Measurement Reports

Erika Vintan
BMO IT Best Practices & QA, Toronto, Canada
September 2008

Abstract

Reports aim to inform, as clearly and succinctly as possible. They should be easy to read, and professional in their presentation. Measurement Reports are no different. Exactly what you include in your report and how you present it will vary according to your discipline and the specific purpose of the report. The intent of this paper is to give general guidelines and to share the benefits and importance of measurement reporting.

Introduction

We use measurements in our daily lives to make better decisions based on facts and information. In the software world:
Measurements - Are techniques or methods that apply software measures to software engineering objects to achieve predefined goals
Measure - Is a mapping from a set of software engineering objects to a set of mathematical objects
Measurement goals - vary with

- the software engineering object being measured
- the purpose of measurement
- who is interested in these measurements
- which properties are being measured
- the environment in which measurement is being performed

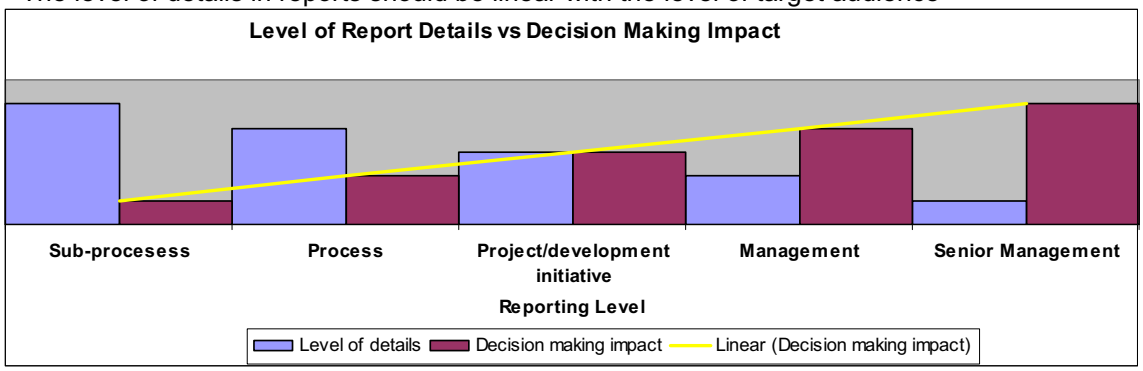
Once a Measurement Process is implemented, results of measurement and analysis are communicated to relevant stakeholders to support decision making through reports.

The report's structure emphasizes the way in which the information was collected as much as the data itself. The following are basic questions to ask and responses taken into consideration when creating reports:

- **Why?**
  - For Effective Communication
- **For whom?**
  - Target Audience (media/stakeholder type) – the user of the metric information
- **How?**
  - Distribution channel
- **How often?**
  - Frequency – event per unit time
- **Data integrity**
  - Completeness, Accuracy, Correctness, Validity

- **Identifiable sources**
  - Works Cited, Reference List, Bibliography etc. - capability to find specific data without ambiguity
- **Support actions**
  - Fulfill stakeholders information needs - level of detail and classification of information support reporting requirements
- **Use Standard Definitions**
  - Pre-defined, published, established for use as a rule or basis of comparison in measuring
- **Be adaptable**
  - Customizable based on requirements

The level of details in reports should be linear with the level of target audience



Sample Report:

The **Defect Measurement Report** is a quantitative tool for summarizing defect results in a project. The data is derived from creating measurement goals for selected indicators in the INITIATION PHASE and comparing them to the actual totals for each indicator after the project has been delivered in CLOSURE.

Defect Measurement Reports are more comprehensive than ROOT CAUSE Analysis, and include additional defect indicators that help to track and identify problems and causes that occur during a project. They allow the development team to analyze the results and make recommendations on a broader base of information

Data reported is saved in an historical process database and used in the creation and update of the indicator baselines and organizational trends. Recommendations are packaged with the Post Implementation Review (PIR) in order to be used for future projects and process improvement.

Defect Measurement Report includes the following sections:

1. Purpose

- Describes the intent of the Defect Measurement Report
  E.g. summarize the analysis of the results of the defect measurement indicators
- Specifies details of the type of defects for the report
  E.g. internal development, vendor, global resourcing

2. Strategy

- Describes the strategy/approach taken
  E.g. phased implementation

3. Data reporting

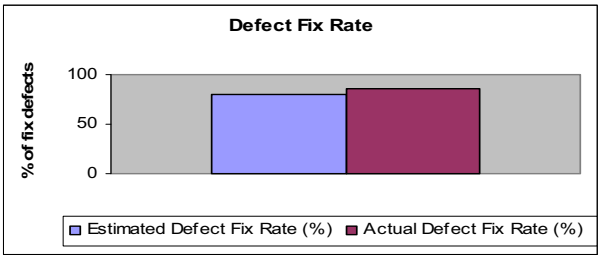
- This section includes the data collected from the project based on pre-defined operational definitions as part of the measurement process requirements.
- Data is normalized using size E.g. Use Case Points
- The data will be stored in the organizational process database

Historical data E.g.

Test Level Assessment (TLA Score)
of Test Plans (including UAT)*
of Test Design Matrices (including UAT)*
of Test Cases (including UAT)*
Total number of Testing Defects (Problem Logs)
Total number of Testing Defects (Problem Logs) by Severity S0
Total number of Testing Defects (Problem Logs) by Severity S1
Total number of Testing Defects (Problem Logs) by Severity S2
Total number of Testing Defects (Problem Logs) by Severity S3

Defect Fix Rate - % of defects open vs closed (sample)

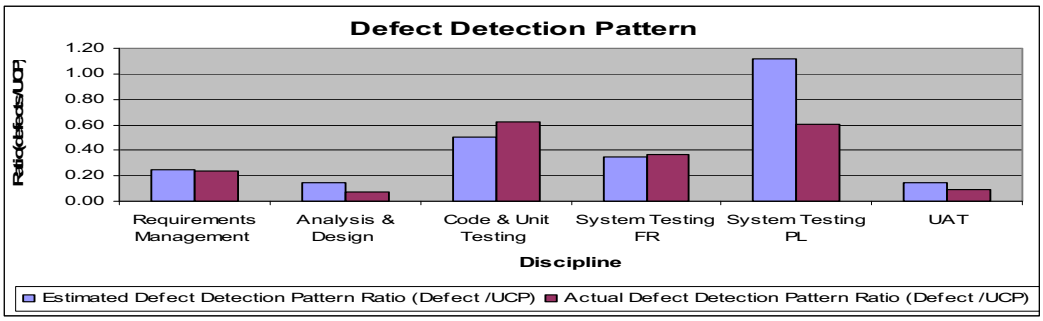
# of Defects (PLs)		Estimated Defect Fix Rate (%)	Actual Defect Fix Rate (%)
Open	Fixed		
195	167	80	85.64



The Estimated Defect Fix Rate is the goal set up in the project initiation phase based on an organizational baseline.

Defect Detection Pattern - Distribution of defects (formal review and testing) per discipline (sample)

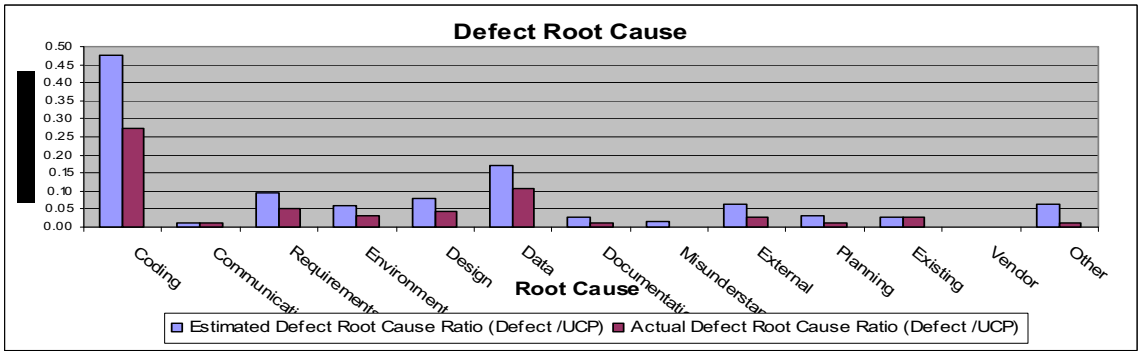
Project Size		189	UCP	
Discipline	Estimated # of Defects (formal review + testing)	Estimated Defect Detection Pattern Ratio (Defect /UCP)	Actual # of Defects (formal review + testing)	Actual Defect Detection Pattern Ratio (Defect /UCP)
Requirements Management	47	0.25	45	0.24
Analysis & Design	28	0.15	14	0.07
Code & Unit Testing	95	0.50	117	0.62
System Testing PR	66	0.35	70	0.37
System Testing PL	211	1.12	114	0.60
UAT	28	0.15	17	0.09



The Estimated Defect Ratio is the goal set up in the project initiation phase using a Defect Detection Model and Monte Carlo Simulation. Data is collected from the project's Tracking System.

Defect Root Cause - Distribution of testing defects per root cause (sample)

Project Size		189	UCP	
Root Cause Category	Estimated # of defects	Estimated Ratio (Defect /UCP)	Actual # of Defects	Actual Ratio (Defect /UCP)
Coding	90	0.48	52	0.28
Communication	2	0.01	2	0.01
Requirements	18	0.10	10	0.05
Environment	11	0.06	6	0.03
Design	15	0.08	8	0.04
Data	32	0.17	20	0.11
Documentation	5	0.03	2	0.01
Misunderstanding	3	0.02	0	0.00
External	12	0.06	5	0.03
Planning	6	0.03	2	0.01
Existing	5	0.03	5	0.03
Vendor	0	0.00	0	0.00
Other	12	0.06	2	0.01



The Estimated Defect Ratio is the goal set up in the project initiation phase using a Defect Detection Model and Monte Carlo Simulation. Data is collected from the project's Tracking System.

4. Analysis and Results

This section contains a description of the analysis from the defect measurement goals to the actual produced during the project.

5. Recommendations

This section contains recommendations derived as a result of the analysis that could prevent the same problems from occurring in the future. The analytical study is done with the purpose to take action on the process that produced the data not on the materials from which the data came.

6. Statistical Testing Defect Analysis Details

This section contains all of the detailed Testing Defects (Problem Logs) information used and analyzed for input into the previous sections. If the project contains multiple phases, this section is broken down by phase.

7. Definitions

This section contains detailed descriptions for data collected. E.g. root causes, disciplines.

Lessons Learned – Do's

- Understand audience needs
- Select metrics based on goals
- Focus on processes, products and services
- Validate data
- Use Tools
- Use size unit to normalize data
- Be concise, flexible
- Monitor reports usage

Lessons Learned – Don'ts

- Report on individuals
- Compare data collected by different methods
- Report invalid data
- Support actions
- Use exclusively contractors for reports creation
- Mask unfavourable results

References:

Humphrey-89 Watts S. Humphrey; Managing the Software Process; Addison-Wesley Publishing Company, Reading, MA; 1989; ISBN 0-201-18095-2
Guideline to Software Measurement release 2 – IFPUG 2004
Measuring the Software Process W Florac & A Carleton – The SEI Series 1999

Author Bio:

Erika Vintan is a Senior Process Improvement Specialist in the Toronto Bank of Montreal (BMO) IT Best Practices and QA CM, RMG & PCG department within Corporate Technology Development.

She has a Master's degree in Science, with certifications as a Software Measurement Specialist, ISO Internal Auditor, and Quality Assurance Analyst Accreditation.

Erika joined the BMO Quality Assurance Management Team eight years ago, supporting an IT staff of approximately 300, with a mission to establish QA Best Practices across the Private Client Group (PCG).
Erika has been a member of the Software Engineering Process Group implementing process improvement models (ISO 9001:2000, ITIL, SEI-CMMI) across PCG.

Erika completed her Statistical Process Control knowledge at the Software Engineering Institute in Pittsburgh and spent her last four years as the Quantitative Management Group representative for PCG co-coordinating the measurement program and defining the framework and context for implementing processes and process improvements that includes supporting a practicing CMMI Level 5 organization.

She is actively involved in the Software Process Improvement community as a steering committee member of the Toronto Software Process Improvement Network group.