Process and Project Metrics

Measurement enables us to gain insight into the process and the project by providing a mechanism for objective evaluation. Measurement can be applied to the software process with the intent of improving it on a continuous basis. Measurement can be used throughout a software project to assist in estimation, quality control, productivity assessment, and project control.

What is it?

Software process and project metrics are quantitative measures that enable you to gain insight into the efficacy of the software process and the projects that are conducted using the process as a framework. Basic quality and productivity data are collected. These data are then analyzed, compared against past averages, and assessed to determine whether quality and productivity improvements have occurred.

Metrics are also used to pinpoint problem areas so that remedies can be developed and the software process can be improved.

Who does it?

Software metrics are analyzed and assessed by software managers. Measures are often collected by software engineers.

Why is it important?

If you don’t measure, judgment can be based only on subjective evaluation. With measurement, trends (either good or bad) can be spotted, better estimates can be made, and true improvement can be accomplished over time.

What are the steps?

Begin by defining a limited set of process, project, and product measures that are easy to collect. These measures are often normalized using either size- or function oriented metrics. The result is analyzed and compared to past averages for similar

Reference:- Roger S Pressman - Software engineering _ a practitioner's approach-McGraw-Hill Higher Education (2010)

projects performed within the organization. Trends are assessed and conclusions are generated.

What is the work product?

A set of software metrics that provide insight into the process and understanding of the project.

How do I ensure that I've done it right?

By applying a consistent, yet simple measurement scheme that is never to be used to assess, reward, or punish individual performance.

We measure:

(1) To characterize in an effort to gain an understanding "of processes, products, resources, and environments, and to establish baselines for comparisons with future assessments";

(2) To evaluate "to determine status with respect to plans";

(3) To predict by "gaining understandings of relationships among processes and products and building models of these relationships";and

(4) To improve by "identify roadblocks, root causes, inefficiencies, and other opportunities for improving product quality and process performance.

25.1 METRICS IN THE PROCESS AND PROJECT DOMAINS

Project metric intention is to provide a set of process indicators that lead to long-term software process improvement. Project metrics enable a software project manager to

(1) Assess the status of an ongoing project,

(2) track potential risks,

(3) uncover problem areas before they go "critical,"

(4) Adjust work flow or tasks, and

(5) Evaluate the project team's ability to control quality of software work products.

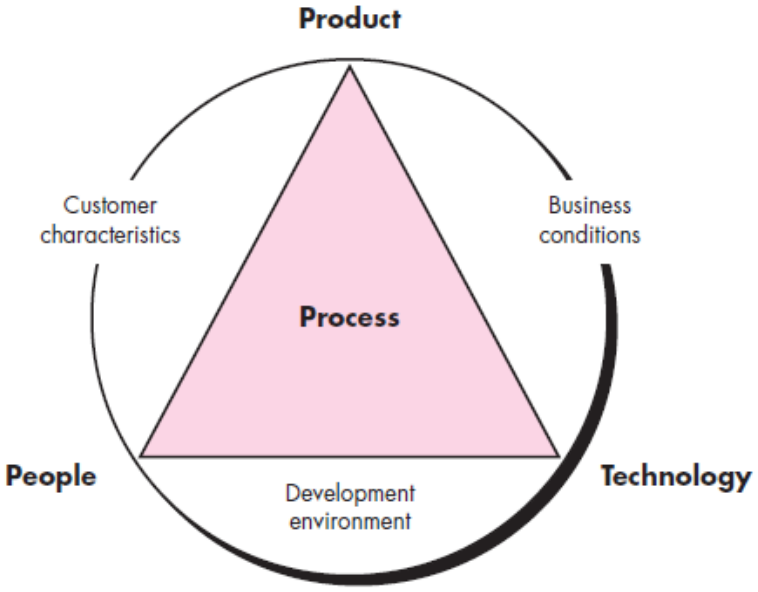
25.1.1 Process Metrics and Software Process Improvement

Reference:- Roger S Pressman - Software engineering _ a practitioner's approach-McGraw-Hill Higher Education (2010)

To improve any process

- measure specific attributes of the process,
- develop a set of meaningful metrics based on these attributes, and
- then use the metrics to provide indicators

FIGURE 25.1
Determinants for software quality and organizational effectiveness.
Source: Adapted from [Pau94].



Process is affected by 3 Ps. Skill and motivation of people affect quality and performance. Complexity of product affect team performance. Technology also has an impact.

Circle shows environmental conditions that include the development environment (e.g., integrated software tools), business conditions (e.g., deadlines, business rules), and customer characteristics (e.g., ease of communication and collaboration).

You derive a set of metrics based on the outcomes that can be derived from the process. Outcomes include measures of errors uncovered before release of the software, defects delivered to and reported by end users, work products delivered (productivity), human effort expended, calendar time expended, schedule conformance, and other measures.

Private and public uses for software metrics

Some process metrics are private to the software project team but public to all team members. Examples include defects reported for major software functions (that have been developed by a number of practitioners), errors found during technical reviews, and lines of code or function points per component or function. The team reviews these data to uncover indicators that can improve team performance.

Public metrics generally integrate information that originally was private to individuals and teams. Project-level defect rates, effort, calendar times, and related data are collected and evaluated in an attempt to uncover indicators that can improve organizational process performance.

Reference:- Roger S Pressman - Software engineering _ a practitioner's approach-McGraw-Hill Higher Education (2010)

Process metrics can be misused, creating more problems than they solve. Grady [Gra92] suggests a "software metrics etiquette" that is appropriate for both managers and practitioners as they institute a process metrics program:

• Use common sense and organizational sensitivity when interpreting metrics data.

• Provide regular feedback to the individuals and teams who collect measures and metrics.

• Don't use metrics to appraise individuals.

• Work with practitioners and teams to set clear goals and metrics that will be used to achieve them.

• Never use metrics to threaten individuals or teams.

• Metrics data that indicate a problem area should not be considered "negative." These data are merely an indicator for process improvement.

• Don't obsess on a single metric to the exclusion of other important metrics.

Reference:- Roger S Pressman - Software engineering _ a practitioner's approach-McGraw-Hill Higher Education (2010)