time utilized in performing individual analysis for CRs. In this phase, Analyses of the CRs were planned to be carried out in the first Time-Box. Analyses and Estimates were then reviewed by the offshore Lead. Most of the Clarifications were also provided by the Lead. Each week, Analysis documents were released to the customer with Estimates and Clarifications Lead.

Estimation and Analysis Documents with Identification of Re-usable components were key deliverables of this phase.

This resulted in clarity right from the beginning of development. Segregation of Analyses helped developers concentrate on Coding on a factory mode.

6.1.3. Project Planning with Time-Boxing/Batch-wise Delivery Approach

A Time-Box of three weeks was agreed upon by the customer for the delivery of each batch. Batch planning was done on the basis of priority, critical path, CRs with the least Clarification(s)/the number of Clarification(s) required, and CRs with common functionality.

6.1.4. Benefits of Time-Boxing/Batch-wise Delivery Approach

Some of the major issues that were faced during the analysis of the first release cycle of the project were resolved using the approach of Time-Boxing. It also had other significant benefits, such as:

- No leakage of analysis.
- Static project plan with the least changes.
- Zero idle time.
- No crowding of CRs. At a time, each developer worked/reworked on only one CR.
- Project management/tracking efforts reduced, while predictability increased.
- Well-managed communication and accountability.
- Availability of a re-usable code library for developers.
- Reduced rework effort and no over-run.
- Because of re-usability of code, process rigor and additional CRs were managed without causing any concern to the stakeholders.

6.2. Productivity Measurement Mechanism

PRODUCTIVITY MEASUREMENT

The following steps indicate the methodology used by the vendor to provide the customer with an estimated time for completion of a given CR:

<u>Step</u>	<u>Description</u>	<u>Responsibility on</u>	
Step 0	Allocation of work/assignment by customer to the vendor includes an initial estimated effort.	Vendor	
Step 1	Analysis of work by the vendor: <ul style="list-style-type: none"> o Analysis of the assigned CR: This comprises the following steps: <ul style="list-style-type: none"> § (1) Bug reproduction; § (2) Root-cause identification; § (3) Impact analysis; § (4) Analysis document submission. 		
Step 2	Analysis document review and approval.		Customer

PRODUCTIVITY MEASUREMENT

Step 3	Efforts agreement:	Vendor & customer
	<input type="radio"/> Efforts for Coding and unit testing of the CR are agreed upon, on the basis of the following parameters:	
	<input type="radio"/> Estimation using task breakdown method:	
	(1) Effort required for Coding the fix.	
	(2) Effort required for the CR's peer review by the customer.	
	(3) Effort required for unit testing of the CR offshore.	
	(4) Effort required for expected rework on the CR.	
	(5) Time required for "check outs" and "data extracts" {In some tracks}.	
	(6) Effort to improve the code quality or fix any outstanding defects in the specific unit {In some cases, this needs to be a part of the CR description or the Analysis document}.	
	(7) Effort for creating the migration document {In some tracks}.	
	(8) Effort for checking the final version.	
	Contingency for any other eventualities like the effort required for framing mails to raise queries or attending teleconferences to get their answers.	
	<input type="radio"/> Method:	
	§ Vendor resource provides the estimation of schedule and effort based on points (1) to (8) above.	Vendor
	§ Customer confirms the estimated effort of a customer employee to complete the same job with all the tasks (1 through 8) mentioned above, based on the agreed Analysis document.	Customer
	§ Vendor and customer's estimations are compared and an agreement is reached on effort and schedule for completing the steps (1) to (8) described above.	Vendor & Customer
Step 4	Work execution:	Vendor
	§ Vendor executes the work on the CR and delivers it.	
Step 5	Monitoring the data:	Vendor & Customer
	§ The data is collected on the 'track sheet' used for weekly tracking.	
	§ The trend of customer's effort estimate vs. vendor's actual effort is plotted on a weekly basis.	Vendor
	§ The resultant graph is an indicator of the relative productivity of the vendor's resource.	

PRODUCTIVITY MEASUREMENT

Step 6	<p>Management review:</p> <p>§ The data, along with fortnightly status reports, is shared with customer Managers.</p> <p>§ Any hiccups in implementing this process are escalated on the event basis.</p> <p>§ After checking the feasibility and results for 2-3 release cycles, the vendor productivity is established and tracked.</p>	Vendor & Customer
--------	---	-------------------

6.2.1. Work Execution Approach**Analysis phase:**

- The customer Lead assigns a set of CRs, for release, to the vendor offshore team.
- The vendor team analyzes the CRs assigned to them.
- Based on this analysis, the vendor team estimates the effort required for carrying out the rest of the activities on the CRs.

Planning phase: This phase is the prelude to deciding the work packet for SLAs measurement.

- The customer PM plans meetings (internal to customer) and comes to an agreement on the CRs to be fixed for the next release and the schedule for it, based on the estimates available.
- The customer Lead works towards arriving at a fixed plan for a minimum of three weeks.
- The customer and vendor Leads arrive at a plan for the prioritized trackers, based on the estimates.
- The plan is confirmed for a set of trackers for a minimum of three weeks and if possible, for the complete release.

Execution phase: Offshore project management and delivery; quality checks on sample basis by the customer team

- The vendor Lead works towards fulfillment of the plan by the offshore team.
- Exceptions are applied wherever required. For instance, in situations devoid of clarity, even after a lot of analysis, a decision may be taken either to exclude those CRs from the purview of SLA-tracking or to move them to a subsequent release.
- Once the CRs have been delivered and scanned through QA, UAT etc., appropriate metrics pertaining to the SLA parameters are collected and compared with the threshold values.

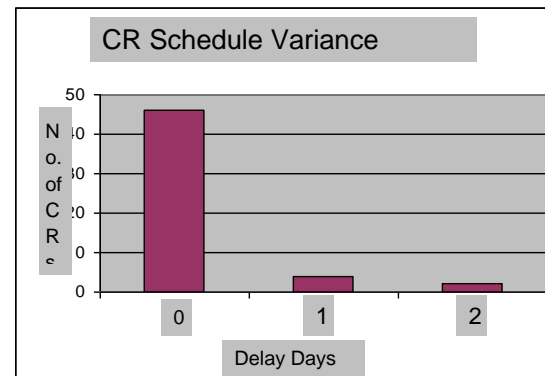
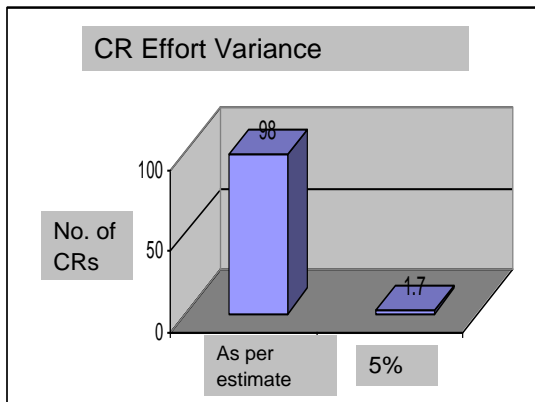
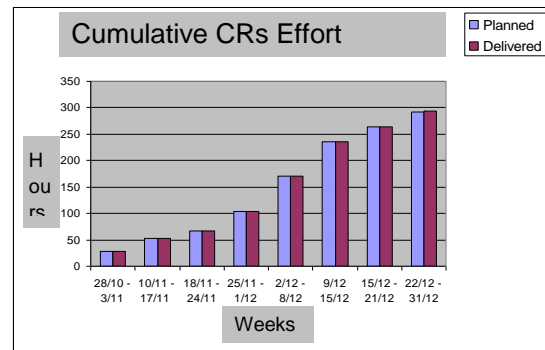
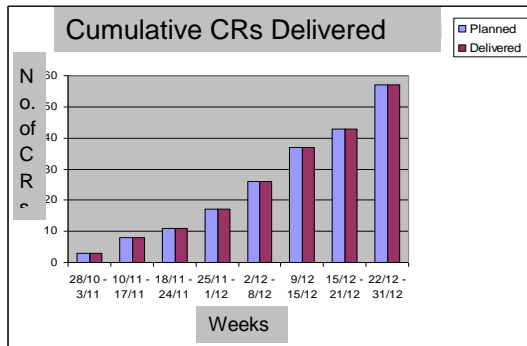
6.2.2. Metrics Used for Productivity Tracking

Parameter	Definition	Formula	Threshold
Effort Variance	Deviation from the mutually agreed effort, measured for each CR.	$\text{Actual effort} - \text{Mutually agreed effort} / \text{Mutually agreed effort} * 100.$	10% of effort agreed.
Schedule Variance	Deviation from the mutually agreed delivery schedule, measured for the release or intermediate milestones, if any, as agreed.	$\text{Actual elapsed time} - \text{Mutually agreed elapsed time} / \text{Mutually agreed elapsed time} * 100.$	10% of mutually agreed elapsed time.
Quality-Functional	Rework cycle = The number of times a CR comes back for bug fixing. There could be more than one defect to be fixed in a cycle of rework.		Rework/failure rate - 90% of the weighted defects must be corrected in the first rework cycle. The SLA fails, if there are more than three rework cycles.
Productivity (Trend analyzed over time)	<p>Notes:</p> <ol style="list-style-type: none"> Scope changes are excluded from this measurement. Rework on “analysis” performed by the offshore team is also measured and monitored. <ol style="list-style-type: none"> Comparison of actual effort against the estimated effort per CR. Total number of weighted CRs corrected per person per month (weight-age is given on the basis of the estimated complexity of the CR). Productivity trend is measured (CRs are categorized on the basis of the effort required for bug fixing, as Simple/Average/Complex. Total weighted CRs delivered release after release are compared for productivity trend). 		

6.2.3. Timelines

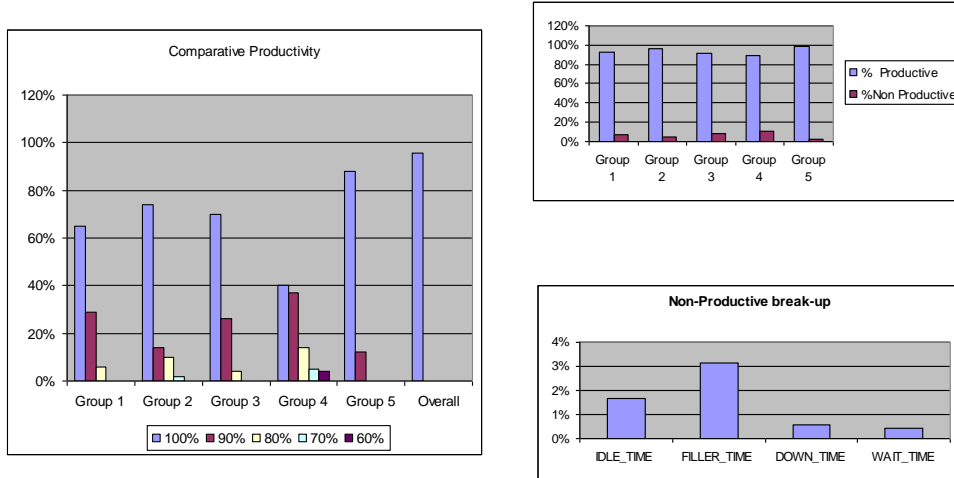
Phase	When	What is expected to be achieved
Trial period	Release x	Execute CRs and measure the SLA parameters as defined above.
Stabilization	Release x+1	<ul style="list-style-type: none"> Monitor performance against CRs. Add/modify the SLA parameters as required. Change the threshold limits as appropriate.
Steady state	Releases following x+1	Operate on the SLA model as evolved.

6.2.4. Project-Level Metrics Collected



Parameter	Definition	Threshold	Actual
Effort Variance	Deviation from the mutually agreed effort, measured for each CR.	10% of the effort agreed.	1.7% of CRs at 5% Effort Variance.
Schedule Variance	Deviation from the mutually agreed delivery schedule, measured for the release or intermediate.	10% of mutually agreed elapsed time.	6% of CRs in <10%Schedule Variance.
Rework Rate	Rework cycle = The number of times a CR comes back for bug fixing. There could be more than one defect to be fixed in a cycle of rework.	90% of the weighted defects must be corrected in the first rework cycle.	100%.
		The SLA fails, if there are more than three rework cycles.	0.

6.2.5. Program Management-Level Metrics Collected



6.2.6. Benefits of Productivity-Tracking Approach

- Delivery and quality are managed by the vendor.
- Customer need not be involved regular work allocation and tracking..
- Clearly-defined parameters for determining the CR's compliance with customer expectations.
- Productivity tracking fine-grained -- to the extent of the resource and the task being delivered.

7. Conclusion

Goals set at the beginning of the project	Status at the end of the project
Improvement in the quality of service and Time-to-Service for business groups	Effort over-run due to idle time; rework was contained by the usage of Time-Boxing approach.
Fine-grained tracking with respect to the budgets	Close tracking was done using detailed tracking sheets and SLAs.
Zero wait-time for team members	Zero wait-time was achieved in most of the groups by the usage of Time-Boxing approach.
Customer Leads' involvement with the offshore team to be optimized over a period of time	By using Time-Boxing approach and reviews, communication by the offshore Lead reduced the customer Leads' involvement to the minimum.
ROI calculation	By using comparative productivity data, one of the key factors of ROI calculation (Sect.4.1) was established. Data was collected for other parameters also and thus, ROI was calculated and presented to the customer.

References:

“Lean Six Sigma,” by Michael L. George.

Abbreviations:

CR: Change Request.

SLA: Service-Level Agreement.

CIO: Chief Information Officer.

KRA: Key Result Area.

For more information, contact info@sonata-software.com

Click here to know more about our [Application Management Services](#)