UniQuE

Metrics Handbook

(Engagement Name and Id)

(Client)

Document History

Version	Date	Author	Changes

Review And Approval

Company	Role	Name	Date	Signature

Distribution

Company	Name	Number	Media	Action

Storage

Location	Access	Administrator

Company Confidential - Copyright © 2022 Capgemini - All rights reserved

Template Version Number: : Group Reference v1.0

Table Of Contents

1. Why Measurement

Purpose

This section clarifies the role of a software measurement program in support of software development and maintenance activities. It provides sound motivation to initiate or expand analysis of data and application of results. There are three key reasons for software measurement:

- Understand and model software/ Mechanical engineering processes and products
- Aid in the management of software/ Mechanical projects
- Guide improvements in software/ Mechanical engineering processes

The underlying purpose of metrics program is to achieve specific results from the *use* and *application* of the measures; collecting data is *not* the objective.

Measurement to Increase Understanding

The most important reason for establishing a measurement program is to evolve an understanding of software and the software engineering processes in order to derive models of those processes and examine relationships among the process parameters. Increased understanding leads to better management of software projects and improvements in the software engineering process.

General questions to be addressed to increase the understanding might include the following:

- How much are we spending on software development?
- Where do we allocate and use resources throughout the life cycle?
- How much effort do we expend specifically on testing software?
- What types of errors and changes are typical on our projects?

Measurement for Managing Software

The second key reason for establishing an effective measurement program is to provide improved management information. Having an understanding of the software environment based on models of the process and on relationships among the process and product parameters allows for better prediction of process results and more awareness of deviations from expected results.

Thus, understanding the software engineering process leads to better management decision-making. The understanding comes from analysing local data; without analysis, any data collection activity is a waste of effort.

The next step is to use the understanding that comes from the engineering models to plan and manage software project activities.

Specifically, the knowledge gained about the software engineering process will be used to:

- *Estimate* project elements such as cost, schedules, and staffing profiles.
- *Track* project results against planning estimates.
- *Validate* the organisational models as the basis for improving future estimates.

Measurement for Guiding Improvement

The primary focus of any software-engineering organisation is to produce a high-quality product within schedule and budget. However, a constant goal, if the organisation is to evolve and grow, must be continual improvement in the quality of its products and services. *Product* improvement is typically achieved by improving the processes used to develop the product. *Process* improvement, which requires introducing change(s), may be accomplished by modifying management or technical processes or by adopting new technologies. Adoption of a new technology may require changing an existing process. In any case, software measurement is a key part of any process improvement program; knowing the quality of the product developed using both the initial and the changed process is necessary to confirm that improvement has occurred.

2. Integrated Software Metrics Framework

Software metrics has application across all the stages of software development lifecycle at multiple levels in the organization. All major processes in the QMS have either a direct or an indirect interaction with the measurement process. This leads to a requirement for developing a well-defined framework for software metrics, integrated with the other process elements that can facilitate clear understanding of metrics application at various levels.

A Case for Metrics Framework

1. Challenge of Complexity

Measurement Program at Capgemini has following commonly visible components:

1. Engineering Goals (Metrics Targets)
2. Standard Metrics List (along with Core Metrics)
3. Metrics Database
4. Metrics Baselines
5. Process Models
6. Supporting Tools, Templates, Records & Reports

These components are related to each other and external components through a defined set of processes / procedures that can be visualized as a protocol for interaction between two entities. For example, Project Plan is a visible component of Project Management Procedure (SDP – Project Management) that interacts with the component-1 (Engineering Goals) for defining goals at project level, with the component-2 (Standard Metrics List) for selecting metrics that needs to be tracked at process and sub-process level other than Engineering Quality Goals, with component-3, 4 & 5 for estimation, metrics limit setting and so on. This example gives a fair idea that if we try to trace all the internal and external relations of the components of a measurement program, we will land in a far too complex network/relationship diagram beyond practical utility.

2. Derailing of a Measurement Program

Owing to this complexity, using a measurement program intelligently (taking an informed decision about what metrics to track, when and what preventive/corrective action to take and from where to get additional objective information for decision making like agreeing on a particular effort figure for a given task) rather than passively (passively means following what majority was doing) becomes a tough call. This result in a measurement program losing its credibility among the stakeholders despite of their commitment and a good deal of effort and turns a measurement program into a ritual.

3. Search of a Solution

Solution is Pareto Principle: 80% of increase in the effectiveness of a measurement program can be achieved by clearly understanding and documenting that vital 20% of interactions between these internal components of measurement program and external component that are the backbone (framework) of the whole network. This is how exactly a metrics framework helps in making measurement program more effective.

Integrated Software Metrics Framework

1. What is Integrated Software Metrics Framework?

It is certainly not a procedure detailing every step of the measurement program.

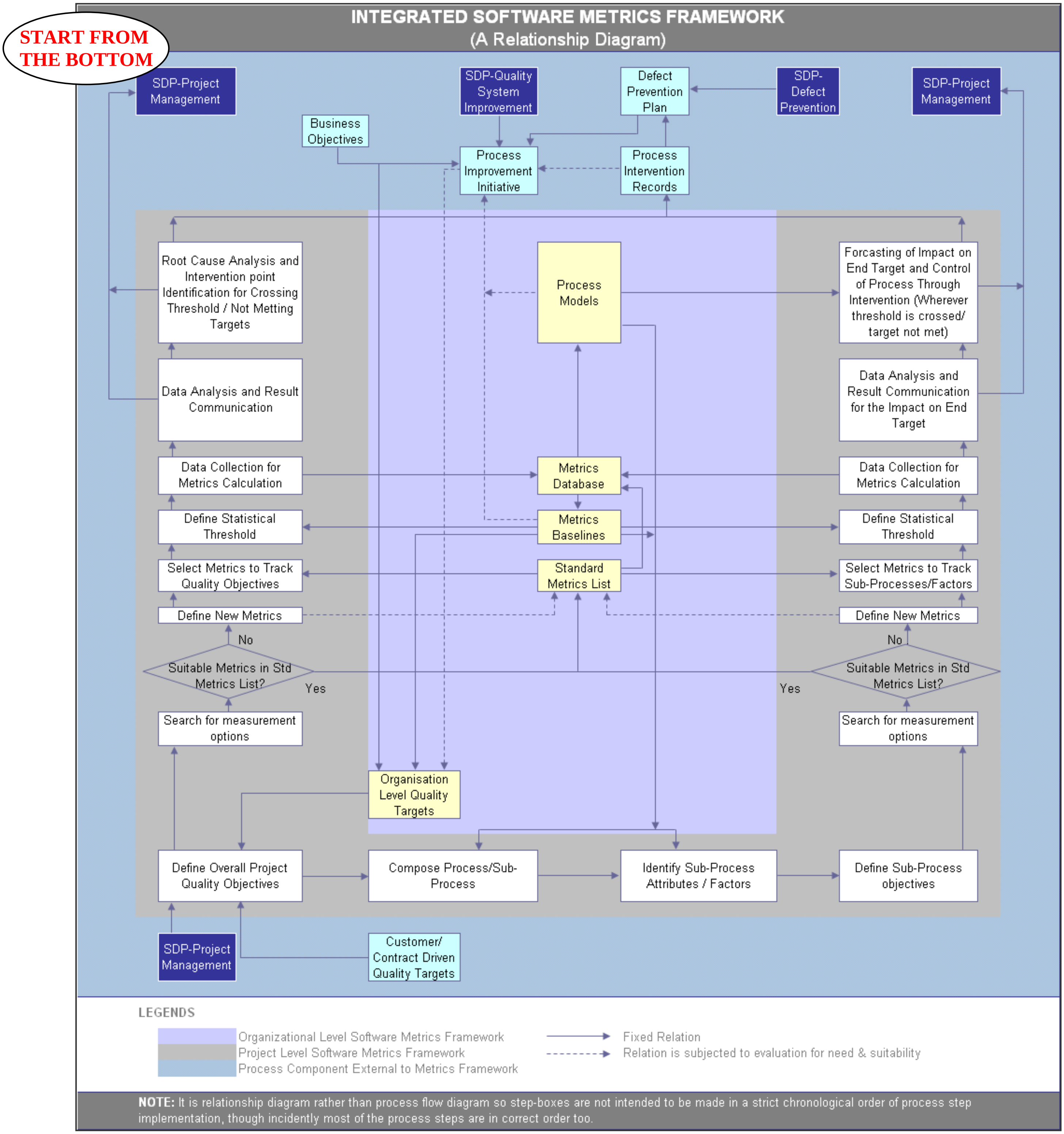
It is not a comprehensive documentation of all the possible interactions of all the interrelated components in all possible scenarios.

It is not intended to be a process flow chart although it may cover all the important process steps in logical order.

It is a framework presented in a form of a relationship diagram that covers all the important measurement program components, external components, important internal and external process steps to a measurement program described at a more abstract level with main interrelations (shown by arrows).So micro level details may be found in relevant SDP(s) and SEPG Guidelines.

2. Integrated Software Metrics Framework at Capgemini

Next page shows integrated metrics framework at Capgemini.


START FROM THE BOTTOM
INTEGRATED SOFTWARE METRICS FRAMEWORK
(A Relationship Diagram)
SDP-Project Management
SDP-Quality System Improvement
Defect Prevention Plan
SDP-Defect Prevention
SDP-Project Management
Business Objectives
Process Improvement Initiative
Process Intervention Records
Root Cause Analysis and Intervention point Identification for Crossing Threshold / Not Metting Targets
Process Models
Forcasting of Impact on End Target and Control of Process Through Intervention (Wherever threshold is crossed/ target not met)
Data Analysis and Result Communication
Data Analysis and Result Communication for the Impact on End Target
Data Collection for Metrics Calculation
Metrics Database
Data Collection for Metrics Calculation
Define Statistical Threshold
Metrics Baselines
Define Statistical Threshold
Select Metrics to Track Quality Objectives
Standard Metrics List
Select Metrics to Track Sub-Processes/Factors
Define New Metrics
Define New Metrics
No
No
Suitable Metrics in Std Metrics List?
Yes
Yes
Suitable Metrics in Std Metrics List?
Search for measurement options
Search for measurement options
Organisation Level Quality Targets
Define Overall Project Quality Objectives
Compose Process/Sub-Process
Identify Sub-Process Attributes / Factors
Define Sub-Process objectives
SDP-Project Management
Customer/ Contract Driven Quality Targets
LEGENDS
Organizational Level Software Metrics Framework
Project Level Software Metrics Framework
Process Component External to Metrics Framework
Fixed Relation
Relation is subjected to evaluation for need & suitability
NOTE: It is relationship diagram rather than process flow diagram so step-boxes are not intended to be made in a strict chronological order of process step implementation, though incidently most of the process steps are in correct order too.

3. Interpreting Integrated Software Metrics Framework

1. During project planning phase, overall quality objectives of the project shall be identified on the basis of organization level quality targets and customer specified targets.
2. After setting overall quality objectives, following two activities start:
    a. Search and identify metrics for tracking these quality objectives
    b. Composing sub processes (selecting sub-processes relevant to the project) to meet these overall objectives.
3. Once sub processes are composed, sub process objectives are defined and metrics are searched to track these objectives.
4. From here onward, whether metrics is for process or for tracking sub-process, flow for next few steps remains same. Statistical limits are identified and metrics data is collected to see if process/ sub-process output is under control limits or not and same is communicated by the means of QMS defined reporting mechanisms in relevant forums.
5. If limits are crossed, causal analysis is initiated and corrective action (intervention point) needs to be identified for both process and sub-process level control.
6. These process intervention points need to be recorded and collated at organization level to help in defect prevention planning.
7. If we go back to composing sub-processes, metrics baseline report and process model help in deciding process composition.
8. Similarly Standard Metrics List at organisation level helps in identification of metrics at project level and if some new metrics are proposed at project level, they are considered for being part of the standard metrics list. (Refer SDP – Metrics Collection & Analysis)
9. While metrics baseline report helps in defining statistical limits, process model help in predicting the impact of a factor going out of control at sub process level on the end results or process level targets.
10. Metrics baselines at organization level that are drawn on the basis of data from projects becomes input for setting engineering targets that are implemented back at project level.

NOTE1: Integrated Software Metrics Framework (diagram) should be read in bottom up direction.

NOTE2: Not all the paths given in the diagram are covered in this section, but they can also be interpreted in the same way by going with the arrow direction to see which component or process step is giving input (tail end of arrows) and which one is receiving it (Head of arrows).

3.Metrics Definition

Metrics definition is a general term used that includes following:

1. Metrics Identification through:
    a. Goals, Process, Sub-process, Metrics mapping
    b. New metrics from projects and other sources
2. Metrics definition:
    a. Mathematical Definition
    b. Definition of primary measurements and their units
    c. Identification of data sources
    d. Elaboration of metrics use considerations
3. Standard Metrics List Maintenance
4. Identification of Core Metrics from Standard Metrics list to be Tracked at Organization Level

Process for Metrics Definition

Based on business objectives, set of policies, processes, procedures and guidelines have been identified in the form of Quality Management System.

In order to track process performance to mean business objectives, appropriate metrics shall be identified, defined and implemented.

In order to manage overall process performance and meet end goals (business Objectives), critical sub-processes shall be identified and metrics for tracking output for these sub-processes shall be identified, defined and implemented.

Project level management for meeting goals shall also be done through metrics at process and sub-process/ factor level. For details about the process for identifying goals and metrics at project and organization level, please refer to the SDP – Metrics Collection & Analysis.

A standard metrics list shall be maintained at organization level that projects can refer to identify metrics for the process/ sub-process/ factor management at project level.

The metrics in this metrics list are prioritized at organizational level based on their relative influence (correlation) on the business objectives. This mapping is periodically visited along with the metrics baselines. Projects refer to this order of priority for the metrics in their project plans unless there is a contractual, customer/ project specific requirement, risk or constraint that requires a different order of priority at the project level.

Whenever some new metrics are defined at project level or defined due to some new business objective at organization level, these shall be included in standard metrics list subjected to the successful evaluation through piloting or by any other means followed by an approval of SEPG.

Process to Reach the Goals and Objectives

The overall process to reach the goals and objectives is done through a 5 step process as defined below:

1. **VISION STATEMENTS**: Understanding the organization's vision and deriving quantified objectives from Vision
2. **BARRIERS TO VISION**: Identifying barriers to the Vision statements to focus on
3. **BUSINESS GOALS**: Formulate business goals
4. **QUALITY & PROCESS PERFORMANCE OBJECTIVE**: Derive Quality & Process performance objectives using goal decomposition matrix
5. **PROCESS PERFORMANCE MEASURES**: Set quantified goals for Process Performance objectives.

STEP 1. VISION STATEMENTS

"Capgemini exists to solve our clients' most important engineering challenges with innovative solutions that connect the world, improve lives, and build a better future".

Translating this vision in to specific measurable objectives,

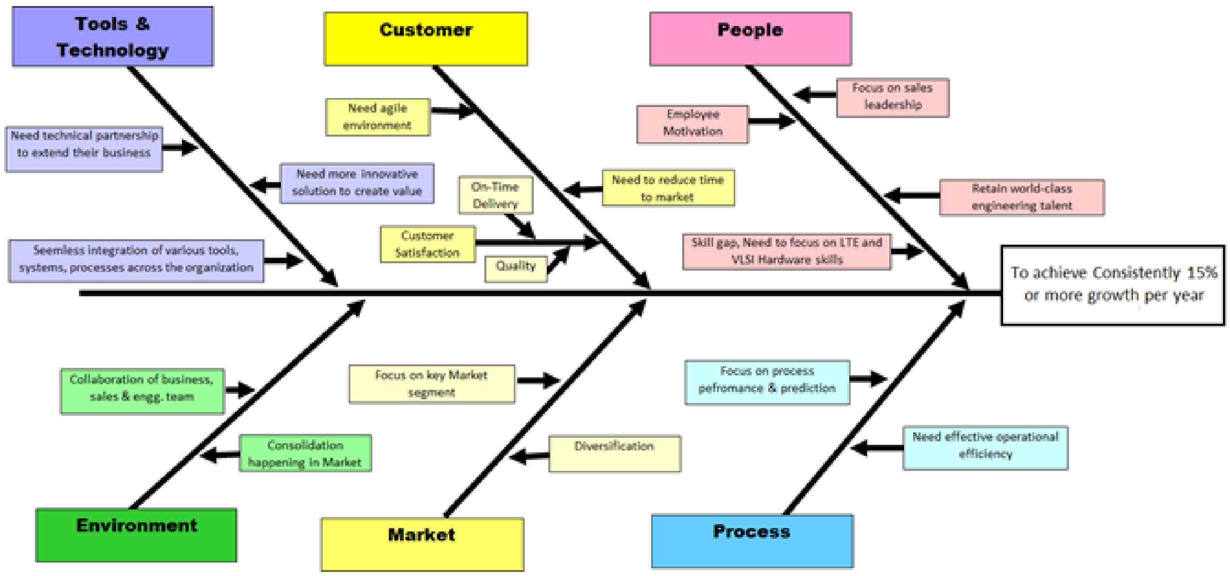
BUSINESS OBJECTIVE

High-Value and Innovative Business focused on providing Product Engineering Services and Software to clients across a diverse set of industries where connectivity, silicon chips, and embedded software create value due to the emerging and global power of the Internet of Things.

The business objective is to consistently grow at mid-to-high teens with high profitability, due to the unique value and quality delivered to clients by the exceptional Capgemini people.

STEP 2. BARRIERS TO VISION

The critical factors which can influence the barriers to vision is categorized under people, customer, process, market, technology and is identified using cause & effect (Ishikawa) diagram as defined,



Using traditional SWOT to identify barriers and opportunities. Please refer SWOT given below:

SWOT Analysis	Strengths	Weaknesses
Opportunities	• Strong legacy in Communication, recognized as the leader in telecommunication industry • With Smartplay acquisition, we are now the market leader in Semiconductor industry, and are now able to offer end to end PES solutions , from Chip to Cloud • Defined and proven processes across Product engineering and R&D space • Enabling software portfolio - our differentiation in the market and creates huge value for clients	• Industry expertise in the new market segments like - auto, industrial etc. (lack of SME's) • Sales restructuring done this year - yet to prove success • We need more investments in the new areas
Threats	• New technologies like IoT and cloud dominating the future, leading to high demand across other industries like Software, automobile, automation, consumer electronics etc • Clients are looking for end to end capability…lots of projects won where we are providing end to end services from Chip design, hardware, and software. • With blurring boundaries between technology areas, new opportunities are arising - for semiconductor skills in embedded / product companies; communication skills in Internet companies.	• Reducing spend in R&D across the telecommunication industry • Consolidation in both telecom and semiconductor industries • Attrition

STEP 3. BUSINESS GOALS

The business goals are derived considering the SWOT. We shall continue to capitalize our areas of strengths to leverage opportunities. We shall focus on setting goals for improvement as part of business objectives to address the weakness which are the barriers for achieving the vision.

The key business objectives for Capgemini are:

- Capgemini will maintain Customer Satisfaction Rating (OTACE) of 4 or more *(scale of 1- 5 where 5 is highly satisfied)* with 90% of confidence.
- Capgemini will consistently grow at mid-to-upper teens per year with high profitability *(Revenue measured as 15%+ per year & EBITDA in the upper quartile of those we compare against)*.

The achievement of business objectives depends on the critical success factors, Capgemini will focus on four critical success factors:

1. Client Value
2. Excellence in Execution (Delivery, Quality and Process)
3. Market Focus *(focus on select markets where Capgemini has deep competence and engineering strength)*
4. Talent (Our Employees)

Fulfilment of requirements for quality & process performance objectives (QPPO) aligned to Business objective:

Business Objective	**Critical Success Factors**	**Requirements for Quality & Process Performance Objectives aligned to Business objective**
EBIT	Market Focus Client Value Talent	High Customer Satisfaction
High Customer Satisfaction	Excellence in Execution Client Value	On-Time Delivery Quality of Deliverables Increase efficiency of Quality Control Process and Enhanced Visibility and Accountability

STEP 4. QUALITY & PROCESS PERFORMANCE OBJECTIVE:

Based on the identification of key process & sub process which can influence the business goals, the metrics for the project outcome (both overall and process/sub-process level) for better product quality and timely delivery are identified in below attached sheets.

Business Objectives, Processes, Sub-processes and Metrics Mapping

Following sheet captures the mapping of business objectives, processes, sub-processes and metrics mapping.

- DEV- Business obj- Process metrics
- SVC- Business obj- Process metrics
- Testing- Business obj-Process metrics

The quantified goals for the QPPO is determined based on the current process performance baseline and the improvement goals set by SEPG. Projects can align or have more aggressive goals based on project objectives set. For more details, please refer to the Chapter 5 Metrics Baseline.

STEP 5. PROCESS PERFORMANCE MEASURES

Process Performance Analysis: Please refer to the Chapter 6 & 7 Process Control and Process Modeling

Standard Metrics

Standard metrics set is the comprehensive list of metrics computed monthly/ defined frequency however actual metrics may vary from project to project. A subset of these metrics that is reported by all the applicable projects monthly across the organization and shared with the senior management is known as the *Core Metrics* set. A slightly bigger set of metrics that contain all the core metrics and some additional milestone/ phase end/ release end metrics is known as the *Common Metrics* set. These are generally identified as part of the project plan (quality assurance plan) and thereby the virtue of being in project plan become mandatory. Total set of metrics (common, and special: that are specific to purpose and limited in use) along with documented metrics definitions, data sources and considerations form standard metrics list.

Standard Metrics List with the Detailed Metrics Definitions:

For the standard metrics, detailed definitions along with the information about data source and metrics use considerations are provided in the following sheet

Std-Metrics-List

Core Metrics

We track following metrics at organization level on Monthly basis:

Metrics	Development	Development – Agile/SAFe model	Re-Engineering	Maintenance : Enhancements	Maintenance : Production Support	Managed Service	Testing	Hardware Development	Mechanical Development
Ontime Delivery (OTD)	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory
First Time Right (FTR)	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory
Schedule Variance (%) (Agile/SAFe : Release Burn-up)	Mandatory	Mandatory	Mandatory	Mandatory	NA	NA	Mandatory	Mandatory	Mandatory
Effort Variance (%)*	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory
Productivity (Agile/SAFe : Velocity)	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory
Defect Density (for all phases including artifacts)	Mandatory	Mandatory	Mandatory	Mandatory	NA	NA	Mandatory	Mandatory	Mandatory
Delivered Defect Rate (External Defect/KSLOC	Mandatory	Mandatory	Mandatory	Mandatory	NA	NA	NA	Mandatory	Mandatory
Defect Removal efficiency (DRE)	Mandatory	Optional	Mandatory	Optional	NA	NA	Optional	Mandatory	Mandatory
Process Compliance Percentage (%)	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory
Test Passed Percentage (Final Round)	Optional	Optional	Optional	Optional	NA	NA	Mandatory	NA	NA
Defects Acceptance Ratio	Optional	Optional	Optional	Optional	NA	NA	Mandatory	Optional	Optional
SLA Adherence (%)	NA	NA	NA	Optional	Mandatory	Mandatory	NA	NA	NA
Bad Fixes (%)	NA	NA	NA	Optional	Mandatory	Mandatory	NA	NA	NA
Content Adherence (Wt.)	NA	Mandatory	NA	NA	NA	NA	NA	NA	NA
Build Pass Ratio (%)	Optional	Mandatory	Optional	Optional	NA	NA	NA	Optional	Optional
Open Defects (Ticket Completion Rate in Maintenance : Production Support & Managed Service)	Optional	Mandatory	Optional	Optional	Optional	Mandatory	NA	Optional	Optional
Test Case creation productivity (Testing projects Only)	NA	NA	NA	NA	NA	NA	Mandatory	NA	NA
Test Case execution productivity (Testing projects Only)	NA	NA	NA	NA	NA	NA	Mandatory	NA	NA
Cost Of Quality (COQ)	Mandatory	Optional	Mandatory	Mandatory	NA	NA	Optional	Mandatory	Mandatory
CSAT	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory	Mandatory

These metrics are called core metrics. These are collected and analyzed at organization level and shared with senior management and board of directors monthly.

4. Metrics Data Collection, Analysis & Collation

Metrics Data Collection, Integrity Check & Analysis

- Metrics data collection at project level is typically done at the minimum frequency of once in a month.
- Data integrity checks shall be done before using collected data for any analysis.
- For Guidelines for Data Integrity Checks, please refer to the details provided on Data Cleaning in Chapter-5 about Metrics Baseline.
- Metrics data analysis is done regularly at the same frequency of metrics data collection or as and when required.
- For detailed process, refer SDP – Metrics Collection & Analysis.

4.1.1 Tools for Metrics Data Collection

- Resource Management Tool
- Metrics Collection Sheet/ as specified by Quality Group
- Excel based schedule tracking sheet / MS Project Reports / linked analyzer
- Excel based Test case / SPR tracking sheet / linked analyzer
- Excel based risk tracking sheet / Risk Management Tool
- Excel based inspection and testing data analysis sheet / , Score card
- Reports generated by CM system, Score card
- Process compliance sheet
- OTACE data collection and analysis sheet
- PQA data collection and analysis sheet
- Health Index Sheet
- Digital dashboard for maintenance projects
- Core Metrics collection and analysis sheet / linked analyzers
- Project Closure data collection sheet/
- Requirement Change Management Sheet

These are available from QG home page

Metrics Collation

4.1.2 Metrics Collation at Project Level

- SQA Report/ Metrics Sheet shall be used as a running template throughout the lifecycle of the project for metrics collation at project level.
- SQA Report/ Metrics Sheet generated reports shall be shared with Project Manager on monthly basis and the same shall be controlled.
- At the end of the project, final status of all the metrics tracked shall be recorded in Project Closure Report, which as a template is an extension of SQA Report/Metrics sheet. Alternatively, wherever QTrack is used available, the same shall be used for this purpose too.
- Templates for SQA Report ,Metrics sheet and Project Closure Report are available on QG Home Page.
- For detailed process, please refer to SDP – Metrics Collection & Analysis and SDP – Project Management.

4.1.3 Metrics Collation at Organization Level

- Metrics identified for monthly tracking at organisation level are reported monthly through BU Dashboard template that provides consolidated figures for core metrics at Business Unit level for EGO, BGO & other stakeholders.
- Collated metrics from all BU Dashboard is used for making organisation level collated report. Same BU Dashboard template is used for this purpose too.
- Data from all the projects at closure shall be collated at organization level in a Metrics Baseline Report. Use of Metrics collated for baseline:
  - Metrics data from metrics database shall be used to draw metrics baselines.
  - This shall provide data for creating and calibrating process models.
  - Metrics of similar kind of projects can be used as an input for estimation

5.Metrics Baseline

Metrics Baseline at organisation level serves as a statistical benchmark to define reasonable process target, compare process output from live projects and evaluate process capability.

Metrics Baseline has three stages:

A. Metrics Baseline Creation
B. Metrics Baseline Reporting / Communication
C. Metrics Baseline Application

Metrics Baseline Creation

Metrics Baseline Creation is a periodic exercise repeatable at a minimum frequency of once in a year.

Metrics Baseline Creation – Process Steps and Methodology:

1. Goal based Metrics selection for Metrics Baseline

   Refer to the Chapter 3 in Process to Reach the Goals and Objectives Section.

2. Collection of Data for the desired Metrics

   Refer to the Metrics Database Section in Chapter-4 in this document and Metrics Collection & Analysis SDP.
   Data adequacy to draw a baseline is as follows:

Data Insufficient	Less than 6 data points
Data Marginally Sufficient	6 to 15 data points
Data Sufficient	More than 15 data points

   If a particular metrics has less than 6 data points, metrics should not be dropped from the metrics baseline report but should be kept in report along with the information about the data sufficiency for reference purposes.

3. Data Pre-processing

   Data Pre-processing is preparing data for processing by Cleaning & Homogenization to obtain correct information.

   i. Data Cleaning – Data Error Correction
      - Data Error can be due to Missing Data or Inconsistent Data
      - Missing Data can either be retrieved or replaced with an Estimated Data.

- Inconsistent Data can be Physically Inconsistent or Logically Inconsistent.
- Some examples of Physically Inconsistent Data:
    - Project Start Date = 13/01/07 in MM/DD/YY
    - Same Name Written in Multiple ways: Alcatel_Lucent, Alcatel & Lucent, Lucent_Alcatel
    - Unit Mismatch:
        - 100 KSLOC of C, 100 KSLOC of Java, 100 SLOC of C
        - SV: 10.00, 10.00%
- Some examples of logically Inconsistent Data:
    - Improbable Data:
        - No. of Defects = 6.2
        - Inspection Rate = -2.4 Pages / Hour, 0 Pages / Hours
        - Productivity = 0 SLOC / Day
        - % Meeting SLA = 110%
    - Unintended Data:
        - Reused 'As it is' code used in Productivity
        - Discarded Defects counted in Defect Density
- Common measures to check/prevent inconsistency can be
- Random Check of Metrics Value
- Clear and Precise Definition of Measurement Unit
- Application of Validations in spreadsheet or database
- Common Error Checklist

ii. Data Cleaning – Outlier Identification & Removal

- Outliers: An observation that is numerically distant from rest of the sample (member of sample can be historical data or currently collected data depending upon the type analysis targeted)
- Outliers can be 'Mild' or 'Extreme' depending upon the distance in term of IQR (1.5 or 3 times) from Q1 or Q3
- Graphical Methods to Identify outlier are Histograms, Scatter Plots, Box Plots and Probability Plots
- Common Statistical test based methods are 'Grubbs' Test' and 'Median of Absolute Deviation method'
- Consider Data Transformation and other alternative before removing outliers
- Find special cause for outlier and see if that can be controlled or if that is rare.
- Outlier Removal is last resort

iii. Data Homogenization through Clustering

- Purpose of Data Clustering:
    - We want to divide data into clusters (buckets) such that –
        - Difference between the members of the same cluster is minimum
        - Difference between members of different clusters is maximum
- Need for Test: To ascertain objectively (statistically) that clusters (buckets) have achieved above two objectives or in other words, the clusters (buckets) made are significantly different.

- If difference is not significant, we can discard those clusters.
- Significance of difference between two clusters can be established through hypothesis testing for mean and variance.
- Common Tests for equality of Mean are Z Test and t Test.
- Select paired or unpaired t Test depending upon if you are testing effect of a cause on a same process or not.
- F Test is for testing variances of two samples.
- Use two tailed test values if hypothesis requires to compare equality
- Use one-tailed values if hypothesis requires to compare inequality (> or <).

4. Calculation of control limits for each cluster of each of the metrics
    - Select appropriate probability distribution to morph the distribution of actual data
        - If data is normally distributed, and theoretically possible value of metrics has no upper or lower limit (like for argument sake, Schedule Variance can theoretically have any value from minus to plus infinity), use normal distribution.
        - If data is normally distributed, and theoretically possible value of metrics has some upper or lower limit (like % Meeting SLA cannot be less than zero and cannot be greater than 100%), use curtailed normal distribution.
        - Normal Distribution or Curtailed Normal Distribution can be used where there is inconclusive information about the distribution.
        - For Non-normal Data, Johnson's curve for a freeform distribution fitting may also be considered. Though transformation of data adds to the complexity of interpretation and this should be given its due consideration while making such decision.
    - Select best measure of central tendency and variability of process parameter depending upon the curve fitting of the selected probability distribution with respect to the distribution of the actual data.
    - Plot appropriate control chart and calculate upper and lower control limits based on the selected value of α (Level of significance).
    - Refer to Appendix A for guidelines about statistical methodology.

5. Calculation of Probability Table and Process Capability Index for defining statistical limits
    - Based on the selected distribution, measure of central tendency and variability; calculate values of metrics associated to probability less than X, where X can be any value on continuous scale from 0 to 1 (or from 0% to 100%, if probability is expressed in terms of Percentage).

- Calculate appropriate process capability index for a given specification limits to identify if process is capable or it needs improvements.
- Cp is recommended process capability index for its ease of understanding and use and a Cp value more than 1 shows process is just capable and more than 1.33 shows that process can easily meet target.

6. Refer to Appendix A for guidelines about statistical methodology. Revision of baselines
    - Baselines shall be revisited minimum twice a year.
    - If there is change observed (from last baseline period to next period) in the baseline performance while revisit, old data should be discarded, and the baselines should be created based on the data of the latest period from the time performance has changed
    - Each baseline (performance) change should result into corresponding special cause identification (and corrective action if the change is in undesirable direction).
    - If there is no change in the performance baselines, control limits should not be revised and data should be maintained cumulative data for continuous baselines.

Metrics Baseline Reporting / Communication

- The Metrics Baseline Reporting shall follow baseline creation and so its minimum frequency is same as that of Metrics Baseline Creation.
- Reporting of Metrics Baseline shall be done using Standard Metrics Baseline Template on excel. This report shall contain following information:
    - General Metrics Baseline Report with a list of all the metrics at process and sub-process level with all the valid clusters (buckets) available in the report and their basic control chart with different control limits for different level of significance values (α values)
    - Probability Table for target setting
    - Process Capability and Probability calculation sheet for a given specification limits (upper and lower target values)
- These Reports shall be made available to all the stakeholders through QG Homepage – Reports and also by sharing through mails, wherever required.

Metrics Baseline Application

Following are common usage of a Baseline Report (though this may not be an exhaustive list):

- Metrics Baseline Report shall be used as an input for arriving at a statistical limit for the metrics selected at process or sub-process / factor level in a project. If baseline for a particular metric doesn't exist, arbitrarily chosen (management defined and approved) limit can also be used after appropriate approval. (Refer SDP for Metrics Collection & Analysis)

- Metrics Baseline Report should be used as one of the inputs in estimating project effort and other parameters during planning stage.
- Metrics Baseline Report, especially probability table, should be used in evaluating customer-defined targets to access risk of not meeting that target.
- In general, for any given metric target, if baseline is available, it should be used to access the probability of meeting this target and current process capability. This can serve as a trigger for identifying process improvement opportunity at project level.
- For existing Targets, metrics baseline shall be used to assess existing capability and wherever process is found incapable, this shall trigger as causal analysis followed by an action plan for capability improvement. This causal analysis can also be triggered by a high variation and adverse trend in metrics baseline from previous one and this should also lead to an appropriate action for implementation.
- At organisation level, metrics baseline report shall be used for quality target setting on the basis of probability of meeting target associated with it.
- Metrics baseline report shall also be used to calibrate process models / quality tools at organisation level (if a process model / quality tool is metrics baseline based model rather than standard parametric model like Rayleigh Curve).

6. Process Control

Process Control shall be applied at both process and sub-process level.

Methodology for Process Control can be:

- Management/ Customer Target Driven Process Control
- Statistical Process Control
  - Statistical Limit based Process Control
  - Control Chart Based Process Control

Targets from Management / Customer shall be tracked monthly through SQA Report and corrective action shall be taken for any instances where process output has gone out of target.

Wherever baselines and probability table is available, statistical limits shall be defined and causal analysis shall be initiated for any process output instance going out of these limits. Metrics Baseline Report can be used for defining statistical limits.

Sub-process/ process factor level control shall be exercise on critical sub-processes / factors wherever statistical limits are possible to define on the basis of historical data.

Data adequacy to draw the natural control limits is as follows:

Data Insufficient	Less than 6 data points
Data Marginally Sufficient	6 to 15 data points
Data Sufficient	More than 15 data points

If a particular metrics tracked for sub-process control has less than 6 data points, either alternate measures should be considered (where one can obtain more data points) or project should wait to collect more data points to reach beyond 6 before freezing on the control limits. So 6 can be taken as a minimum number of data points to draw the process control limits and 15+ is a desirable number of data points to draw those limits.

Control charts should also be applied wherever there is a time series data available and data points are sufficient to establish a control chart otherwise if data points are few or not time series, simple statistical limits will provide the same information what a control chart can provide. For details about control charts, refer to the section on Control Charts in Appendix A.

Common (though not mandatory and exhaustive) processes/ sub-processes / factors where control chart can be applied are as follows:

- Post release Software Defects reporting rate
- Post release Software Defects fixed per day
- Test case execution per person per day for testing projects or development projects where testing duration is significantly long.
- Inspection Rate and Defect Density during Inspection
- Turn Around Time for maintenance project

Wherever control chart is applied, natural control limits shall be revised on the basis of the current data after causal analysis of reason for change in the natural control limits. For details about implementing control charts, refer to the section - Method of Construction & Application of a Control Chart in Appendix A.

Capgemini has developed sub-process / factor monitoring and control tools based on process control principles. These are as follows:

- 2 x 2 Matrix for Controlling Inspection Process Output (Based on Statistical Limits)
- Health Index for controlling processes/ sub-processes/ factors in a maintenance project (Based on Control Charts and Index No.s)
    - X Bar, MR Control Charts are used for following measures in health index:
        - Reporting Rate Index
        - Closing Ratio Index
        - Turnaround Ratio Index
        - Aging Data Index
    - Health Index is a composite index of above mentioned four indices and so health index is also subjected to control charts
- For complete details about 2X2 Matrix and Health Index, refer to the Appendix B.

Health Index shall be used to control process/ sub-process/ factors in all the maintenance projects.

2x2 Matrix shall be applied to all the instances of inspection for inspection process output control unless customer has his own methodology implemented in a customer driven project.

7.Process Modeling

Making mathematical / stochastic models of process based on various observable attributes of a process help in process understanding, process composition and outcome prediction. So models enable pro-active actions and thereby help in defect prevention.

Mathematical / Stochastic Process Modelling is of two flavours:

a. Explanatory Process Modelling

b. Parametric Process Modelling

Explanatory Process Modeling

- When process model is developed on the basis of a set of measurable factors that are correlated to the response variable (process outcome parameter for which prediction is needed), this is called as Explanatory Process Modelling.
- Common method for stochastic explanatory process modelling is through regression analysis. For details about methodology, refer to the section on Regression Analysis in Appendix A.
- Regression Analysis based explanatory process model require re-calibration as soon as process performance changes. So frequency of coming up with regression based process models or their calibrated version shall be equal to the frequency of metrics baseline report (once in a year at minimum).
- Regression Analysis based process models shall be reported/ shared through a excel based template that shall provide following details:
  a. Factors in explanatory model and their coefficient of determination.
  b. Value of standard error.
  c. Model equation with auto-calculation features for its usages.
  d. Suggestion about usage.
- These models shall be made available through QG Home Page – Tools in a form of a report.
- Non- stochastic simple mathematical relation based explanatory models can also be used wherever stochastic models are unavailable.

- Current Examples of such models are Defect Prediction Model based on baseline data and Effort Distribution Model.

Parametric Process Modeling

- If a process model models a process parameter as a function of time by simply tracing its grossly stable repeated pattern over timeline established through data collection of same instances in multiple projects, it is called parametric process modelling.
- It is also known as black box process modelling because we are not interested in the factor(s) that cause(s) this pattern but the consistent pattern itself as it is sufficient for the purpose of prediction.
- Current parametric models used at Capgemini are for defect prediction and reliability estimation models that are part of 'defect prediction tool suite'. These models are as follows
  - Rayleigh Defect Prediction Model
  - Gompertz & NHPP Model Reliability Estimation and Post Release Defect Prediction

  Refer to Appendix B for detailed account of these models
- Development and implementation of Parametric Process Models is a continuous activity, so as and when a requirement for such model arises, the same shall be developed and evaluated through pilot results or other suitable means followed by the approval of SEPG and deployment.

Application of Process Models

- Process Model shall serve as an input for project estimation and planning.
- Process Models along with Metrics Baseline Report should also be used as an input for deciding over sub-process composition and sub-process targets to keeping in line with the overall targets.
- Process Models shall be used for predicting defect quality (i.e., defects for incomplete phases, post release period, re-open defect rate or count from sustenance or maintenance activity).
  - This can help in defect prevention in later phases based on the root cause analysis for the deviation of actual defects from the predicted defect or by setting harder target (on the basis of the anticipation of not meeting target from the predicted post release defects) for mining out most of the defects in early phases of the lifecycle.
- Process Models can also be used as an input for determining adequacy of code review, test cases, testing and other such critical processes. Though for such decisions, whole context is

considered through multiple types of inputs and so input from process model may or may not be most relevant input depending upon the nature of project and context.

- For scenarios like one described in previous point, where relationship is –
    - Non-linear,
    - Complex involving too many factors,

  Monte-Carlo simulation is recommended to be used to model the uncertainties of the process.
- Monte-carlo simulation shall also be used to verify life-cycle process composition with respect to meeting end goals of a project.
- Process model based on Monte-carlo simulation shall be revisited as soon as actual data is available to replace part of model parameter that was earlier simulated.
- Monte-carlo simulation based models shall also be revisited if there is a process change implemented or observed (due to some corrective or preventive action as part of causal analysis and resolution or due to other scenarios) in between the project in order to ascertain the effect the change on probability of meeting end goals of the project.

  (Ref. – Appendix A.4 for mathematical details of Monte-Carlo simulation method)

Updating & Revising Process Performance Models

- With every metrics revisit, process performance models shall be revisited for calibration due to change in the process performance baselines.
- Models shall be revised based on the change in the process performance and the version history of each of the models shall be maintained accordingly to reflect these periodic updates.
- Models that have changes shall be released with an updated version to the users at Capgemini along with the QMS release information.
- In those cases where process performance models are developed/tailored specific to a program or project using data from previous releases under the same program/ project as baseline, whenever a new release is completed, the model should be revisited and a new version should be made available (specially baseline part should add data from the latest finished releases).

Note: Guideline for Process Performance Modelling for development and services projects available at QG Homepage under QMS – Guidelines section

Appendix A

Common Statistical Methods

A.1 Hypothesis Testing

i. Hypothesis Testing - Definitions:

Hypothesis: A provisional idea whose merit needs evaluation and can be evaluated (falsified) on the basis of some parameter.

Test Statistic: Representative of parameter taken from sample to evaluate hypothesis

Examples:

S.No.	Propositions (Hypothesis)	Test Statistics
1	Lets go for movie 'Shivaji' rather than 'Ocean 13'	Not falsifiable, so it is not a hypothesis and so it can not be tested.
2	Coke is better than Pepsi in Taste	% of respondents for Coke and for Pepsi
3	A TL can write a good design document	Not falsifiable, so it is not a hypothesis and so it can not be tested.
4	Function Based Testing is as good as Flow Based Testing	Average defect density of next phases after function based and flow based testing
5	Bigger Project have more SV than smaller Projects	Average SV in Bigger and smaller Projects

Hypothesis Testing: Decision of rejecting or not rejecting a 'Proposition' in a 'Test Statistic' based comparison against a 'Counter-Proposition'.

Null Hypothesis: A proposition that is being evaluated.

Alternate Hypothesis: A counter-proposition of what has been proposed in null hypothesis.

Examples:

Proposition to be tested	Test Statistic	Null Hypothesis	Alternate Hypothesis
Coke is better than Pepsi in Taste	Test Statistic: % of respondents favoring a particular brand drink	Null Hypothesis H0 : % favoring Coke = % favoring Pepsi	Alternate Hypothesis H1 : % favoring Coke > % favoring Pepsi
Function Based (FNB) Unit Testing is as good as Flow Based (FWB) Unit Testing	Test Statistic: Mean of Post UT Defect Density	Null Hypothesis H0 : Post UT DD Mean (FNB) = Post UT DD Mean (FWB)	Alternate Hypothesis H1 : Post UT DD Mean (FNB) ? Post UT DD Mean (FWB)

P Value: A term used to denote probability of occurrence of a value less than or equal to the sample statistic when null hypothesis is true.

Level of Significance (α): The probability below which you reject null hypothesis (H0).

Notes:

i. If P Value < Level of Significance ⇒ Reject H0

ii. Level of Significance is a subjective decision.

iii. Commonly it is selected as less than 10%, 5% or 1% probability.

Level of Confidence: Mathematically, it is ⇒ 100% - (Level of Significance in %).

Errors in Decision: Errors in decision are of just two types:

i. Error of rejecting truth.

ii. Error of not rejecting what is untrue.

Error of Rejecting Truth (Type 1 Error): When you reject Null Hypothesis when it is true.

Notes:

i. Probability of doing this error is equal to the level of significance (α) selected.

ii. That is why if data is error free and reliable, it makes sense to select α as small as possible.

Error of Not Rejecting Untrue (Type 2 Error): When you don't reject Null Hypothesis when it is untrue.

Notes:

i. Probability of doing this error is equal to the probability of a sample statistic (like sample mean) falling in acceptable range.

ii. This is not equal to level of confidence (1- α).

iii. But if you select smaller α, bigger will be Type 2 Error (denoted by β)

	Hypothesis True	Hypothesis Untrue
Hypothesis Rejected	Type - I Error	Correct Decision
Hypothesis Not Rejected	Correct Decision	Type - II Error

Errors in Decision (Figure – 1)

ii. Hypothesis Testing – Process Steps:

i. State H0 and H1.

ii. Set alpha (α).

iii. Gather data for the test.

iv. Calculate the test statistic.

v. P-value to reach a conclusion.

iii. Testing Hypothesis Using Excel:

Following document attached below explains:

i. What test should be used when?

ii. How to use excel for testing hypothesis?

Testing Hypothesis
Using Excel

A.2 Statistical Process Control

i. Purpose:

It is used to monitor, control, and improve process performance over time by studying variation and its source. This achieve through the use of control charts.

ii. Control Chart:

A method of continuous hypothesis testing at fixed time intervals to detect any shift in mean or change in variability of a process output and thereby exercising a control over the process by means of the corrective actions(process interventions), whenever required.

iii. Theory of Process Control:

i. Process output varies because of the environmental noise caused by the complex interaction of the factors affecting process output.

ii. Factors that are cause of variability in the process output may be

   i. Known and controllable factors

   ii. Known and uncontrollable factors in current process setup

   iii. Unknown factors

iii. Since Unknown factors are not known, so they are always uncontrollable; And so factors affecting process output can be simply divided into ‘Controllable’ and ‘Uncontrollable’ factors.

iv. Uncontrollable factors are known as ‘Chance Causes’ (‘Common Causes’ in modern terminology) of variation and controllable factors are known as ‘Assignable Causes’ (‘Special Causes’ in modern terminology) of variation in process output.

v. Common Causes of variation can be eliminated only through process setup redesigning or technological improvement that can make uncontrollable factors controllable.

vi. Common causes induce a stable variability in the process output; as a result process output parameter shows almost a fixed upper and lower bound of varying parameter values naturally.

vii. These upper and lower bound of varying parameter value are known as ‘Natural Control limits’ and probability of finding a parameter value outside these limits without a role of any other factor (Special Cause) is almost negligible.

viii. If any value of process parameter from the sample of process output goes beyond these limits, a causal analysis is initiated to identify the responsible special cause(s) and remove/control it to bring process output back in natural control limits.

iv. Method of Construction & Application of a Control Chart

i. Find stable data to characterize the process output parameter

ii. Identify the type of probability distribution

iii. Establish central value, UCL (Upper Control Limit), and LCL (Lower Control Limit) for the measure of central tendency (Mean, Median etc.) and variability (Range, Standard Deviation etc.)

General Mathematical Model for Control Chart

- Let 'w' be a sample statistic that measures some process characteristic that we want to control, general model for control chart can be defined as:
  - UCL = Mean of 'w' + L * Standard Deviation of 'w'
  - Central Line = Mean of 'w'
  - LCL = Mean of 'w' – L * Standard Deviation of 'w'

    Where L is an arbitrarily chosen distance of the control limits from Central Line and generally it is taken as 3.
- Control chart developed on the above model are known as 'Shewhartian Control Chart'.

iv. Plot data point on the graph with central limit with UCL and LCL

v. Perform causal analysis of the data point going out of control (out of UCL or LCL)

vi. Remove out of control point after removing/controlling 'Special Cause'

vii. Revise control chart (Revise Control Limits if process pattern changes)

v. Type of Control Charts

i. Control charts can be classified on the basis of type of data into two types:

- Variable Control Chart (Based on Variable Data)
- Attribute Control Chart (Based on Attribute Data)

ii. **Variable Data:** Variable data are measured on a continuous scale Time, weight, distance or temperature can be measured in fractions or decimals. The possibility of measuring to greater precision defines variable data.

<u>Examples of Variable Control Charts:</u> X bar, R chart, S chart,XmR Chart

<u>Sample measures:</u> Defects Vs Size, Defects/hr., Delivery variance, Test cases/hr., FP's / Person days, KSLOC/ Per Days

iii. **Attribute Data:** Attribute data are counted and cannot have fractions or decimals. Attribute data arise when you are determining only the presence or absence of something: success or failure, accept or reject, correct or not correct.

<u>Examples of Attribute Control Charts</u>: p chart, np chart, c Chart, u chart

Sample Measures: No of defects reported, No of resignation, No of system problem reported, No of absentees.

vi. How to Construct X Bar MR Chart

X Bar Chart:

UCL: (+ 3 Sigma) = Mean(X) + 2.66*MR
CL: = Mean(X)
LCL: (- 3 Sigma) = Mean(X) – 2.66* MR
Where MR = Moving range and 2.66 is constant

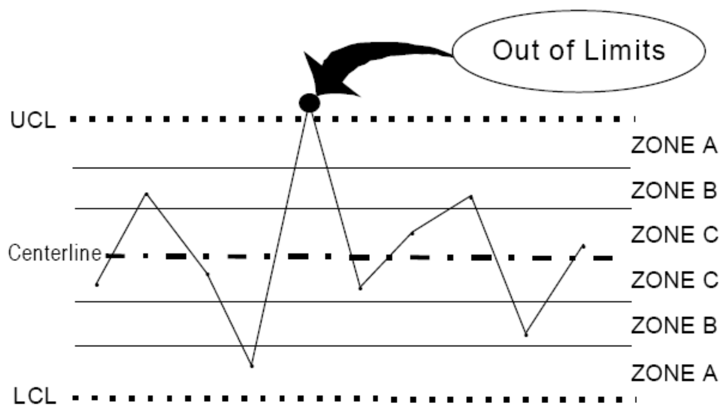
MR Chart:

UCL: (+ 3 Sigma) = 3.267 * Mean(MR)
CL: = Mean (MR)
LCL: = 0

vii. Sensitization Rules for Control Charts

These rules are used to make early detection of process shift.
Four basic sensitization rules popularized by Western Electric are as follows:

Rule 1: A single point outside the control limits (outside 3 Sigma).



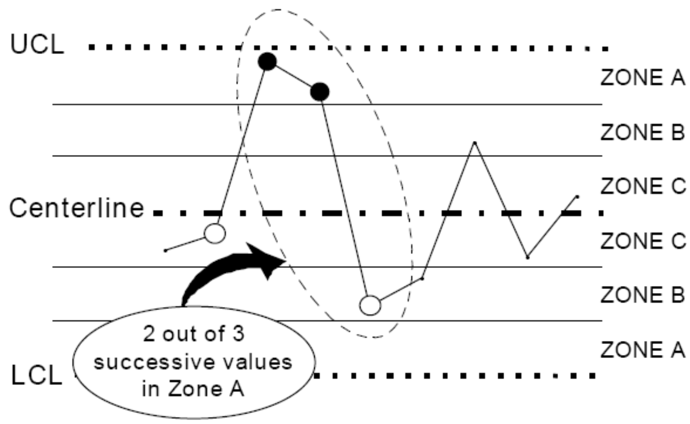
Note:

Zone C is region that is 1 Std. Deviation away from the central line

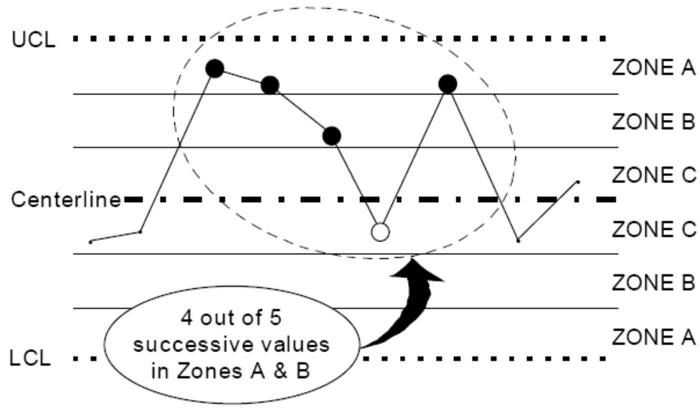
Zone B is region that is 1 to 2 Std. Deviation away from the central line

Zone A is region that is 2 to 3 Std. Deviation away from the central line

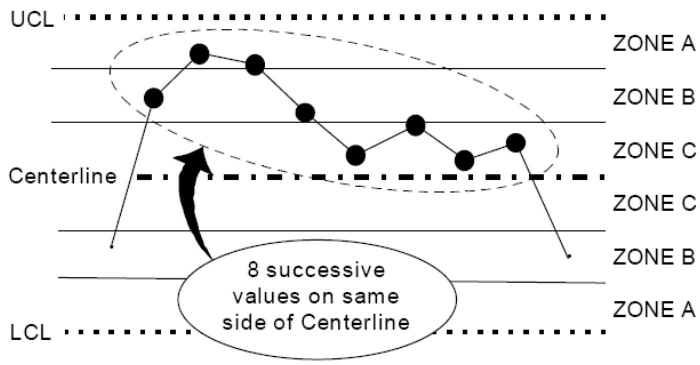
Rule 2: Two out of three successive points are on the same side of the centerline and farther than 2 σ from it.



Rule 3: Four out of five successive points are on the same side of the centerline and farther than 1 σ from it.



Rule 4: A run of eight in a row is on the same side of the centerline. Or 10 out of 11, 12 out of 14 or 16 out of 20.

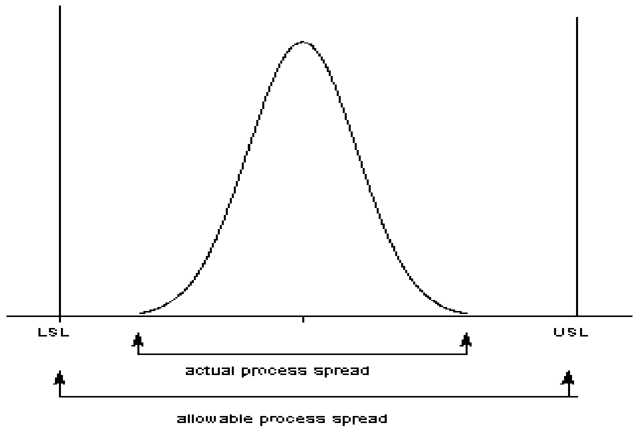


viii. Process Capability

i. Process capability compares the output of an in-control process to the specification limits by using capability indices.

ii. The comparison is made by forming the ratio of the spread between the process specifications (the specification "width") to the spread of the process values, as measured by 6 process standard deviation units (the process "width").

iii. A capable process is one where almost all the measurements fall inside the specification limits. This can be represented pictorially by the plot below:



iv. There are several statistics that can be used to measure the capability of a process: Cp, Cpk

v. Most capability indices estimates are valid only if the sample size used is 'large enough'. Large enough is generally thought to be about 50 independent data values.

vi. The Cp, Cpk statistics assume that the population of data values is normally distributed.

vii. **Potential Capability (Cp):** It is mathematically expressed as:

Cp = (USL – LSL) / 6*σ

Note:

1. USL stands for Upper Specification Limit and LSL stands for Lower Specification Limit.

2. Value of Cp more than one makes process just ‘capable’ of meeting Specifications (Targets) so Cp more than 1.3 is recommended value.

viii. **Demonstrated excellence (Cpk):** We can adjust Cp for the effect of non-centering by computing:

Cpk = (1-k) * Cp

Where k=abs (D - Mean)/(1/2*(USL - LSL)); D = (USL+LSL)/2

ix. **Process Capability Index for Non-normal Process:** Index for non-normal process is called Cnpk (for non-parametric Cpk). Its estimator is calculated by:

Cnpk = Minimum of {(USL – Median) / (99.5th Percentile – Median)} and {(Median – LSL) / (Median – 0.5th Percentile)}

A.3 Linear Regression Analysis for Process Modeling

i. Linear least squares regression is by far the most widely used modeling method.

ii. Used directly, with an appropriate data set, linear least squares regression can be used to fit the data with any function of the form

$$f(\vec{x};\vec{\beta}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots$$

in which

1. Each explanatory variable in the function is multiplied by an unknown parameter,
2. There is at most one unknown parameter with no corresponding explanatory variable, and
3. All of the individual terms are summed to produce the final function value.

i. Calculation of Model Constants:

i. To illustrate, consider the straight-line model,

$$y = \beta_0 + \beta_1 x + \varepsilon.$$

Where ‘y’ is a dependent variable, ‘x’ is an independent variable, β1 and βo are constants and ε is error part.

ii. For this model the least squares estimates of the parameters would be computed by minimizing:

$$Q = \sum_{i=1}^{n} \left[y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)\right]^2$$

iii. Taking partial derivatives with respect to β1 and βo yields the following estimators for the parameters:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

ii. Assumptions of Linear Regression based Process Models

Linear regression has following two underlying assumptions that must be validated before using for process modeling:

1. It is assumed that the relationship between variables is linear
2. It is assumed in multiple regression that the residuals (predicted minus observed values) are distributed normally

iii. Model Fitting, Correlation Coefficient and Coefficient of Determination

i. Correlation coefficient (r) measures the strength and the direction of a linear relationship between two variables.

ii. The mathematical formula for computing r is:

$$r = \frac{n\sum xy - \left(\sum x\right)\left(\sum y\right)}{\sqrt{n\left(\sum x^2\right) - \left(\sum x\right)^2}\sqrt{n\left(\sum y^2\right) - \left(\sum y\right)^2}}$$

Where n is the number of pairs of data of x and y variables between which relationship is being evaluated.

iii. A perfect correlation of ± 1 occurs only when the data points all lie exactly on a straight line. If r > 0, the slope of regression line is positive. If r < 0, the slope is negative.

iv. Square of Correlation Coefficient is known as Coefficient of Determination (r^2) and it represents explained variability of dependent parameter 'y' by the independent parameter 'x' through regression equation out of total variability.

v. Coefficient of Determination can be 1 or 100% if all variability of 'y' can be explained by 'x'. That means, if you put value of 'x' in regression equation, value of 'y' that you are getting should be exactly same as observed value of 'y' or in other words, your model is completely fit over the observed values.

iv. Coefficient of Determination should be 0.8 (80% in percentage) or more to show a good model fitting.

A.4 Monte-Carlo Simulation Technique

i. Historical Background

i. The Monte Carlo method, as it is understood today, encompasses any technique of statistical sampling employed to approximate solutions to quantitative problems.

ii. W. S. Gossett (pen name "Student") randomly sampled from height and middle finger measurements of 3,000 criminals to simulate two correlated normal distributions. He discusses this methodology in two mathematical papers in 1908.

iii. Ulam's (known for invention of H-Bomb) contribution was to recognize the potential for the newly invented electronic computer to automate such sampling.

iv. Working with John von Neuman and Nicholas Metropolis, he developed algorithms for computer implementations.

v. Ulam and Metropolis (who named the new methodology after the casino of Monaco with the name Monte Carlo) published the first paper on the Monte Carlo method in 1949.

ii. Mathematical Details

i. Monte Carlo simulation is a method for iteratively evaluating a Deterministic model using sets of random numbers as inputs to get stochastic output.

ii. It is used when the model is complex, nonlinear, or involves more than just a couple uncertain parameters.

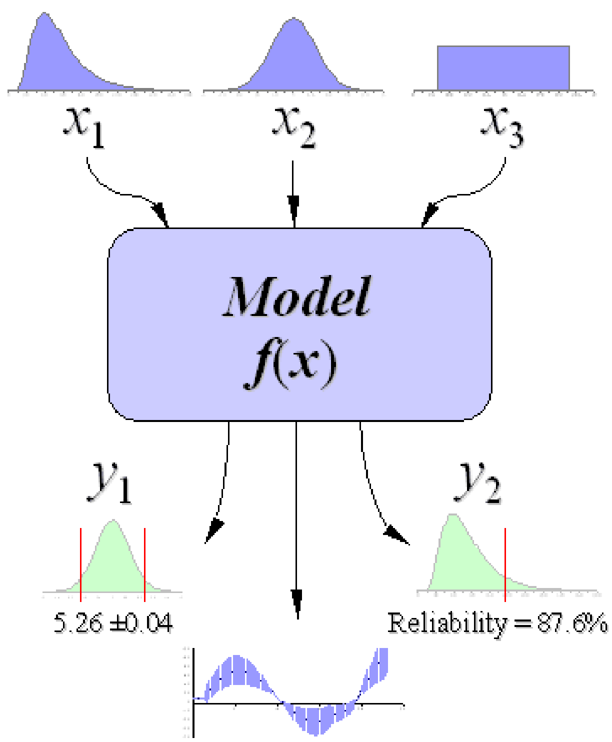
iii. The Monte Carlo method is just one of many methods for analyzing *uncertainty propagation*.

iv. Goal is to determine how *random variation*, *lack of knowledge*, or *error* affects the *sensitivity*, *performance*, or *reliability* of the system that is being modeled.

v. Monte Carlo simulation is categorized as a *sampling method* because the inputs are randomly generated from *probability distributions* to simulate the process of sampling from an actual *population*.

vi. The data generated from the simulation can be represented as *probability distributions* or converted to *error bars*, *reliability predictions*, and *confidence intervals*.

Uncertainty Propagation



Schematic showing the principal of stochastic uncertainty propagation. (The basic principle behind Monte Carlo simulation.)

iii. Monte-Carlo Simulation Algorithm

Step 1: Create a parametric model, y = *f*(x1, x2, ..., xq).

Step 2: Generate a set of random inputs, xi1, xi2, ..., xiq.

Step 3: Evaluate the model and store the results as yi.

Step 4: Repeat steps 2 and 3 for *i* = 1 to *n*.

Step 5: Analyze the results using histograms, summary statistics, confidence intervals, etc.

iv. Steps to Apply Monte-Carlo Simulation in Process Performance Modeling (PPM)

i. Define PPM objective.

ii. Identify life-cycle process steps.

iii. Establish baseline performance measures for each of the process step.

iv. Link performance measures of each step to next to arrive at overall performance measure value.

v. Simulate each step performance measure using Monte-Carlo simulation to account for inherent variability in process performance of each step.

vi. Calculate impact of variability through simulation on overall performance measure value.

vii. Evaluate alternate paths of life cycle process steps (Choices).

viii. Use sensitivity analysis to evaluate critical step of lifecycle process.

ix. Recalibrate simulation results as actual data recorded during the progress of project.

v. PPM Using Monte-Carlo Simulation – A Case Study

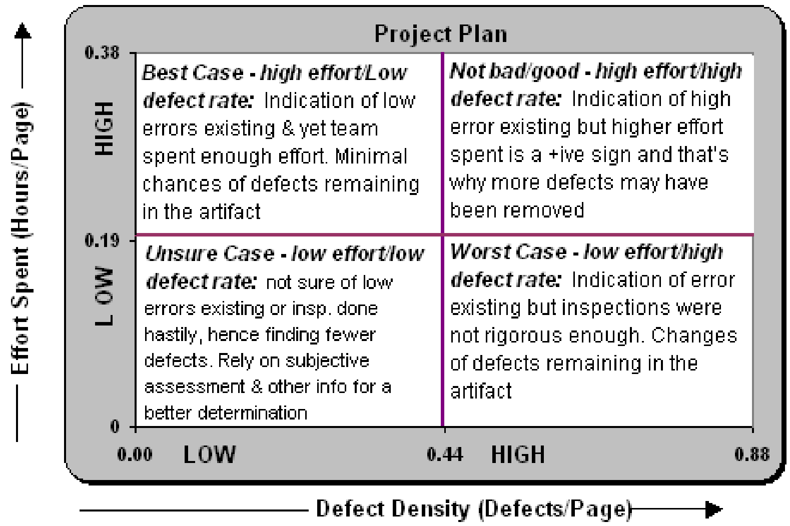
Monte-Carlo
Simulation Case Study

Appendix B

Process Control & Modeling Tools

B.1 2x2 Matrix for Inspection Process Control

Following matrix is used for measuring the effectiveness of inspection and review process and exercising control by initiating re-inspection wherever inspection is ineffective. It gives a fair idea of defects found with respect to effort spent by inspection team.



Guidelines for using this Matrix:

Excel sheets are available on QG home page for various types of artifacts. It contains the boundary values for defect density and effort spent to divide the overall graph area in four parts.

User need to fill in the data at the specified cells in the excel sheets. It will be plotted in the graph and will lie in one of the quadrant as shown above. Based on the scenario explained in above graph, proper interpretations can be made. Corrective and preventive action (if any) need to taken.

B.2 Health Index for Maintenance Projects

Capgemini has developed an Index to assess the performance of projects that are in the maintenance phase. This index is based on:

- Problems reported by the customer,
- Severity of the reported problems,
- Problems closed by the engineering team,
- Rate at which problems are reported,
- Service Level Agreement (SLA) with the customer, &
- Age of open problems, etc.
- Hold duration for a problem

From the above-mentioned parameters, Calculate

- Closing ratio
- Reporting rate of problems
- Turnaround ratio
- Age, for each category of the problem (Critical, Major, Minor, & Enhancement or Others)

These metrics are calculated and converted into four different indices i.e. Closing ratio, Reporting Rate, Turnaround Ratio, & Aging index.

To concentrate on the overall project performance rather than on individual project performance indicators, take a weighted sum of all the above-mentioned indices to calculate the Health Index. This is the overall indicator of the performance or health of the project.

Calculating Health Index:

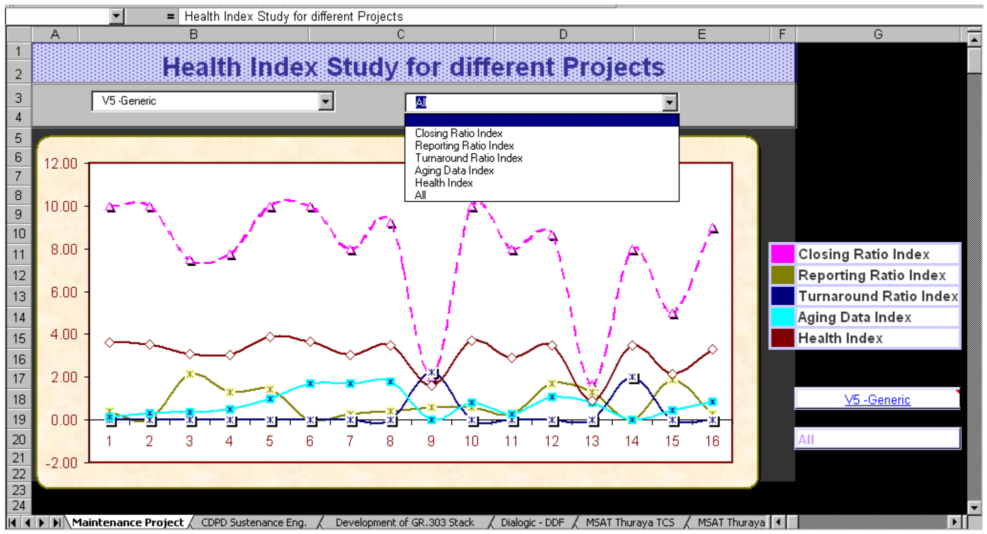
Using MS Excel, Overall process has been automated for calculating these indices from the data generated by the project. On the basis of the raw data generated by the project, all the five indices i.e. Closing Ratio, Reporting Rate, Turnaround Ratio, Aging, and Health index are calculated automatically.

This is mandatory for each maintenance kind of projects. PM gets in touch with QG for implementing this. QG helps in:

- Training PM and SQA on HI
- Fixing weights for different parameters
- Defining the mechanism of reporting and analysing data
- Interpretations of the data and HI

Digital Dashboard:

A screen shot of a combined study of all the indices of all maintenance projects at CAPGEMINI is given below. It is updated automatically from all the data reported from different projects, it acts like a digital dashboard for senior management. It shows all the indices for a particular project. Using two boxes shown in the picture, one can select any maintenance project and any index for the selected project. Currently it is showing all the indices for a particular project. On clicking the project name we see all other details related to defined metrics.



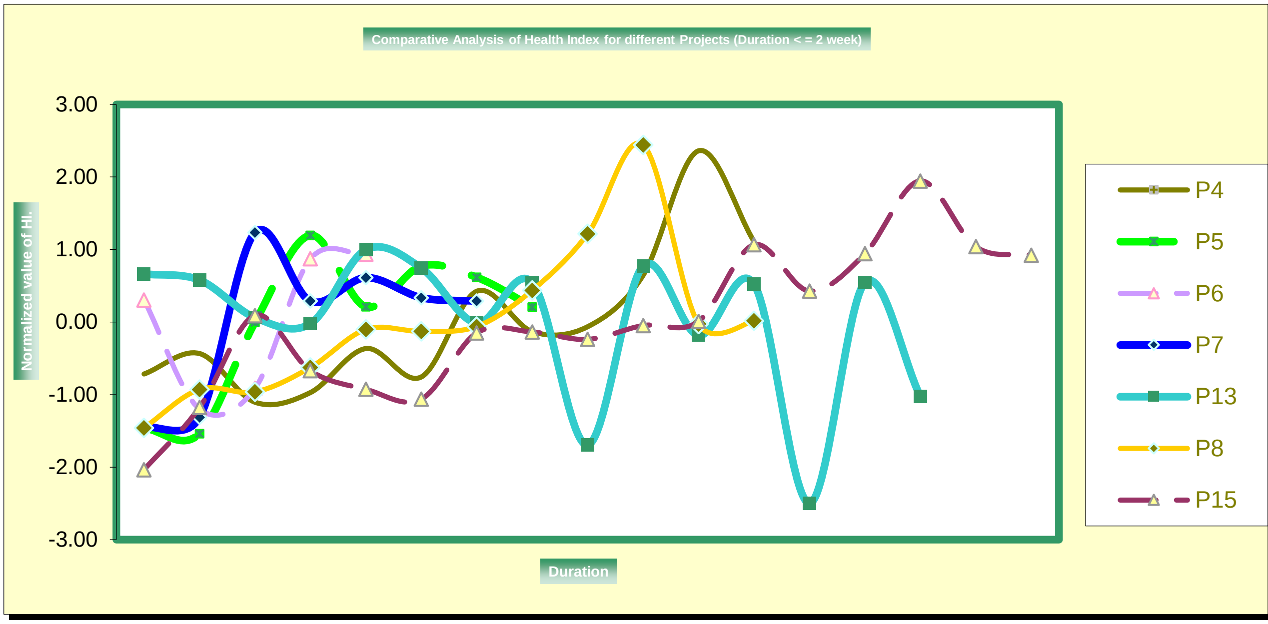
The lower the indices, the better the health of the project. In absolute terms the value of all the indices does not signify much, and these need to be interpreted in relative terms. It helps project managers and sr. management in:

- Controlling the rate at which problems are closed,
- Measuring the degree to which project is meeting the SLA,
- Knowing the overall trend of problem reporting rate, which can be further analysed w.r.t. time or other factors for better resource allocation,
- Distinguishing between the controllable and uncontrollable factors (e.g. Reporting Rate),
- Reflecting overall age of the open problems,
- Analysing overall performance w.r.t. other indices,
- Predicting the customer satisfaction, & many more.

Comparison across the projects:

To compare, standardize value of health index for each project using standard statistical formulas. This standardized data is plotted on a graph and trend line was observed for each project. Thus slope of trend line is comparable across all the projects. Based on value of the slope of trend line, all maintenance projects can be ranked from the worst to the best.

Following Graph shows the comparison across the projects. P4, P5, P6, etc. are different projects, and each graph line shows standardized health index corresponding to each project. This kind of analysis is sent to Sr. Management on periodic basis.



Process for tracking HI:

- On monthly basis, SQAs report data to QG
- QG collates data for all projects
- QG prepares digital dashboard and comparative report
- QG shared % Meeting SLA Metrics from it through BU Report till the highest level of management.

B.3 Rayleigh Defect Prediction Model

Basic Principle:

i. Rayleigh distribution-based model is one of the earliest models to be used to model defects throughout the entire life cycle of software.

ii. It has been empirically well established that software projects follow a lifecycle pattern described by the Rayleigh density curve (Norden, 1963; Putnam, 1978).

iii. The Rayleigh model is a special case of the Weibull distribution when m = 2. Its PDF(Probability Density Function) and CDF (Cumulative Distribution System) are :

$$PDF = f(t) = K * t/c^2 * e\hat{}[-(t^2/2c^2)], \quad \text{for } 0 \leq t < \infty, c > 0$$

$$CDF = F(x) = K*[1 - e\hat{}[-(t^2/2c^2)]], \quad \text{for } 0 \leq t < \infty, c > 0$$

Here 'K' parameter can be physically interpreted as total number of expected defect for the entire life cycle of the software. 'C' parameter has no direct interpretation but it is product of square root of 2 and time when maximum defects are expected to occur (tm).

Model Use Scenario:

i. Defects of at least three phases are available.

ii. Defects (of available phases) are following Rayleigh Distribution Pattern (R-Square (coefficient of determination) is more than 80%).

Model Inputs:

i. There are just two basic inputs required:

   a. Time duration (Planned) for incomplete phases and time duration (Actual) for phases that have been completed. Unit of time can be any unit (Months, Weeks, Days, hours) as long as same unit is used for the time duration of all the phases.

   b. Defects (Critical, Major & Minor) found during phases that are complete

ii. Shown below is the snapshot of input sheet of automated tool for Rayleigh Model available at QG Homepage:

PHASE NAME	CUM. TIME	CUM. DEFECTS
Rqmt.	39	669
Design	99	2399
CUT	141	3341
SSI/SI	171	

Model Output:

i. Output of model comes in from of predicted defects.

ii. Defects Predicted for the phases that are complete serve as a mean to determine model error and co-efficient of determination for the fitness of model use.

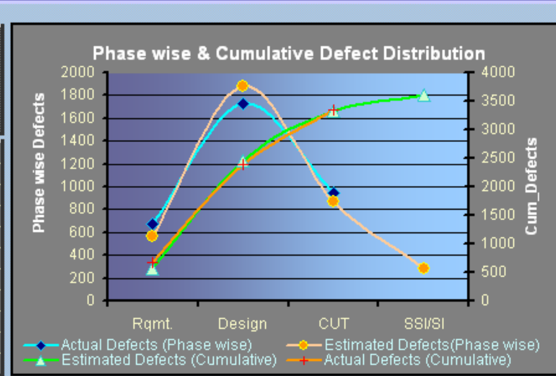
iii. A value of co-efficient of determination less than 80% shows model may not be best fit to the actual data and any interpretation should take care of this error.

iv. It also provides range of defects based on level of confidence selected for calculating confidence interval.

v. Shown below are the snapshots of information present on the output sheet of automated tool for Rayleigh Model available at QG Homepage:

Phase Wise Estimated Vs Actual Defect Details

Phase Name	Actual Defects	Estimated Defects
Rqmt.	669	568
Design	1730	1884
CUT	942	867
SSI/SI		281



Defects Passed To The Customer	161

Model Parameters

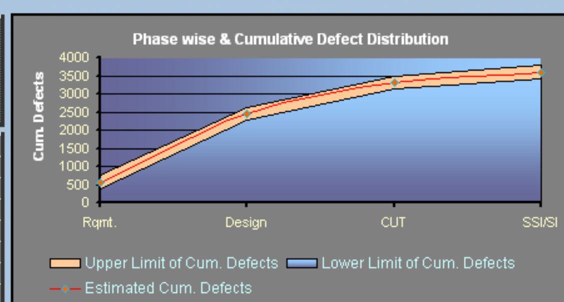
K (Total No. of Injected Defects)	3760.582087
C = tm * Squr(2)	96.34718385
tm (Peak of curve)	68.12774705
Sum of Squares of Error (SS)	13456.76523
R Sqr. (Coeffi. of Determination in %)	99.89%

Interval Estimates of Phase Wise Defects

Level of Confidence	95%

This is defined is terms of % and typical values can be 80%, 90% or 95% but never 100%.

Phase Name	Upper Lim. of Cum. Def.	Lower Lim. of Cum. Defects
Rqmt.	750	388
Design	2623	2282
CUT	3498	3141
SSI/SI	3791	3408



Select a phase for phase wise defects	SSI/SI	294 268	Defect Range

Model Limitations:

i. This model cannot be used where phase wise defect pattern does not follow Rayleigh Distribution.

ii. This is based on a parametric model and like any parametric model, it relies on the assumption that if the same pattern continues in future then its output will be valid but this assumption is not always valid in real condition.

iii. It cannot be used unless three phases are complete.

iv. Best application scenario for this model is waterfall lifecycle of software development so if life cycle is very much different from waterfall like completely incremental, RAD, spiral etc. this model is not fit for use.

B.4 Gompertz & NHPP Model for Reliability Estimation & Defect Prediction

Basic Principle

i. These models are based on the basic principle of ‘Reliability Growth Modeling’

ii. Basic principle behind reliability growth models is that there can be only a finite set of defects in software so during testing as we keep detecting and fixing defects, the probability of finding more defects decreases.

iii. This decrease in probability of detection of defect that is also proportional to the failure rate during execution test cases results in ‘growth in software reliability’.

iv. Any reliability growth model traces this pattern of failure of test cases to predict reliability growth.

v. **Gompertz Model:** The Gompertz reliability growth model is mathematically given by Virene as:

$$R = ab^{c^T}$$

Where ►

R = the system's reliability at development time T, or at launch number T, or stage number T.

a = the upper limit that the reliability approaches asymptotically as T approaches infinity or the maximum reliability that can be attained.

ab = the original level of reliability at T = 0, or the starting reliability value.

c = the growth pattern indicator (small values of c indicate rapid early reliability growth and large values of c indicate slow reliability growth).

vi. **NHPP Model:** It is known as non-homogenous Poisson process model because failure rate in this model is Weibull distributed and so failure rate may or may not be homogenous over a period of time.

vii. Expected number of failures in a time interval are Poisson distributed with probability of ‘n’ failures in time ‘T’ given by:

$$\Pr[N(T) = n] = \frac{(\lambda T^{\beta})^n e^{-\lambda T^{\beta}}}{n!}; \; n = 0, 1, 2, \ldots$$

Where ►

λ is scale parameter of model and depends upon the unit of time chosen.

β is shape parameter of the model and its value equal to 1 shows constant failure rate. Beta less than 1 shows decreasing failure rate (Reliability Growth) and more than 1 shows increasing failure rate (Decrease in Reliability)

Model Use Scenario

Gompertz and NHPP Model can be best used in the last phase of testing when test case wise, date wise outcome and effort of testing is recorded.

Model Inputs

i. Test case fail/pass data recorded along with the date and time of execution of those test cases serves as an input of Gompertz Model.

ii. For NHPP model, other than the above described inputs, test case execution time from system is also needed. In some cases it can be grossly approximated with test case execution time at each setup.

Model Outputs

i. Gompertz model provides reliability growth curve as an output that can be one of the objective input for a 'stop testing' decision.

ii. Gompertz model provides reliability status at the end of testing. It can also predict reliability for a specified time in future.

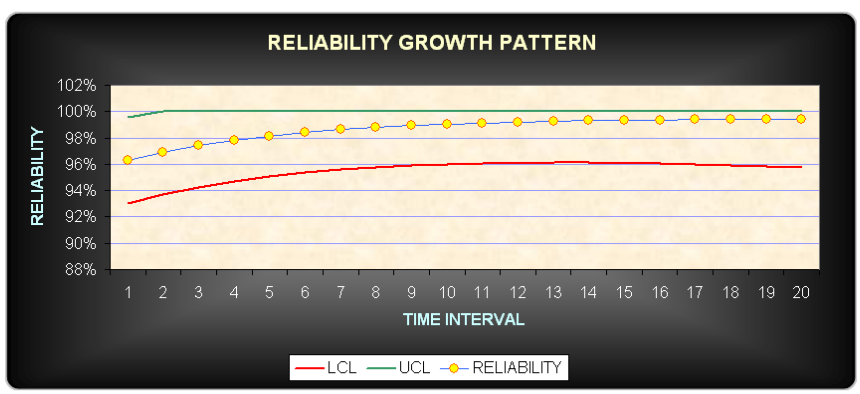
iii. NHPP model provides a prediction of post release defects expected to be reported from the field during a specified period of time. It can also provide expected failure rate and MTBF at the end of testing.

iv. Shown on next page are the snapshots of information present on the output sheet of automated tool for Gompertz and NHPP Model available at QG HomePage.

Gompertz Model Output

CURRENT STATUS OF SOFTWARE RELIABILITY

DATE FOR WHICH S/W RELIABILITY STATUS IS GIVEN	28-Jul-07
SOFTWARE RELIABILITY ON ABOVE MENTIONED DATE	99.34%
UPPER LIMIT OF 95% CONFIDENCE INTERVAL	Almost 100%
LOWER LIMIT OF 95% CONFIDENCE INTERVAL	96.08%



NHPP Model Output

Model Related Information

$\beta = n / (n*\ln(T) - sum(\ln(ti)))$	0.2777
$\lambda = n / (T^\wedge\beta)$	1.3238
$\lambda_i(T) = \lambda*\beta*(T^\wedge(\beta-1))$	0.0681
$MTBF = 1/\lambda_i(T)$	14.6778
$\lambda_c = \lambda*(T^\wedge(\beta-1))$	0.2454
$MTBFc = 1/\lambda_c(T)$	4.0755

Model Based Prediction

Time Interval		Failures
From Time T1	To Time T2	
0	365	7

Model Limitations:

i. This model is based on a parametric model and relies on the assumption that-

   a. All the scenarios have been covered adequately while testing the product.

   b. The test cases written are adequate enough to find defects and the maximum code coverage has been achieved.

   c. The entire product is tested and not only the added or enhanced feature.

ii. There is no way that this model can itself validate or assure that above specified assumptions are met, but if above assumptions are not met user is likely to get invalid results.

iii. In case Reliability growth curve is Sigmoid and its tails are very bent, standard Gompertz model will give only approximate results because that type of curves are either modeled by Modified Gompertz model or by other Sigmoid models.

iv. Post-release defects predicted by NHPP model is based on assumption that the software is not kept idle and is in use with same rigor that was applied during testing. So in case, software is not in use at customer end or is not that much rigorously used at customer end, post release defects are likely to come with lesser rate than predicted.

Appendix C

Related Handbooks

- Metrics Handbook-TL9000 (7.2.2)
- Metrics Handbook-TL9000 (8.6.1)