

Masterarbeit

---

# Systematisches Werkzeugunterstütztes Anpassen von Metriken

Systematic Tool Supported Tailoring of Metrics

von

**Ricardo Iván Tavizón Ramos**

---

Vorgelegt der: Fakultät für Mathematik, Informatik  
und Naturwissenschaften der Rheinisch-  
Westfälischen Technischen Hochschule  
Aachen im März 2011

Angefertigt am: Lehr- und Forschungsgebiet Informatik 3  
Prof. Dr. rer. nat. Horst Lichter

Gutachter: Prof. Dr. rer. nat. Horst Lichter  
Prof. Dr. Bernhard Rumpe

Betreuer: Dipl.-Inform. Matthias Vianden



Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, den 2. März 2011

---

Ricardo Iván Tavizón Ramos

---



# Contents

<b>1. INTRODUCTION</b>	<b>1</b>
1.1. Objectives . . . . .	2
1.2. Research Methodology . . . . .	3
1.3. Overview . . . . .	4
<b>2. FOUNDATIONS</b>	<b>5</b>
2.1. Preliminaries to Metrics . . . . .	5
2.1.1. Metrics in the Organization . . . . .	5
2.1.2. Measurement Information Model . . . . .	11
2.1.3. Representation of Metrics . . . . .	13
2.2. Metamodeling . . . . .	14
2.2.1. Meta-models . . . . .	14
2.2.2. Metrics at the Meta Level . . . . .	17
2.2.3. MeDIC Tool . . . . .	18
2.3. Reusable Software Component Models . . . . .	21
2.3.1. Library . . . . .	21
2.3.2. Object Oriented Framework . . . . .	22
2.4. Software Product Line Engineering (SPLE) . . . . .	25
2.4.1. Software Product Lines (SPL) . . . . .	25
2.4.2. Feature Modeling . . . . .	27
2.4.3. Variation Management . . . . .	28
2.5. Paper Prototyping . . . . .	29
2.6. Metaphor in Computer Science . . . . .	32
<b>3. RELATED WORK</b>	<b>35</b>
3.1. Organizational Metric Program . . . . .	35
3.2. Metric Repository . . . . .	37
3.3. Metric Framework . . . . .	40
3.4. Discussion . . . . .	42
<b>4. TAILORING CONCEPT</b>	<b>45</b>
4.1. Tailoring . . . . .	45
4.2. Metric Framework Metaphor . . . . .	47
<b>5. CONCEPTUAL APPROACH</b>	<b>51</b>
5.1. Requirements . . . . .	51
5.1.1. Non-Functional Requirements . . . . .	52
5.2. Abstract Interpretation . . . . .	52
5.2.1. Metric Meta-Model . . . . .	54
5.2.2. Metric Framework . . . . .	54

5.2.3. Metric Instances . . . . .	55
5.3. Metric Frame . . . . .	55
5.3.1. Variability Model . . . . .	56
5.3.2. Graphical Representation . . . . .	60
5.4. Tailoring Process . . . . .	60
5.4.1. Example . . . . .	62
<b>6. TOOL SUPPORT</b>	<b>65</b>
6.1. Concept . . . . .	65
6.2. Application Scenario . . . . .	68
6.3. Horizontal Prototype . . . . .	68
6.3.1. Tool Support Motivation . . . . .	68
6.3.2. Tool Mode View . . . . .	76
<b>7. EVALUATION</b>	<b>77</b>
7.1. Concept . . . . .	77
7.2. Tool Support . . . . .	79
7.3. Discussion . . . . .	80
<b>8. SUMMARY AND OUTLOOK</b>	<b>83</b>
8.1. Summary . . . . .	83
8.2. Outlook . . . . .	84
<b>A. Metrics</b>	<b>85</b>
A.1. METRICS - GLOSSARY . . . . .	85
A.2. REPORTING . . . . .	86
<b>BIBLIOGRAPHY</b>	<b>88</b>

## List of Figures

1.1. Tailoring of project specific metrics from the organizational wide defined metrics . . . . .	2
2.1. Metric number of failure reports by status . . . . .	7
2.2. Measurement Information Model . . . . .	10
2.3. Measurement construct of coding progress . . . . .	12
2.4. Type-Instance relationship . . . . .	15
2.5. The core from the metric meta-model . . . . .	19
2.6. Screenshot from the MeDIC Graphical User Interface (GUI) . . . . .	20
2.7. An application making use of many software libraries . . . . .	21
2.8. The inversion of control . . . . .	23
2.9. Hot-spots based on inheritance (Class A) and composition (Class B) . . . . .	24
2.10. Comparison of software product development . . . . .	26
2.11. A feature diagram representing a configurable system . . . . .	27
2.12. Variation management of production line . . . . .	28
2.13. Material used during paper prototyping . . . . .	30
2.14. A hand-drawn paper prototype of the screen from a mobile phone application used to search restaurants nearby the user . . . . .	31
2.15. Metaphors in computer science: A) Garbage Collection and B) Debugging . . . . .	32
3.1. An abstract GQM structure . . . . .	36
3.2. Process maturity examples: A) a repeatable process (level 2), B) a defined process (level 3), C) a managed process (level 4) and D) an optimizing process (level 5) . . . . .	38
3.3. Fact and dimension tables in a dimension model . . . . .	40
3.4. The metric framework and its approach . . . . .	42
4.1. Metric framework metaphor . . . . .	48
5.1. Overview of the abstract interpretation . . . . .	53
5.2. UML component diagram used to represent a metric framework . . . . .	54
5.3. Metric frame structure . . . . .	56
5.4. Variability meta-model . . . . .	57
5.5. Graphical notation for metric frames. A) Variation Point "Measure Interval" and B) Variation Point "Information Need" . . . . .	60
6.1. Creation of the target model: A) Using the target and the intermediate models B) Using only the intermediate model . . . . .	66
6.2. UML Use Case Template for Tailor Metrics . . . . .	67

6.3. BPMN Tailor of Metrics . . . . .	69
6.4. Creation of a metric frame . . . . .	70
6.5. Paper prototype of the Graphical User Interface (GUI) from the tool . . . . .	71
6.6. Box displaying the metric frame <i>Actual Cost (AC)</i> . . . . .	72
6.7. List of variation points and their rationales . . . . .	72
6.8. Wizard from the metric frame <i>Cost Performance Index (CPI)</i> . . . . .	73
6.9. Variation points of type opened . . . . .	74
6.10. Interface used to create a new variant . . . . .	74
6.11. Variation points of type opened . . . . .	75
6.12. Variation point of type closed and alternative choice variability . . . . .	76
A.1. A) A graphic report and B) A tabular report . . . . .	87



## A. Metrics

### Contents

---

A.1. METRICS - GLOSSARY . . . . .	85
A.2. REPORTING . . . . .	86

---

Software organizations collect and disseminate metrics to provide stakeholders with information about how resources are being used to achieve project objectives [Com04]. This process includes:

**Status reporting** describing where the project now stands. For example, status related to schedule and budget metrics.

**Progress reporting** describing what the project team has accomplished. For example, percent complete to schedule, or what is completed versus what is in process.

**Forecasting** predicting future project status and progress. Performance reporting should generally provide information on scope, schedule, cost and quality.

Constructive Cost Model II (COCOMO II), is a model used to estimate the cost, effort and schedule when planning a new software development activity [Boe95].

### A.1. METRICS - GLOSSARY

**Actual Cost (AC).** Total costs incurred that must relate to whatever cost was budgeted within the planned value and earned value (which can sometimes be direct labor hours alone, direct costs alone, or all costs including indirect costs) in accomplishing work during a given time period.

**Budget Estimate.** See estimate.

**Budget at Completion (BAC).** The sum of the total budgets for a project.

**Cost Performance Index (CPI).** The cost efficiency ratio of earned value to actual costs. CPI is often used to predict the magnitude of a possible cost overrun using the following formula:  $BAC/CPI = ProjectedCostAtCompletion$ .  
 $CPI = EV/AV$ .

**Cost Variance (CV).** 1) Any difference between the budgeted cost of an activity

and the actual cost of that activity. 2) In earned value,  $EV - AC = CV$ .

**Duration.** The number of work periods (not including holidays or other nonworking periods) required to complete an activity or other project element. Usually expressed as workdays or workweeks.

**Earned Value (EV).** The physical work accomplished plus the authorized budget for this work. The sum of the approved cost estimates (may include overhead allocation) for activities (or portions of activities) completed during a given period (usually project-to-date).

**Effort.** The number of labor units required to complete an activity or other project element. Usually expressed as staff hours, staff days, or staff weeks. Should not be confused with duration.

**Estimate.** An assessment of the likely quantitative result. Usually applied to project costs and durations and should always include some indication of accuracy (e.g.,  $\pm x$  percent). Usually used with a modifier (e.g., preliminary, conceptual, feasibility). Some application areas have specific modifiers that imply particular accuracy ranges (e.g., order-of-magnitude estimate, budget estimate and definitive estimate in engineering and construction projects).

**Planned Value (PV).** The physical work scheduled, plus the authorized budget to accomplish the scheduled work.

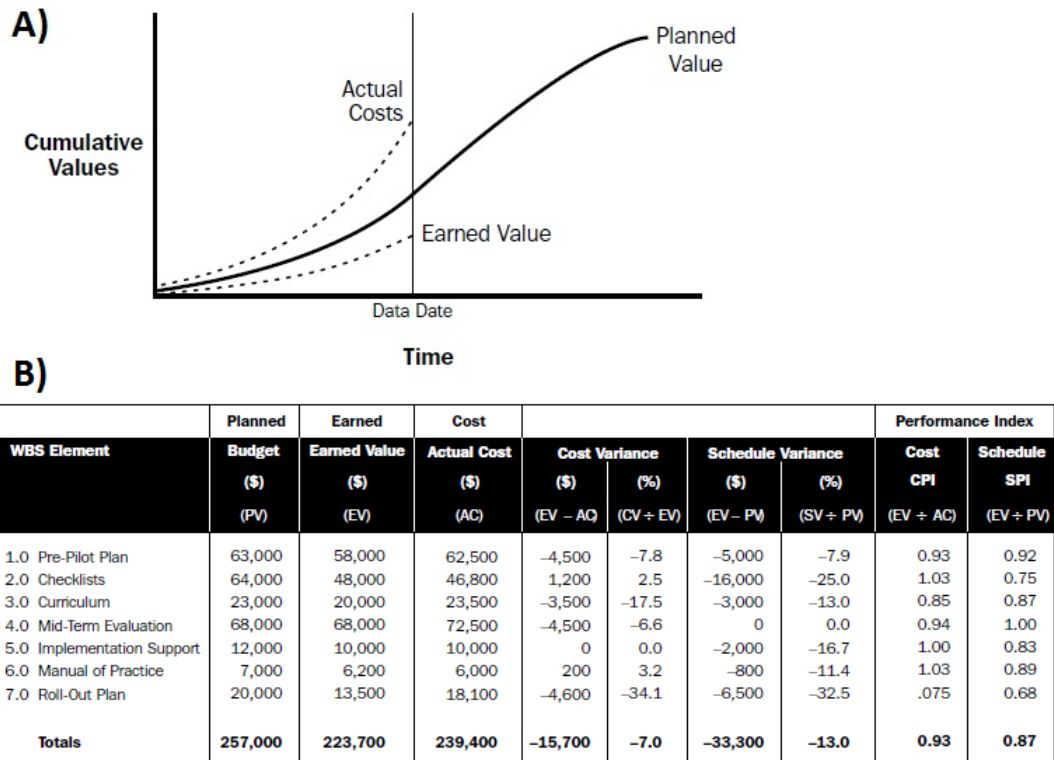
**Schedule Performance Index (SPI).** The schedule efficiency ratio of earned value accomplished against the planned value. The SPI describes what portion of the planned schedule was actually accomplished. The  $SPI = EV/PV$ .

**Schedule Variance (SV).** 1) Any difference between the scheduled completion of an activity and the actual completion of that activity. 2) In earned value,  $EV - PV = SV$ .

## A.2. REPORTING

Software organizations make use of reports to organize and summarize the information gathered and present the results of any analysis. Reports should provide the kinds of information and the level of detail required by various stakeholders.

Common formats for reports include bar charts (also called Gantt charts), S-curves, histograms and tables. Figure A.1 (A) uses S-curves to display cumulative EV analysis data, while Figure A.1 (B) displays a different set of EV data in tabular form [Com04].



Note: All figures are project-to-date.

\*Other units of measure that may be used in these calculations may include: labor hours, cubic yards of concrete, etc.

Figure A.1.: A) A graphic report and B) A tabular report



## Bibliography

- [Apo10] APOSTOLOS, B.: *Business Process Management: A Data Cube To Analyze Business Process Simulation Data For Decision Making*. 2010.
- [BCR94] BASILI, V., G. CALDIERA and D. ROMBACH: *The Goal Question Metric Approach*. 1994.
- [BFOS84] BREIMAN, L., J. H. FRIEDMAN, R. A. OLSHEN and C. J. STONE: *Classification and regression trees*. Wadsworth & Brooks/Cole Advanced Books & Software, 1984.
- [Bha94] BHANDARI, I: *Attribute Focusing: machine-assisted knowledge discovery applied to software production process control*. 1994.
- [Boe95] BOEHM, B.: *COCOMO II*. [http://sunset.usc.edu/csse/research/COCOMOII/cocomo\\_main.html](http://sunset.usc.edu/csse/research/COCOMOII/cocomo_main.html), 1995.
- [Bos98] BOSCH, R.: *An Object-Oriented Framework for Measurement Systems*. 1998.
- [Bos00] BOSCH, R.: *Design & Use of Software Architectures - Adopting and Evolving a Product-Line Approach*. Addison-Wesley, 2000.
- [CCS93] CODD, E.F., S.B. CODD and C.T. SALLEY: *Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate*. 1993.
- [CN02] CLEMENTS, P. and L. NORTHROP: *Software Product Lines, Practices and Patterns*. Addison-Wesley, 2002.
- [Com04] COMMITTEE, SOFTWARE ENGINEERING STANDARDS: *A Guide to the Project Management Body of Knowledge*. 2004.
- [Dav87] DAVIS, S.: *Future Perfect, Reading*. 1987.
- [Dun94] DUNLAVEY: *Building Better Applications: a Theory of Efficient Software Development*. International Thomson Publishing, 1994.
- [Ear60] EARNED VALUE MANAGEMENT SYSTEMS: *Metric Earned Value*. <http://www.earnedvaluemanagement.com/>, 1960.
- [Ecl10] ECLIPSE FOUNDATION: *Eclipse Modeling Framework - EMF*. <http://www.eclipse.org/modeling/emf/>, 2010.
- [GFd98] GRISS, M., J. FAVARO and M. D'ALESSANDRO: *Integrating feature modeling with the RSEB*. 1998.

- [GRCP06] GARCÍA, F., F. RUIZ, J.A. CRUZ and M. PIATTINI: *Gestión Integrada del Modelado y de la Medición del Proceso Software*. 2006.
- [GV02] GAMMA, E., HELM R. JOHNSON R. and J. VLISSIDES: *Design Patterns*. Addison Wesley Professional, 2002.
- [Har04] HARRISON, W.: *A flexible method for maintaining software metrics data: a universal metrics repository*. 2004.
- [ISO08] ISO/IEC 9075-1:2008: *ISO/IEC 9075-1:2008*. [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=45498](http://www.iso.org/iso/catalogue_detail.htm?csnumber=45498), 2008.
- [JF98] JOHNSON and FOOTE: *Designing Reusable Classes*. 1998.
- [JGJ97] JACOBSON, I., M GRISS and P. JONSSON: *Software Reuse: Architecture, Process and Organization for Business Success*. Addison-Wesley Professional, 1997.
- [Joi00] JOINT TECHNICAL SUBCOMMITTEE: ISO/IEC: *ISO/IEC 15504*. <http://www.iso15504.es/>, 2000.
- [KCH<sup>+</sup>90] KANG, K., S. COHEN, J. HESS, W. NOVAK and A. PETERSON: *Feasibility Study Feature-Oriented Domain Analysis ( FODA )*. 1990.
- [KKL<sup>+</sup>98] KANG, K., S. KIM, J. LEE, K. KIM, G KIM and E. SHIN: *FORM: A Feature-Oriented Reuse Method with Domain-Specific Reference Architectures*. 1998.
- [KR02] KIMBALL, R. and M. ROSS: *The Data Warehouse Toolkit 2nd. Edition*. Wiley & Sons, 2002.
- [Kru01] KRUEGER, C.: *Variation Management for Software Production Lines*. 2001.
- [Lic07] LICHTER, H.: *Software Engineering - Grundlagen, Menschen, Prozesse, Techniken*. Morgan Kaufmann Publishers, 2007.
- [Lic08] LICHTER, H.: *Object Oriented Software Construction, Lecture Scripts*. 2008.
- [LL10] LUDEWIG, J. and H. LICHTER: *Software Engineering*. dpunkt.Verlag GmbH, 2010.
- [LMW<sup>+</sup>90] LANDIS, L., F. MCGARRY, S. WALIGORA, R. PAJERSKI, M. STARK, R. KESTER, T. MCDERMOTT and J. MILLER: *Manager 's Handbook for Software Development Revision 1*. 1990.
- [MB00] MENDONSA, M.G. and V.R. BASILI: *Validation of an approach for improving existing measurement frameworks*. 2000.
- [McG01] MCGARRY, J.: *Practical Software Measurement: Objective Information for Decision Makers*. Addison-Wesley, 2001.

- [Mey00] MEYER, B.: *Object-Oriented Software Construction, Second Edition*. Prentice Hall, 2000.
- [Obj09] OBJECT MANAGEMENT GROUP: *Business Process Management Notation (BPMN)*. <http://www.bpmn.org/>, 2009.
- [Obj10] OBJECT MANAGEMENT GROUP: *Unified Modeling Language (UML)*. <http://www.uml.org/>, 2010.
- [OJ97] OFFEN, R.J. and R. JEFFERY: *Establishing software measurement programs*. 1997.
- [Opp09] OPPERMAN, R.: *User-Oriented System Design and Personalized Information Services, Laboratory*. 2009.
- [Ora10] ORACLE TECHNOLOGY WORK: *Enterprise JavaBeans Technology - EJB*. <http://www.oracle.com/technetwork/java/javaee/ejb/index.html>, 2010.
- [PBvdL05] POHL, K., G. BÖCKLE and F. VAN DER LINDEN: *Software Product Line Engineering*. Springer-Verlag, 2005.
- [PFA03] PALZA, E., C. FUHRMAN and A. ABRAN: *Establishing a generic and multidimensional measurement repository in CMMI context*. 2003.
- [Pfl93] PFLEEGER, S.L.: *Lessons learned in building a corporate metrics program*. 1993.
- [Pre97] PREE, W.: *Komponentenbasierte Softwareentwicklung mit Frameworks*. dpunkt.Verlag Heidelberg GmbH, 1997.
- [SD06] SINNEMA, M. and S. DEELSTRA: *Modeling Quality Attributes in Product Families with COVAMOF*. 2006.
- [Sny03] SNYDER, C.: *Paper Prototyping*. Morgan Kaufmann Publishers, 2003.
- [Sof10] SOFTWARE ENGINEERING INSTITUTE: *Capability Maturity Model Integration (CMMI)*. <http://www.sei.cmu.edu/cmmi/>, 2010.
- [Sta05] STANDARD, INTERNATIONAL: *IEEE Std 1061-1998 (R2004), Inc. IEEE Standard for a Software Quality Metrics Methodology*. 2005.
- [Sta07] STANDARD, INTERNATIONAL: *ISO/IEC 15939*. 2007.
- [The95] THE STANDISH GROUP: *The Standish Group research study "CHAOS"*. <http://www.projectsmart.co.uk/docs/chaos-report.pdf>, 1995.
- [VHLN08] VIANDEN, M., V. HOFFMANN, H. LICHTER and K-J NEUMANN: *Ein metamodel basierter Ansatz zur Metrikdefinition und Dokumentation Hintergrund und Motivation*. 2008.