Improving Software Processes using Metrics Plan

[Based on CMMI Level II Process Areas]

Annie Ibrahim Rana — mscs03-9123@nu.edu.pk
Khadija Saeed — mspm04-0915@nu.edu.pk
Uzaira Saeed — mscs02-9105@nu.edu.pk
Department of Computer Science, Software Engineering Research Group,
National University of Computer and Emerging Sciences, Lahore, Punjab, Pakistan

Abstract

The purpose of measurement programs is to effectively improve the software engineering methodologies followed in an organization. However the potential impact of measurement on the practices of industrial software development has not yet been fully recognized.

In this paper we strive to generate a metrics plan that satisfies the requirements of the Software Engineering Institute (SEI) Capability Maturity Models Integration (CMMI) Level II goals partially as well as meets any organizational goals. This plan covers only two process areas, 'Project Monitoring and Control' and 'Process and Product Quality Assurance' of Level II of CMMI.

This paper starts with an introduction to the CMMI and the two process areas, addressed above, for which we have defined a metrics plan. A brief introduction of the GQM (Goal Question Metric) approach, used in this paper to define metrics, is also given. The 'Metrics Program' framework that is used in this paper to generate the metrics plan is described followed by a complete 'Metrics Plan' which includes GQM Analysis, Metrics Definition, Data Collection, Data Analysis and Reporting. At the end we have mentioned some of the work that we intend to continue in future.

Keywords

CMMI, GQM, SQA, SQAE, PPQA, SG, SP, Work Product, Sub-Practice, Key Practice, Process Area, Metric

1 Introduction

1.1 CMMI Overview

Capability Maturity Model Integration (CMMI) models provide guidance to use when developing processes. CMMI models are not processes or process descriptions. The actual processes used in an organization depend on many factors, including application domain(s) and organization structure and size. CMMI model typically do not map one to one with the processes used in an organization. And this one big misconception that it is solution for every thing so implement CMMI components without taking care of what is required.

Organization can use a CMMI model to help set process-improvement objectives and priorities, improve processes, and provide guidance for ensuring stable, capable, and mature processes. A selected CMMI model can serve as a guide for improvement of organizational processes.

Professional judgment is required to interpret CMMI practices. Although process areas depict behavior that should be exhibited in any organization, all practices must be interpreted using an in-depth knowledge of the CMMI model being used, the organization, the business environment, and the circumstances involved and there are no hard and fast rules to interpret the CMMI. [7]

The components of a CMMI model are grouped into three categories that reflect how they are to be interpreted:

- Required: Specific goals and generic goals are required model components.
- Expected: Specific practices and generic practices are expected model components.
- Informative: Sub practices, typical work products, discipline amplifications, generic practice elaborations, goal and practice titles, goal and practice notes, and references are informative model components.

The general information is as follows:

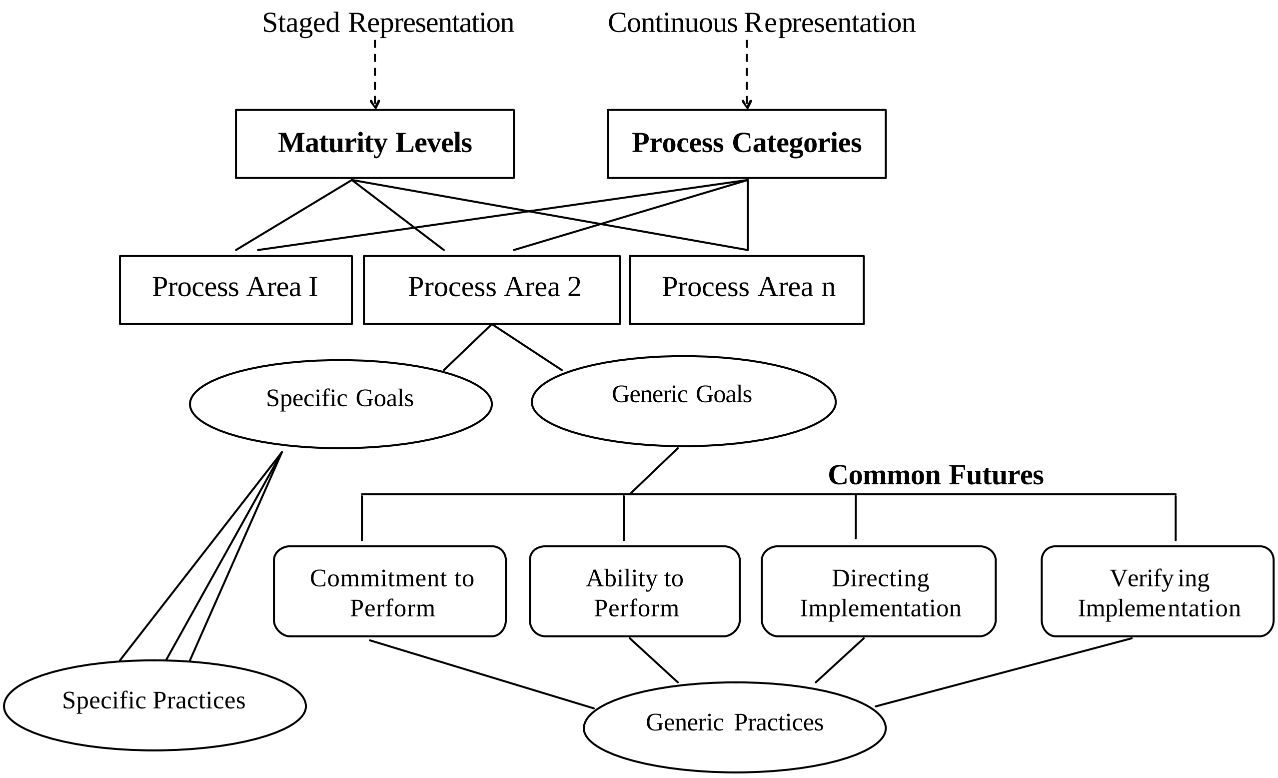

Figure 1: CMMI Structure

There are multiple CMMI models available, as generated from the CMMI Framework. Consequently, but initially there are four purposed models (bodies of knowledge):

- Systems Engineering
- Software Engineering
- Integrated Product And Process Development
- Supplier Acquisition

There are two CMMI model representations:

- Continuous Representation
- Staged Representation

CMMI model structure is shown in figure 1.

There are five maturity levels for staged representation as shown in Table 1:

Maturity Level	Staged Representation Maturity Levels
1	Initial
2	Managed
3	Defined
4	Quantitatively Managed
5	Optimizing

Table 1: Maturity Levels for Staged CMMI

And there are six capability levels for continuous representation as shown in Table 2:

Capability Level	Continuous Representation Capability Levels
0	Incomplete
1	Performed
2	Managed
3	Defined
4	Quantitatively Managed
5	Optimizing

Table 2: Capability Levels for Continuous CMMI

There are 25 process areas organized in staged representation as shown in Table 3.

In continuous representation, Process Areas are organized under process categories as shown in Table 4.

Level		Focus	Process Areas
5	Optimizing	Continuous Process Improvement	1. Quantitative Project Management 2. Organizational Innovation and Deployment
4	Quantitatively Managed	Quantitative Management	1. Organizational Environment For Integration 2. Organizational Process Performance 3. Causal Analysis and Resolution
3	Defined	Process Standardization	1. Requirements Development 2. Technical Solution 3. Product Integration 4. Verification 5. Validation 6. Organizational Process Focus 7. Organizational Process Definition 8. Organizational Training 9. Integrated Project Management For IPPD 10. Risk Management 11. Integrated Teaming 12. Integrated Supplier Management 13. Decision Analysis And Resolution
2	Managed	Basic Project management	1. Requirements Management 2. Project Planning 3. Project Monitoring and Control 4. Supplier Agreement Management 5. Measurement and Analysis 6. Process and Product Quality Assurance 7. Configuration Management
1	Initial		

Table 3: Process Areas organized under maturity levels for Staged CMMI

Category	Process Areas
Process Management	1. Organizational Process Focus 2. Organizational Process Definition 3. Organizational Training 4. Organizational Process Performance 5. Organizational Innovation and Deployment
Project Management	1. Project Planning 2. Project Monitoring and Control 3. Supplier Agreement Management 4. Integrated Project Management 5. Risk Management 6. Quantitative Project Management
Engineering	1. Requirements Management 2. Requirements Development 3. Technical Solution 4. Product Integration 5. Verification 6. Validation
Support	1. Configuration Management 2. Process and Product Quality Assurance 3. Measurement and Analysis 4. Causal Analysis and Resolution 5. Decision Analysis and Resolution

Table 4: Process Areas organized under process categories for Continuous CMMI

The continuous representation uses capability levels to measure process improvement, while the staged representation uses maturity levels.

1.1.1 Maturity Level II: Basic Project Management

The Repeatable Level (Level II) has notion to develop the policies for different software projects and to develop the procedures to implement the policies. Usually, old projects experience is used to plan and manage the new similar projects. Basically, it is an objective of level II to repeat successful practices based on earlier projects to manage new similar projects, although practices may differ in nature but managing style may apply to all similar projects. An effective process can be characterized as practiced, documented, enforced, trained, measured, and able to improve.

Project Monitoring and Control (PMC)

Project monitoring and controlling is required to put the project on right track, with out proper tracking, the risk of project failure raises very high. Normally, monitoring is done with the help of project plan, project plan is the baseline of scheduled work and it is likely compared with actual. In comparison of schedule with actual, we not only compare the task status according to their dates but we also measure the status of other resources i.e. cost, time, resources and effort etc.

The project plan covers all other plans i.e. development plan, management plan, quality assurance plan, configuration management plan etc. The plan should be comprehensive and detailed. When actual status deviates significantly from the expected values, corrective actions are taken as appropriate. These actions may require re-planning, which may include revising the original plan, establishing new agreements, or including additional mitigation activities within the current plan.

Process and Product Quality Assurance (PPQA)

Software Quality Assurance intends to provide management with appropriate visibility into the process being used by software projects and of the, products being built. This will determine as to whether the software project is adhering to its established plans, standards, and procedures.

The software quality assurance process will be implemented during the early stage of a software project to establish plans, standards, and procedures that will not only add value but will also satisfy constraints of the project using policies. The software quality assurance Process intends to fulfill the project's needs and verifies that it will be usable for performing reviews and audits throughout the software development life cycle by software quality assurance group.

1.2 Goal Question Metric

The Goal Question Metric (GQM) was developed to create a goal-oriented approach for the measurement of processes and products in the software engineering. The GQM has top-down approach to define the goals to focus on key objective of software process and product and using these goals to measure (or defining metrics for) processes and products. The GQM additionally supports a bottom-up approach to interpreting data based on the previously defined goals and questions. [8]

GQM has three major parts:

- Goal
- Question
- Metric

Goal is further divided into sub parts:

- Object
- Purpose
- Quality Focus
- Viewpoint
- Environment

A single goal plus the sets of questions and metrics that provide an operational definition of that goal, is called GQM Model or Q\GQM Plan. A GQM plan documents the refinement of a precisely specified measurement goal via a set of questions into a set of metrics. Thus, a GQM plan documents which metrics are used to achieve a measurement goal and why these are used - the questions provide the rationale underlying the selection of the metrics. On the other hand, the GQM plan is used to guide analysis tasks because it documents for which purpose the respective data were collected. [4, 8]

When coupled with a GQM Plan, a measurement plan specifies who collects the data required by the GQM Plan, how the data is collected, and when the data must be collected. A measurement plan usually includes the data-collection forms as well as descriptions of tools that perform online data collection.

2 THE METRIC PROGRAM

We have chosen a well defined framework for measurement. The details of metric program, what we have followed, are given below:[9]

2.1 Documenting the Software Process

In this modern age today we live in one Universe, one World and working together in different organizations with different cultures. The foremost advantage of the metric program is focus on the software process of an organization typically with the concern that while developing the software the organization complies with the documented processes.

Goals	Considerations for Choosing the Goals
Development Process Improvement	✍ To confirm with some specified standards (e.g. CMMI, ISO etc.) ✍ To increase maturity level with a certain standard (e.g. CMM, CMMI etc.) ✍ To increase the efficiency and output of staff within the same resources
Improvement in Software Estimation	✍ To be more accurate with the project proposals ✍ To be accurate with project budgets ✍ To be accurate with project schedule ✍ To be more profitable ✍ To ignore requirements volatility ✍ To minimize any possible risks
Improvement in Project Tracking	✍ To timely predict the need for corrective action ✍ To ensure the conformance with some already specified standard
Minimize the Time for Software Development	✍ To ensure timely delivery of products ✍ Minimize the time for maintenance or production of a software product
Minimize the Cost for Software Development	✍ To ensure the delivery of any software product within budget ✍ To increase the profitability, in monetary terms, in software development
Software Quality Improvement	✍ To ensure that the software products comply with the end user requirements ✍ To minimize bugs ✍ To minimize the time being wasted in any rework
Improvement in Software Performance	✍ To ensure that maximum performance goals are met ✍ To reduce in hardware performance requirements
Increase Productivity	✍ To bring in stability in staffing levels

Table 5: Process Goals and Considerations

Thus the metrics program may also help in software process improvement efforts. To bring consistency within our work environments, we need to define and document the software process formally so that it is followed uniformly across all the projects which most of the organizations do in a Quality Plan. Once the software process is defined, documented and being followed we can easily and consistently find data across all the projects.

Here we would like to mention that, we have presented any process for documentation as we are considering the generic processes defined by CMMI and also we suppose that the organizations that are launching metrics program with the target for achieving CMMI standardization will like to consider such work.

2.2 State the Goals

For generating a metrics program we need to have a set of goals as the goals specify our aim for generating a metrics program. Normally the software organizations may have some or all of the many goals mentioned in Table 5. As we can see that most of the goals mentioned in Table 5 is closely related and thus more or less similar data will be required to satisfy these goals.

For achieving a goal we have got our own motivations and objectives such that if our objectives are achieved then the specified goals are satisfied automatically. Since we are working only for the Project Monitoring and Control and Software Quality Assurance process areas of CMMI, therefore, we will just consider the specific goals, defined by SEI, for these process areas.

Normally for a metrics program, the goals are well documented and explained in the "Metrics Program Description" document.

2.3 Define Metrics Required to Reach Goals

At this step we have to analyze the goals that we have specified in step2. Based on the analysis of the goals and by following the GQM (Goal/ Question/ Metrics) approach, we come forward with a set of questions to address every goal. To answer these questions, we have to determine some specific metrics on which our program will be based.

The proposed metrics should be simpler in the beginning just to take a start with the metrics program and later more sophisticated and complicated metrics can be determined but at every stage the metrics must be realistic enough, to be analyzed and for whom we can collect data, to answer these questions properly and efficiently.

Here we will like to mention that the selection and definition of right metrics is a very risky task to be done as if those metrics are under use which do not really target our goals, then sheer time and resources being used to determine their value are wasted without much of productivity.

2.4 Identify Data to Collect

Since we have now come up with a set o logical metrics, therefore, we now need to collect meaningful data for each and every metric. Data collection procedures and methodologies are explained in details in another section. We have also explained what is the valid form of data to be collected and how it should be collected and finally how should it be stored. Following the mentioned practices and methodologies, any software organization can collect data and store it in some repository preferably a database and then will use it for its metrics program.

2.5 Define Data Collection Procedures

Of course it is impossible to devise data collection procedures that are practically applicable to any software organization. However, since we have not collected any data by ourselves, in this section we will explain some general data collection procedures and guidelines that can be customized according to the needs for collecting data from any organization.

The general guidelines for collecting the data are as follows:

How to collect data:
While collecting data we need to take care for its origination and also at what stage the data was collected by the measurement team. Here, if the collected data comes out to be the output from some tool or an automated procedure, it's more secure to take care of the following points:

- Where are the tools found?
- Who keeps the tool in running order?
- How to get data from the tool? and
- Where are the methodologies about the usage of tool documented?

When the data is collected:
We need to take care that, we formally document every exact point at which we have collected the data.

Formally identify, who is responsible to collect data:
For security purposes we formally need to identify; who collected the data, stored it in metrics database and transformed it to complex metrics; along with the data actually collected.

Where the collected data is stored:
All the means of data storage that are being used are identified. These may include emails, forms, documents as well as the name and location of metrics database.

Ensure the validity and correctness of data:
We have to make sure that the data being collected is consistent all over and also need to have procedures that help dealing with erroneous data.

So far we have been with the overview of the data collection procedures. In the coming lines we will look into these procedures in more details. Here, we will like to add that while following these procedures in practice, we need to take care that the data collection methodologies must be simple otherwise it may grow practically impossible to collect any data!

2.6 Assemble a Metrics Toolset

To, ease the process of data collection for metrics we need to have an effective tool. Selection and use of an effective toolset compliments the effectiveness of the metrics program. The selected software tool should be user friendly, should be simple and rich in functionalities and preferably should be platform independent.

Many professional tools are commercially available for different data groups that we have once identified in the above step.

Although there are many tools that are commercially available for data collection but we have just make use of two tools: Microsoft Word and Microsoft Excel. The reason for their selection was that these tools are not only professional and user friendly, as they equipped with help at every step, but also they these are general purpose tools, simple and help collecting many types of data.

2.7 Creating a Metrics Database

If we look into the processes of an organization and try to collect data for metrics then naturally we have to look across all the projects. Thus we will come up with a very large data set which needs to be stored in some metrics database. The database should comprise of the following characteristics:

- Database should be easy enough in use typically in terms of adding or updating any data set.
- The structure of database must be flexible such that if new data is collected for the metrics then it's easily accommodated.
- The database must not be troubling in case interfacing with multiple tools is required. If the database can conveniently interface with different kind of tools then its expected that the consistency of data is maintained across the organization.
- The database must have got a professional reporting facility in it.
- The database must also provide the users with large storage capacity so that the historical data can also be maintained.
- The database must avoid data redundancy, that is, it should support facilities like the concepts of foreign key etc.
- It should also be equipped with security features, that is, should provide users with the restrictions to use only that part of data which is assigned to their roles.

Other than the basic features of the databases mentioned above, the database may also provide users with facilities like data distribution etc. based upon the requirements. Two most commonly use data storage facilities are databases and spread sheets. Altho ugh the spreadsheets provide the users with a very friendly interface and also it is in expensive in use but databases are effective for handling very large data sets and thus provide us with an effective and long lasting solution.

Based upon the level of the metrics program (that is, whether it's a basic level program or an advanced level program), its future considerations and above all depending upon the size of the organization for which we are working we have to select some very appropriate tool based upon our fair judgments. Since we have not collected any data from any software organization, therefore, we are not using any of the above mentioned methodologies for data accommodation.

2.8 Define the Feedback Mechanism

So far, the following tasks must need to be carried out:

- We have got all the information in an organized manner and the information is also presented to all the staff such as to ensure that the information is authentic and correct.
- We have received recommendations for process improvement from within the staff. These recommendations are then analyzed and prioritized and finally selected to bring in use.
- Changes and enhancements in process are defined, documented and brought to practice.
- Metrics program is generated and tracked for our goals.
- Once the metrics program is successful, we can later revise the program with better and more advanced revised goals.

A proper feedback mechanism which is comes to be with minimal communication gap and is liberal in nature assures the stake holders that all information is communicated in a pretty proper way and that the metrics program is initiated in a right direction.

For making the metrics program run, the data related to metrics is tabulated and presented in a useful format. Other than this the data can be presented to all the stakeholders through organized presentations. The results that we get from the metric program can be presented in one of the two agreed upon ways:

- Reports
- Presentations

Reports are supposed to be more formal and detailed than a presentation with the focus on the pre defined goals. On the other hands, through presentations we can just give an overview of the work and not its details. Through the meetings, more and more of data can be further collected and authenticated. In case we need to have huge amount of data then there should be more of meetings as we can get clues about the process customization only through the data collected especially by these meetings. On the other hand, these meetings are also useful to convey and discuss the changes in processes for the organization.

As discussed above, the reports and generations are prepared while targeting the goals, however, the performance issues must be considered along with as to increase in performance is one of the basic themes for generating a metrics program.

Based upon the meetings for metrics review the baselines need to be revised and redefined. Based upon the baseline data we can access the software development project at any stage. Also on the basis of results being analyzed and discussed we will try modifying our development processes to target the goals more effectively. The changes decided are propagated and spread across the staff and management by emails, memos or more formally by meeting in order to make every one aware of what is being decided and why. Every body's concerns, in this regard, must be recognized and considered.

With the advancements in the metrics program, the goals will later be revised and prioritized and thus of course new metrics will be required to be collected. This all will be done by again repeating through all the lengthy steps from 1 till the end.

Metrics program is a going strategy that needs to be revised, improved and enhanced after regular intervals of time. This will for sure lead us to a stage of continuous process improvements which is a full world in itself. And this is just the start…

3 Metrics Plan

Our Metrics plan is based on the Measurement Program discussed in the previous section. The scope of the metrics plan is: Metrics Identification, Data Collection, Data Analysis and Reporting. [1, 2, 3, 4]

3.1 Goal, Question, Metric Analysis

As explained earlier, A GQM plan documents the refinement of a precisely specified measurement goal via a set of questions into a set of metrics. The GQM for PPQA and PMQA are separately documented in Report of 65 pages.

3.2 Metrics Identification

After coming up with metrics by doing GQM analysis, the following final list of Metrics has been identified along with the Description, Data Group, Metric Type, Complexity and Application Area of each metric.

3.2.1 Process & Product Quality Assurance

No.	Metrics	Description (Measures)	Data Group	Type	Complex	Area
1	Training Variance	Planned no. of Trainings - Actual no. of Trainings	Requirements data	Indirect /Derived Measurement	Low	Process
2	Trainer Variance	Planned no. of Trainers - Actual no. of Trainers	Staffing data	Indirect /Derived Measurement	Low	Process
3	Trainees Variance	Planned no. of Trainees - Actual no. of Trainees	Performance data	Indirect /Derived Measurement	Low	Process
4	Evaluation Variance	Planned no. of Evaluations - Actual no. of Evaluation	Performance data	Indirect /Derived Measurement	Low	Process
5	Work Products Completion Ratio	Actual No. of Work Products/Planned No. of Work Products	Performance data	Indirect /Derived Measurement	Low	Product
6	Processes Completion Ratio	Actual No. of Processes/Planned No. of Processes	Performance data	Indirect /Derived Measurement	Low	Process
7	Environment Efficiency	No. of Issues reported	Performance data	Direct Measurement	Low	Process
8	Criteria Establishment Analysis	No. of Planned Criteria Activities	Requirements data	Direct Measurement	Medium	Process
9	Criteria Effectiveness/ Usage	No. of criteria usage cases	Performance data	Direct Measurement	High	Process
10	Evaluation Deviation	No. of evaluation cases Vs No. of criteria usage cases	Project management data	Indirect /Derived Measurement	High	Process

No.	Metrics	Description (Measures)	Data Group	Type	Complex	Area
11	Non-Compliances Issues	No. of Non-Compliances	Defect data	Direct Measurement	Low	Process, Product
12	Issue Analysis Efficiency	No. of Issues Vs No. of Issue Evaluation Reports	Performance data	Indirect /Derived Measurement	Medium	Process
13	Corrective Action Efficiency	No. of Corrective and Preventive Actions Vs No. of Issues Reported	Performance data	Indirect /Derived Measurement	High	Process
14	Planned Work Products	No. of Planned Evaluation Work Products	Requirements data	Direct Measurement	Low	Process
15	Criteria Variance	Planned No. of Criteria Activities - Completed No. of Criteria Activities	Project management data	Indirect /Derived Measurement	High	Process
16	Work Products Status	Planned no. of work products to evaluate Vs Actual no. of evaluated work products	Performance data	Indirect /Derived Measurement	Low	Product
17	Milestone Status	Planned evaluation milestones Vs Actual evaluated milestones	Performance data	Indirect /Derived Measurement	Low	Process
18	Work Products Status	Planned no. of work products to be evaluated in progress VS Actual no. work products evaluated	Performance data	Indirect /Derived Measurement	Low	Product
19	Services Status	Planned no. of services to be evaluated in progress VS Actual no. of services evaluated	Performance data	Indirect /Derived Measurement	Medium	Process
20	Work Products Status	Planned no. of services to be evaluated incrementally VS Actual no. work products evaluated	Performance data	Indirect /Derived Measurement	High	Product
21	Impact Analysis	No. of reported issues, No, of solved issues, No. of un resolved issues	Project management data	Direct Measurement	Low	Process
22	Issue Resolution Commitment	Average no. of resources assigned to each issue, No. of unresolved Issues, No. of escalated issues	Project management data	Direct Measurement	Medium	Process
23	Issue Resolution Efficiency	No. reported Issues Vs Resolved Issues	Performance data	Indirect /Derived Measurement	Low	Process
24	Issue Trend	No. of similar Issues, No. of issues having similar solution	Project management data	Direct Measurement	High	Process
25	Stakeholder Involvement	No of evaluation reports Vs No. of evaluations	Project management data	Indirect /Derived Measurement	Low	Process
26	Periodic reviews Efficiency	Planned no. of reviews Vs Actual No. of reviews, No. of open Issues reviewed	Performance data	Direct/ Indirect Measurement	Low	Process

No.	Metrics	Description (Measures)	Data Group	Type	Complex	Area
27	Tracking Status	No. of Issues with open status, No. of Issues with close status	Performance data	Direct Measurement	Low	Process
28	Issue Reporting Effectiveness	No. of status reports, No. of non compliance reports, No. of deviation reports	Performance data	Direct Measurement	Low	Process

Table 6: Process and Product Quality Assurance Metrics

3.2.2 Project Monitoring & Control

No.	Metrics	Description (Measures)	Data Group	Metric Type	Complex	Area
1	Number of tasks	Number of tasks = Total number of scheduled tasks	Requirements data	Direct Measurement	Low	Process
2	Variance in progress	Variance in progress = (Num ber of planned tasks - actual number of tasks performed)	Requirements data	Indirect/Derived Measurement	Low	Process
3	Variance in time schedule	Variance in schedule = (Number of planned man hours - actual number of man hours spent)	Effort data	Indirect/Derived Measurement	Low	Process
4	Variance in monetary resources	Variance in monetary resources = Variance in time schedule * average cost per man hour	Financial data	Indirect/Derived Measurement	High	Process
5	Variance in work products w.r.t product	Variance in work products w.r.t product = # of inconsistencies found between work products and product	Performance data	Direct Measurement	Medium	Product
6	Variance in work products w.r.t. tasks	Variance in work products w.r.t. tasks = # of inconsistencies found between work products and tasks	Performance data	Direct Measurement	Medium	Product
7	Variance in resources	Variance in resources = (Number of planned resources - actual number of resources)	Project management data	Indirect/Derived Measurement	Low	Product
8	Effect on internal commitments	Effect on internal commitments = Variance in time schedule * average cost of a single man hour	Financial data	Indirect/Derived Measurement	High	Product
9	Effect on external commitments	Effect on external commitments = Variance in time schedule * average cost of a single man hour	Financial data	Indirect/Derived Measurement	High	Product
10	Number of commitments	Number of commitments = Total number of commitments that have been made	Project management data	Direct Measurement	Low	Product
11	Number of commitments due so far	Number of commitments due so far = Total number of commitments that were supposed to be fulfilled so far	Project management data	Direct Measurement	Low	Product
12	Number of commitments successful so far	Number of commitments successful so far = Total number of successful commitments	Project management data	Direct Measurement	Low	Product

No.	Metrics	Description (Measures)	Data Group	Metric Type	Complex	Area
12	Number of commitments successful so far	Number of commitments successful so far = Total number of successful commitments	Project management data	Direct Measurement	Low	Product
13	Number of remaining commitments	Number of remaining commitments = Number of commitments - Number of commitments due so far	Project management data	Indirect/Derived Measurement	Low	Product
14	Total number of risks	Total number of risks = Sum[Nk(Ck)] where, Nk = Number of risk factors for commitment number k; C is a commitment whose index number is k and k is a constant whose value ranges from 0 to the total number of commitments	Estimation data	Indirect/Derived Measurement	High	Product
15	Number of risky commitments	Number of risky commitments = Sum of commitments where Nk > T where, T is a threshold value which varies from project to project and from schedule to schedule	Estimation data	Indirect/Derived Measurement	High	Product
16	Number of reviewed commitments	Number of reviewed commitments = Total number of commitments that have been reviewed	Effort data	Direct Measurement	Low	Product
17	Number of documented and reviewed commitments	Number of documented and reviewed commitments = Total number of reviewed commitments that have been documented	Effort data	Direct Measurement	Low	Product
18	Task Completion Ratio	No. of tasks with close status/Total No. of tasks	Implementation data	Indirect/Derived Measurement	Low	Process
19	Schedule Slippage/Estimation Deviation	No. of scheduled man hours - No. of actual man hours	Effort data	Indirect/Derived Measurement	Low	Product
20	Work Product Consistency Ratio	No. of work products with inconsistencies/ Total no. of work products	Performance data	Indirect/Derived Measurement	Medium	Process
21	% of Expected Risks	(Number of risks expected in the current phase / Total number of risks estimated initially) *100	Estimation data	Indirect/Derived Measurement	Medium	Process
22	% of Updated Risks	(Number of risks modified / Total number of estimated risks) *100	Estimation data	Indirect/Derived Measurement	Low	Process
23	% of Incoming risks not communicated to stakeholders	(Number of incoming risks not communicated to stakeholders / Total number of incoming risks) *100	Estimation data	Indirect/Derived Measurement	Medium	Process
24	Difference in Planned and Actual Work product Reviews	Planned number of reviews of work products - Actual number of reviews of work products	Performance data	Indirect/Derived Measurement	Low	Product
25	Deviation in Project Baselines	Planned number of project Baselines - Actual number of Project baselines	Project management data	Indirect/Derived Measurement	Low	Process
26	Difference in Planned and Actual number of work Products	Planned number of work products - Actual number of work products	Requirements data	Indirect/Derived Measurement	Low	Product

No.	Metrics	Description (Measures)	Data Group	Metric Type	Complex	Area
27	Activity Reviews Conformance (Work Products)	(Number of activity reviews in which non-conformities are identified / Total Number of reviews) *100	Project management data	Indirect/Derived Measurement	Low	Process
28	Number of non-conformities identified (Work Products reviews)	Total number of non-conformities identified in the reviews	Performance data	Direct Measurement	Low	Product
29	Percentage of Stakeholder Reviews conducted	(Number of meetings actually conducted to review the activities / Number of meetings planned with the stakeholders to review the status of activities) *100	Project management data	Indirect/Derived Measurement	Low	Process
30	Percentage of incomplete activities	(Activities planned to finish but have not yet been completed / Total Activities that were scheduled to be completed to date) *100	Performance data	Indirect/Derived Measurement	Medium	Process
31	Undocumented Significant Issues (stakeholder reviews)	Number of significant issues identified - Number of significant issues documented	Project management data	Indirect/Derived Measurement	Low	Process
32	Stakeholder status review reports generation accuracy	Number of stakeholder involvement status review reports generated / Number of stakeholder involvement status reviews conducted	Project management data	Indirect/Derived Measurement	Low	Process
33	Percentage of High Priority Issues (Stakeholder status Reviews)	(Number of high priority issues identified in each review / Number of issues highlighted in each review) *100	Project management data	Indirect/Derived Measurement	Medium	Process
34	Percentage of Progress Reviews conducted	(Actual number of progress reviews with project team / Planned number of progress reviews with project team) *100	Project management data	Indirect/Derived Measurement	Low	Process
35	Percentage of Sr. Mgmt Reviews conducted	(Actual number of reviews with sr.mgmt / Planned number of reviews with sr.mgmt) *100	Project management data	Indirect/Derived Measurement	Low	Process
36	Percentage of Client Reviews conducted	(Actual number of reviews with client / Planned number of reviews with client) *100	Project management data	Indirect/Derived Measurement	Low	Process
37	Effort estimation Accuracy (Measurement Activities)	Actual Effort expended on measurement activities / Planned effort on measurement activities	Effort data	Indirect/Derived Measurement	Low	Process
38	Undocumented Significant Issues (Progress reviews)	Number of significant issues identified - Number of significant issues documented	Project management data	Indirect/Derived Measurement	Low	Process
39	Undocumented Significant Deviations	Number of significant deviations identified in the project - Number of significant deviations documented, i.e, corrective actions suggested and preventive actions taken.	Project management data	Indirect/Derived Measurement	Medium	Process
40	Service Efficiency (Change	(Number of changes implemented / Number of changes requested) *100	Implementation data	Indirect/Derived Measurement	Medium	Product

No.	Metrics	Description (Measures)	Data Group	Metric Type	Complex	Area
41	Service Efficiency (Problem Reports)	(Number of problems fixed / Number of problems reported) *100	Defect data	Indirect/Derived Measurement	Medium	Product
42	Progress review reports generation accuracy	Number of progress review reports generated / Number of progress reviews conducted	Project management data	Indirect/Derived Measurement	Low	Process
43	Percentage of High Priority Issues (Progress Reviews)	(Number of high priority issues identified in each review / Number of issues highlighted in each review) *100	Project management data	Indirect/Derived Measurement	Low	Process
44	Percentage of Assigned CRs	(Number of change requests assigned to the developers / Total Number of change requests submitted) *100	Project management data	Indirect/Derived Measurement	Medium	Process
45	Percentage of Tested CRs	(Number of change requests implemented in the system and tested / Total Number of change requests submitted) *100	Implementation data	Indirect/Derived Measurement	Medium	Product
46	Percentage of UAT Tested CRs	(Number of change requests tested by the client/ Total Number of change requests submitted) *100	Implementation data	Indirect/Derived Measurement	Medium	Product
47	Number of documented issues	Number of documented issues = Number of reported issues + Number of issues that have not been reported	Project management data	Indirect/Derived Measurement	Low	Product
48	Total number of corrective actions	Total number of corrective actions = Sum[Nok(Ik)] where, Nok = Number of corrective actions for issue number k; I is an issue whose index number is k and k is a constant whose value ranges from 0 to the total number of issues	Project management data	Direct Measurement	High	Process
49	Change in internal commitments	Change in internal commitments = (Number of resources * Number of man hours per resource required to document the actions) + (Number of resources * Number of man hours per resource required to implement the actions)	Staffing data	Indirect/Derived Measurement	Medium	Process
50	Complexity of issue	Number of resources * Number of man hours required to resolve the issue	Effort data	Indirect /Derived Measurement	Medium	Process
51	Change in internal commitments	(Number of resources * Number of man hours required to document the actions) + (Number of resources * Number of man hours required to implement the actions)	Effort data	Indirect /Derived Measurement	Medium	Process
52	Change in external commitments	Change in external commitments = (Number of resources * Number of man hours per resource required to document the actions) + (Number of	Effort data	Indirect /Derived Measurement	Medium	Process

No.	Metrics	Description (Measures)	Data Group	Metric Type	Complex	Area
53	Variance in corrective actions	Variance in corrective actions = Planned number of corrective actions - actual number of corrective actions that produced	Project management data	Indirect /Derived Measurement	Low	Process
54	Variance in resolved issues	Variance in corrective actions = Planned number of corrective actions - actual number of corrective actions that produced desired results	Project management data	Indirect /Derived Measurement	Medium	Process

Table 7: Project Monitoring and Control Metrics

3.3 Data Collection

Data collection is an important part of the metric plan. This part has many referenced data extracted from different research papers. The detail of data collection is discussed below: [2, 3, 10]

3.3.1 Data Collection Procedure

Manual recording is subject to bias, error, omission, and delay. Automatic data capture is therefore desirable, whenever possible. In case if automatic data capturing tool is not available then we should do it manually.

A form should be self-explanatory, allow also free-format remarks, and have pre-printed data where possible (no-changing data).

3.3.2 Data Collection Scheduling Procedure

It is essential that the data-collection activities become part of the regular development process. Thus, it is helpful to compare a model of the normal development process with a list of desired measurements, and then map the measurements directly to the process model. In this way, if something needs to be measured and there is no obvious place to collect data, the process can be modified to enable measurement.

3.3.3 Data Storage and Extraction Procedure

Raw Software Engineering data should be stored on a database. A database structure is highly flexible and, provided the necessary measurements have been recorded properly, will permit the extraction of sets of refined data for various kinds of analysis.

3.4 Data and Metric

There are many related data with every metric. E.g. data groups, metrics type, complexity level, process area etc. Related information is discussed below.

3.4.1 Data Groups

There are many groups of data; groups help to differentiate data for reporting purpose. Different groups used in this research study are given in the Table 10.

3.4.2 Metric Types

There are three types of metrics, as given in following table. Direct metrics and indirect metrics have been used in this research paper.

Types help in reporting purpose as shown in Table 11.

3.4.3 Metric Complexity Levels

There are three levels of complexity for metrics. Measurement should be started with simple and less complex metrics.

The complexity levels are given in the Table 12.

3.4.4 Application Areas

There are two types of application areas for metrics, metrics for a process, and metrics for a product.

Sr. No	Data Group	Description
1	Estimation data	Initial Cost And Schedule Estimation Data
2	Effort data	Data Measuring Person Hours Worked
3	Staffing data	Project staffing level data
4	Project management data	Activity, Unit, And Task Status Information
5	Financial data	Data Concerning Project Financials
6	Requirements data	Project Requirements And Specifications
7	Implementation data	Data Generated During Implementation
8	Testing data	Data Generated During Testing
9	Defect data	Data Concerning Software Defects
10	Performance data	Measured Software Performance Data

Table 8: Data Groups

Sr. No.	Metric Type	Description	Used
1	Direct Measurement	These measurement activities involve no other attributes or entities	Yes
2	Indirect/Derived Measurement	These are combinations of other measurements	Yes
3	Predictive Measurement	These measurements are systems consisting of mathematical models together with a set of prediction procedures for determining unknown parameters and interpreting results.	No

Table 9: Metric Types

Sr. No	Complexity	Description
1	Low	Simple Metric, easily measurable, no expert level skills and specialized tools
2	Medium	Medium complexity Metric, requires expert level skills but no specialized tools
3	High	High complexity Metric, requires expert level skills and specialized tools

Table 10: Complexity Level

Sr. No.	Areas	Description
1	Process	Effort & Time
2	Product	Quality, Complexity, Size & Functionality

Table 11: Metric Focus Areas

Step No.	Description
Step 1:	Decide which data collection procedures are applicable to the data identified
Step 2:	Create any forms that are required for the collection.
Step 3:	Assign responsibility for metrics collection and ensure that the responsibility is agreed upon and documented.
Step 4:	Update any design or project documentation templates to include sections for data that must be collected.
Step 5:	Update the metrics documentation to include details on metrics collection procedures for the metrics coordinator(s).
Step 6:	Update any development process documentation to include relevant data collection procedures. Indicate which tasks and activities are affected.

Table 12: Data Collection Steps

In our research study, majority of the metrics are related to the processes and few are indirectly for product (through work products). The description is given in the Table 13.

Sr. No.	Role.	Description	Responsibilities
1	SQAM	Software Quality Assurance Manager	Track and Monitor the Measurement Process
2	PM	Project Manager	Track and Monitor The Project Progress
3	SQATL	Software Quality Assurance Team Lead	Measure the Project Performance and Report
4	PTL	Project Team Lead	Measure Project Performance at Development Team level
5	PC	Project Coordinator	Measure and Coordinate between SQA and Development Team
6	SQAE	Software Quality Assurance Engineer	Review and Report

Table 13: Roles and Responsibilities

3.4.5 Measurement Actions

There are six steps involved in measurement. The detail of each step is given in the Table 14.

3.4.5.1 Roles and Responsibilities

There are six roles to collect the data for metrics and detail of each role with responsibility is given in the Table ?.

3.4.6 Data Collection Forms

We have designed two generic forms, one is project or product specific and other is process specific. The details are given below:

3.4.6.1 Process Data Collection Form

Form A is process specific form to collect measurement data to analyze the performance of the process. Form has two parts: header and footer.

Header part has general data fields: Name (Process name), ID (Process Id if there is any), Process Objective (Goal) and data collection Date.

Data Collection Form A			
Name:		**ID:**	
Objective:		**Date:**	

No.	Project Name	Group	Planed	Actual

Resource Person Name:	

Form 1: Data Collection Form A

Data Collection Form B			
Name:		**ID:**	
Objective:		**Date:**	
Project:		**Group:**	

No.	Task	Planned Date	Actual Date	Status

Resource Person Name:	

Form 2: Data Collection Form B

The Footer part has the list of projects under study for measurement, list of group working on those projects and the parameters planned and actual property. Parameter means the target measure which is required to be monitored e.g. resources, cost, time, effort etc.

And parameter property means any thing related to the parameter which depicts the meaning of measuring or interprets the current state of parameter e.g. in case of cost as a parameter, it is easy to see planned cost verses actual cost but in case of time as parameter, we will need to see the time related property "Planned Date Verses Actual Date". The data collection template is given in Figure 2.

3.4.6.2 Project Product Data Collection Form

Form B is project/product specific form to collect measurement data to analyze the performance of the project/process in terms of any process or feature. Form has two parts: header and footer.

Header part has general data fields: Name (Project name), ID (Project Id if there is any), Project Objective (Goal) and data collection Date.

The Footer part has the list of task under study for status measurement or performance measurement, and the parameters planned and actual property.

Parameter means the target measure which is required to be monitored e.g. resources, cost, time, effort etc.

And parameter property means any thing related to the parameter which depicts the meaning of measuring or interprets the current state of parameter as described earlier. The data collection template is given in Figure 3.

3.5 Data Analysis

3.5.1 Process Overview

The purpose of this process is to analyze and interpret the data gathered during the Data Collection phase in order to determine whether project's performance is going on as planned. This involves determining the current value of any of the metrics at a given point in time and taking corrective actions in case the value deviates from the expected values.

In addition, it also involves analyzing those project goals against which some sort of deviation is expected in the near future and suggesting corrective actions in case those goals deviate from their expected values.

At organization level, the data collected from all projects is analyzed at certain points to see whether the organization's performance is going as planned or not. The results of the analysis are documented in analysis reports and published in a measurement database.

The figure further 2 illustrates the Data Analysis process.

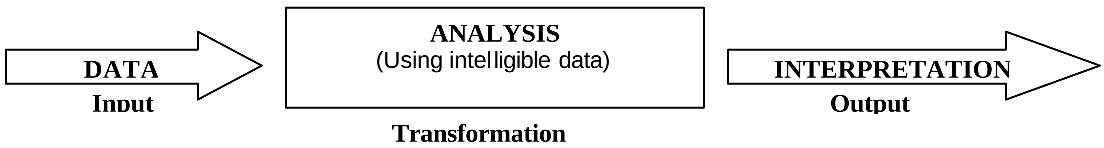

Figure 2: Process Overview

3.5.2 Roles and Responsibilities

Following are the roles and responsibilities specifically for the Data Analysis phase.

SEG (Software Engineering Group):

- Project Manager and Team Leads are responsible for:
  - Performing the analysis of data collected and comparing the actual measurements to the limits set for the measurements.
  - Taking corrective actions incase the project's actual performance conflicts with its goals.
  - Making adjustments to bring the actual process performance inline with the defined acceptable limits, as appropriate.

SEPG (Software Engineering Process Group):

- SEPG is mainly responsible for:
  - Comparing the mean of the projects' actual measurements with the expected values set for the organization.
  - If the organization goals are not met then actions are taken to resolve the conflict.
  - Adjustments are made in the measurement plan defined for the organization to bring the actual process performance inline with the defined acceptable limits, as appropriate
  - Providing the projects with analysis reports to document the results of the Analysis and to suggest corrective actions to bring the project's performance inline with the goals, in case any deviation occurs.

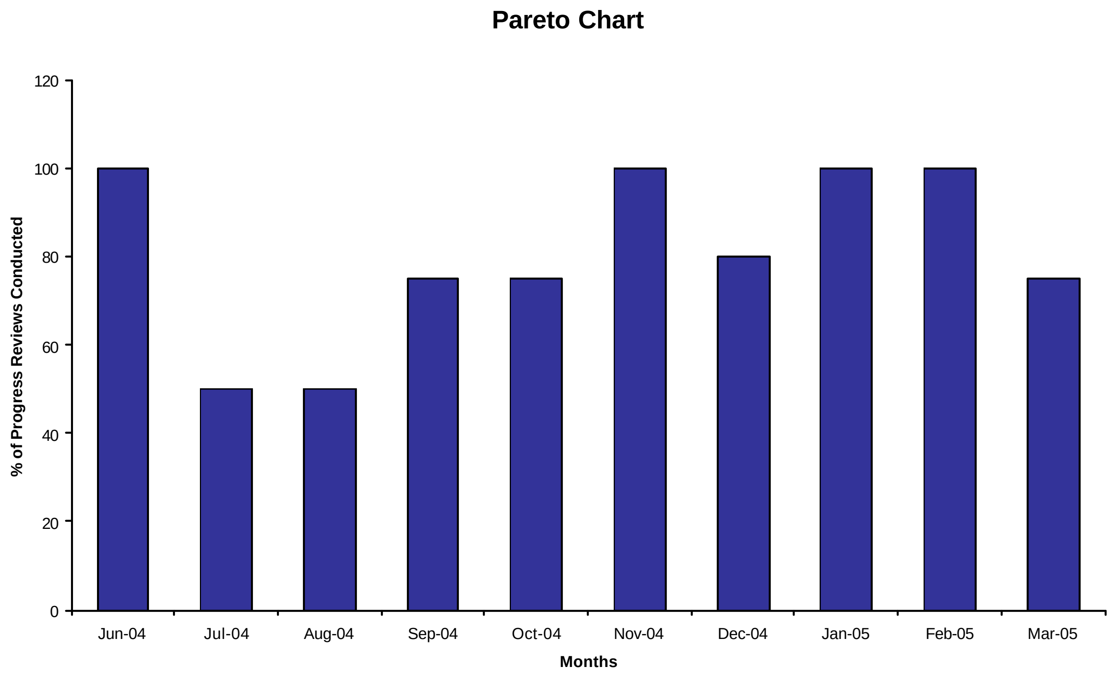

Figure 3: Pareto Chart

3.5.3 Types of Analysis

There are two types of data analysis:

- Comparison to Specifications and
- Comparison to Averages.

Specifications are targets, plans or goals set for a project. For example there is a goal for a project that Delivery of the project must be on time with 10% variance in the schedule.
This is the project goal and ± 10% are the acceptable limits of the goal. Average is the mean value of a specific metric for an Organization.

By comparing the actual value of a metric, for a specific project, with its average value, within the organization, we can determine the performance of that project relative to the other projects within the organization.

3.5.4 Statistical Process Control

The use of statistical tools and techniques to analyze a process or its outputs in order to control, manage, and improve the quality of the output or the capability of the process.

3.5.5 SPC Tools

There are 7 SPC basic tools used to analyze the process behavior:

- Scatter Diagram
- Run Charts
- Cause & Effect Diagram
- Histogram
- Bar Charts
- Pareto Charts
- Control Charts

Among all of the above, Control Charts are the most useful such that they let you know what your processes can do so that you can set achievable goals. But this depends upon the organization whether it is ready to spend money on people to train them on the use of such tools. In the following section we shall discuss the use of some of the above-mentioned SPC tools.

3.5.6 Data Analysis using SPC Tools

Data Analysis involves understanding the current state of the process and predicting the future process performance so that we can take corrective actions to correct the problem at hand and control the process by taking preventive actions respectively.

Following are some of the metrics, identified in this report, for which graphs are plotted using different SPC tools. Sample data has been used to plot the graphs.

3.5.6.1 Percentage of Progress Reviews conducted

A Pareto Chart is Special form of histogram or bar chart. It is different from a typical Bar Chart such that it is a bar chart in descending order. It helps project Managers to better investigate into the problem with the help of the ranking and helps the SEPG to set priorities in Software Process Improvement.

The Pareto chart shown in Figure 5 is showing the % of Progress Reviews conducted within a project team from Jan-4 till Mar-05 in descending order. Looking at the above graph, the project manager can analyze whether the progress reviews are being conducted as planned or not and can predict the future performance, i.e., if the performance remains as such then what will be the future performance. This will help him / her to take better decisions. The chart shown in Figure 5 can be drawn weekly / monthly depending upon the size of the project and its criticality.

3.5.6.2 Work Product Status

Another metric that we have identified in our report is the 'Work Product Status' that explains the planned vs. the actual number of work products to be evaluated. The Graph selected for this metric is the bar chart. Since this metric belongs to the PPQA in our report, the SQA manager can use this graph to see the planned and actual number of work products to be evaluated by the SQAE at a certain point in time in a certain project.

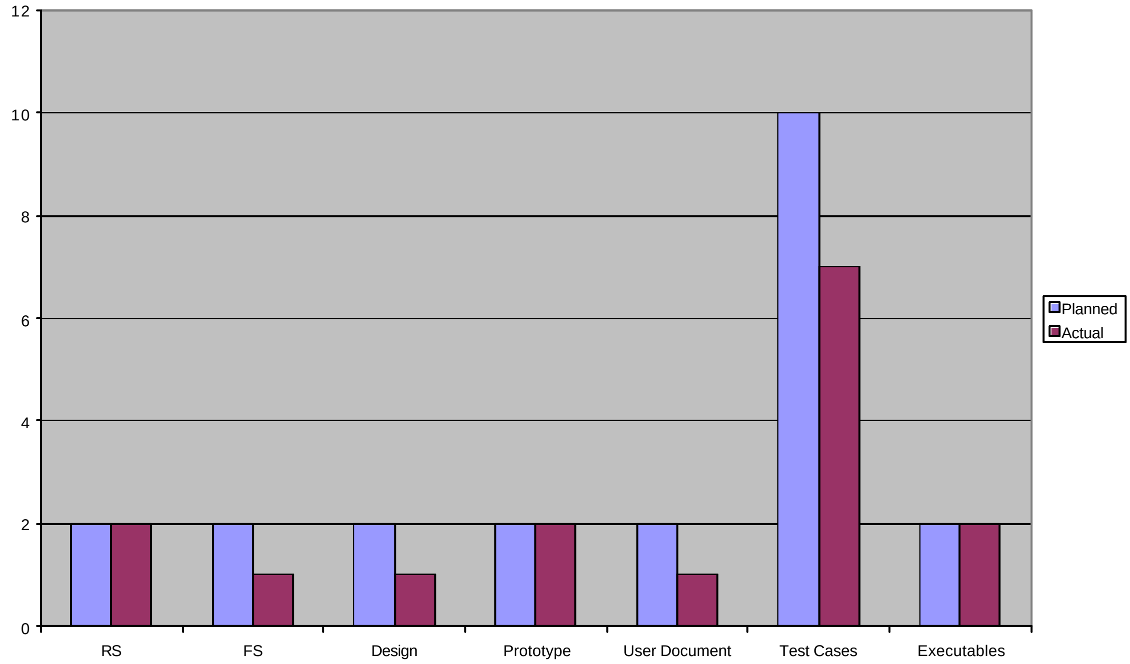

Figure 4: Bar Chart

It may help him / her to take corrective actions in that project or preventive actions in future projects to ensure the work products are evaluated as planned.

Such kind of analysis can be performed on monthly basis or at the completion of a milestone.

3.5.6.3 Work Product Consistency Ratio

The third and final type of chart is the Control Chart and is plotted for the 'Work Products Consistency Ratio' metric.

Control charts are advanced form of run charts. Control charts let you know what your processes can do, so that you can set achievable goals. There are different types of control charts that can be plotted for either the attribute or the variable data. 'Attribute' data has discrete information such as # of defects found etc. whereas 'Variable' data is continuous such as Cost of Rework and Effort Expended etc.

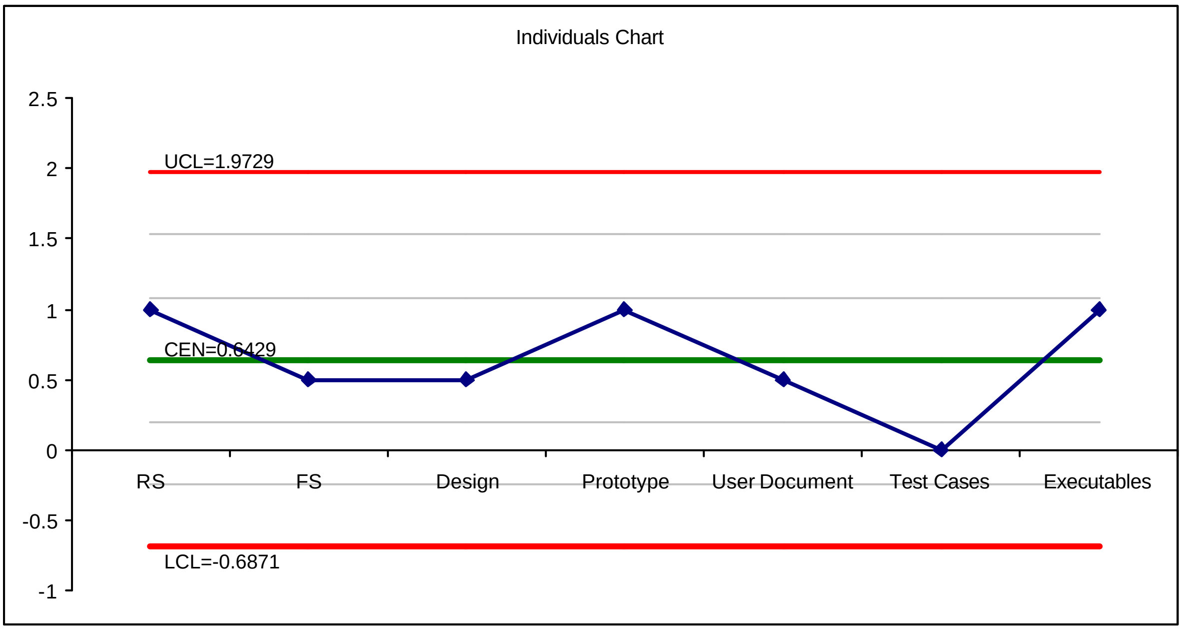

Figure 5: Control Chart

Figure 5 is the Individuals Chart plotted against the variable data of the above-mentioned metric. Individual data values are plotted. The UCL is the 'Upper Control Limit' and the LCL is the 'Lower Control Limit' that is automatically calculated from the available data. CEN is the 'Center Line' that is also auto-calculated through the behavior of the plotted data. We can also specify UCL and LCL (if there are any in the project) for this metric. The interesting thing about a control chart is that by looking at the current situation, you can predict the behavior of the process and can take steps to prevent the problem. And this is an advantage that you can get in a control chart over the other charts.

3.6 Reports

We have come up with four types of reports: Product/Project Measurement Report, Process Measurement Report, Process Results Need Activities Worksheet, Project/Product Results Need Activities Worksheet, and Corrective and Preventive Actions Reports.

3.6.1 Product/Project Measurement Report

This report is specifically designed to report issues identified after measuring a project/product performance. Report is divided into two parts. First part relates to general information related to a project/product. And second part covers all issues identified after the measurement.

Product/Project Measurement Report						
Project Name:				Project Status:		
Project Manager:				Project Starting Date:		
Team Lead:				Project Closing Date:		
Team Members:		Name	Role	Comments		
	1					
	2					
Improvement Margin	**No.**	**Issue**	**Solution**	**Required Resources**	**Start Date**	**Close Date**
%	1					
%	2					
%	3					
%	4					
%	5					
%	6					
%	7					
%	8					

Form 3: Product/Project Measurement Report

Process Measurement Report					
Process Name:				Effected Groups:	
Process Manager:					
Team Lead:					
Team Members:		Name	Role	Comments	
	1				
	2				
	3				
	4				
	5				
Improvement Margin	No.	Issue	Solution	Resource Assigned	Roles
%	1				
%	2				
%	3				
%	4				
%	5				
%	6				
%	7				
%	8				

Form 4: Process Measurement Report

Product/Project Results, Need, Activities Worksheet						
Project	Result	Impact	Urgency	Needs	Activities	Frequency
A						
B						

Form 6: Product/Project Results, Need, Activities Worksheet

Process Results, Need, Activities Worksheet						
Process	Result/Score	Impact	Urgency	Needs	Activities	Frequency
A						
B						

Form 5: Process Results, Need, Activities Worksheet

Corrective and Preventive Report (Process/Product)	
What is great performance for a project/process after measurement analysis?	
Lesson 1	
Lesson 2	
Lesson 3	
What we want to achieve and What we can do after analysis report?	
Wants	Capability
Lesson 1	
Lesson 2	
Lesson 3	
What should be avoided for future?	
Lesson 1	
Lesson 2	
Lesson 3	

Form 5: Corrective and Preventive Report (Process/Product)

Every issue will be described in terms of a solution, required resources and with time limit constraint to resolve the issue. There is another data field, improvement margin, which tells the overall impact of improvement in project/product.

3.6.2 Process Measurement Report

This report is specifically designed to report issues identified after measuring a process performance. Report is divided into two parts.

First part relates to general information related to a process. And second part covers all issues identified after the measurement.

Every issue will be described in terms of a solution, required resources and with time limit constraint to resolve the issue. There is another data field, improvement margin, which tells the overall impact of improvement in process.

3.6.3 Process Results, Need, Activities Worksheet

This report is specifically designed to report the results or score gained after the measurement activity.

Scores basically depicts the condition or overall performance status of any target process.

The impact is also described to indicate the importance of the process in overall scenario to show the ripple effects of improvement on other processes or projects.

Again urgency indicates the importance of the process, and immediate improvement requirement. What is required to improve will come under the needs. Required activities will also be given to get a task set with required frequency e.g. daily, quarterly, annually.

The report template is given in the Form 5.

3.6.4 Product/Project Results, Need, Activities Worksheet

This report is specifically designed to report the results or score gained after the measurement activity.

Scores basically depicts the condition or overall performance status of any target on going projects. Normally projects are of short duration so it is better if we say that the report is for the products. The impact is also described to indicate the importance of the project in overall scenario to show the ripple effects of improvement on other processes or projects.

Again urgency indicates the importance of the project/product, and immediate improvement requirement. What is required to improve will come under the needs. Required activities will also be given to get a task set with required frequency e.g. daily, quarterly, annually.

The difference between a project and process report is their impact on effected groups and their areas of concerns. Idealistically project covers all processes, so on going projects measures can be done in quarterly or bi-yearly audits. And process on the other hand has limited vision as compared to project.

The report template is given in the Form 6.

3.6.5 Corrective and Preventive Report (Process/Product)

This report is generically designed for processes and projects/products. This report has three parts: Good actions, Corrective actions and preventive actions.

In good actions part, we capture what are good lessons to learn and repeat. In corrective actions part, we capture what we want and what we can do to improve a project or process. And in the last part, Prevent actions part, we capture what should be avoided in future to improve the performance of a project or process.

4 Conclusion

We have undertaken this metric program development to measure the CMMI level II process areas. We have developed the metric plan for two process areas: ‘Software Project

Monitoring and Control' and 'Process and Product Quality Assurance.

Metrics plan highlights documentation of software process which is already documented in CMMI, stating the goals for processes, defining metrics for the goals, identifying the data to collect, defining data collection procedures, assembling the tool for metrics, creating metric repository and developing a feedback mechanism.

The metric plan has procedures and templates for data collection, quantitatively analyzing the collected data, which is the first step to improve it and lastly reporting. This metrics plan can address the goals, the abilities and activities required in CMMI level II process areas in terms of improvement sufficiently and effectively.

The metrics plan has all types of metrics, simple and complex. As we have tried to design a generic plan for all kind of projects and organizations (interested in process improvement with regards to CMMI), so we can divide this plan in multiple ways depending on the requirements.

This plan will help to focus on those areas which are more problematic, and results in higher quality and productivity. By capturing the metrics developed in this plan above more energy consumer processes can be identified in current software development process. In that way, changes can be made to the process and hopefully improve it. These changes may result in new metrics being gathered or show changes in trends in the old ones. Either way, changes in process results in changes in metrics which results in changes in process etc.

Process improvement requires measurement, with out measuring, you can not improve. Improvement is a time consuming process, it will not occur in one night shot. We will have to invest time, cost, and resources and effort from all levels. Moreover, we will have to be patient. "The metrics require improvement from time to time. All these require us to be understanding and cooperative to make this happen. It is not easy, but definitely worthy of pursuing." [1]

5 Future Work

We have developed a metric program and plan to measure the CMMI level II process areas but we have done it only for two process areas. Some of work is already done on Requirement Management and Software Project Planning, but rest of process areas of level II and other levels still need deliberation and there is sufficient margin of work.

The CMMI structure has many mandatory and optional components: Process Areas, Specific Goals, Generic Goals, Specific Practice, Generic Practices, Sub Practices and Work Products. We covered all components except work products. In fact work product should be incorporated and related measures should be defined.

There are two types of goals in CMMI related to any process area: specific goals and generic goals. In this metric plan we have focused only on the specific practices and we have not covered the generic goals. Further work needs to be done for generic goals as well.

CMMI has two representations: Staged and Continuous. In this research work, we have only covered staged representation as there two types of practices involved in continuous representation: basic and advance, so might have covered basic practices but our work lacks in advance practices.

6 References

[1] Terence L. Woodings, Gary A. Bundell, *"A Framework for Software Project Metrics"*, 12th ESCOM Conference on Software Control and Metrics, 2001

[2] Alexander L. Wolf, David S. Rosenblum, *"Study in Software Process Data Capture and Analysis"*, Proceedings of the Second International Conference on the Software Process, 1993

[3] Philip M. Johnson, Hongbing Kou, Joy Agustin, Christopher Chan, Carleton Moore, Jitender Miglani, Shenyan Zhen, William E.J. Doane, *"Beyond the Personal Software Process: Metrics collection and analysis for the differently disciplined"*,

25th International Conference on Software Engineering, 2003

[4] Paul Wernick, "*Identifying and Justifying Metrics for Software Evolution Investigations Using the Goal-Question-Metric Method*", 2000

[5] Eduard Metzker, "*An Experimental Process Metrics Support Environment and a Cross-Organizational Study on its Acceptance by Practitioners*", Ninth International Software Metrics Symposium, 2003

[6] Marcello Visconti, Liliana Guzmán, "*A Measurement-Based Approach For Implanting SQA & SCM Practices*", Proceedings Of The XX International Conference Of The Chilean Computer Science Society, 2000

[7] *"CMMI Online"* by Tera Quest, www.teraquest.com/CMMI/CMMI Staged MainPage.html

[8] Christiane Differding, Barbara Hoisl Christopher M. Lott, "Technology Package for the Goal Question Metric Paradigm", *AG Software Engineering, Department of Computer Science, Fraunhofer Institute for Experimental Software Engineering, University of Kaiserslautern Sauerwiesen*, Germany, 1996

[9] *"Creating a Metrics Program*", Software Productivity Center Inc, 2000, http://spc.ca/resources/metrics

[10] *"Software Metrics Collection"*, *http://www.abo.fi/~atorn/SQuality/SQ.html*

7 Bibliography

[1] Dale Couprie , John Frankovich , Bin Li, "Software Metrics Plan For The Project Planning Key Process Area Of The Capability Maturity Model for The SENG 623 Company", University of CALGARY, 1997

[2] Alan D. Goodbrand, Qian Wang, "Software Measurement Plan for the Requirements Management Key Process Area of the Capability Maturity Model for SENG623 Inc", University of CALGARY, 1997

[3] Kim Caputo, "CMM Implementation Guide", Publisher Addison Wesley, 3rd edition, 1999

[4] Caper Jones, "Software Quality Analysis and Guidelines for Success", International Thomson Computer Press, 1997

[5] Watts S. Humphrey, "Managing the Software Process", Thomson Press, 2002

8 Acknowledgement

We would like to thank Mr. Waqqas Zyad, our course coordinator, for helping us throughout to carry out the work. Further, we would like to thank Mr. Talha Javed for his guide that made us able to start with the work and take steps further. We are also indebted to NETSOL for providing us with environment in which we can continuously learn..