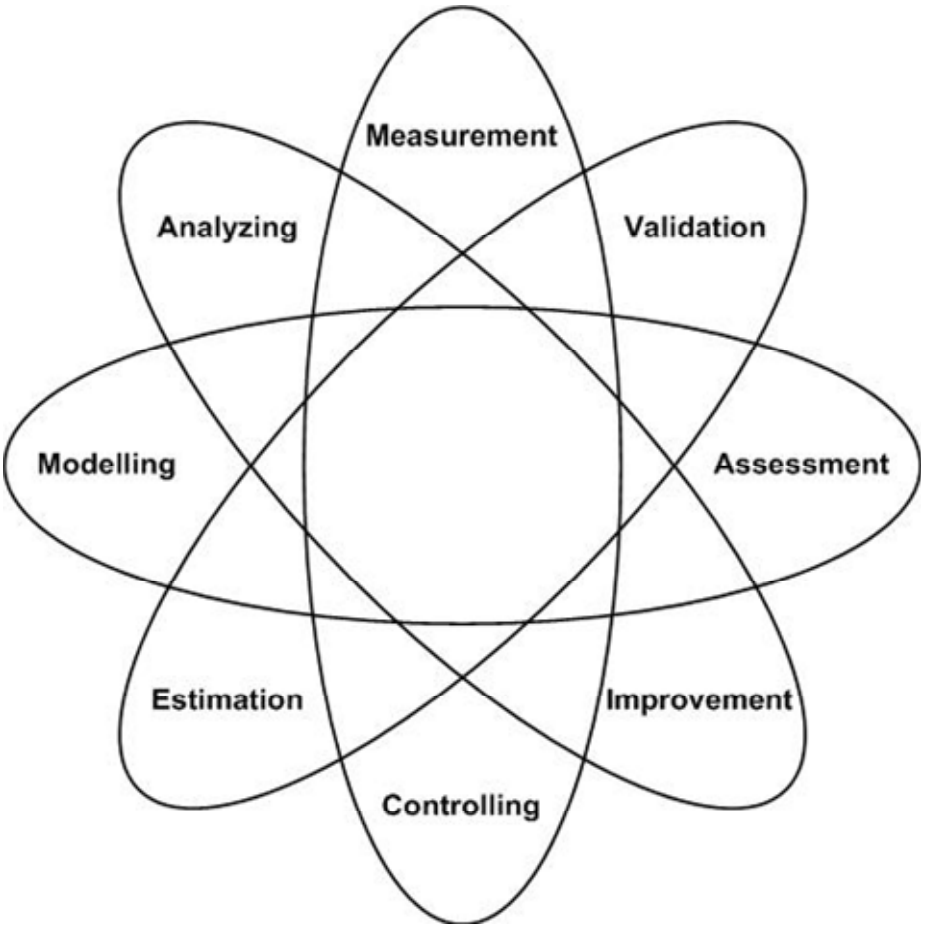
Volume 15, Number 2, August 2010

SOFTWARE MEASUREMENT NEWS

Journal of the Software Metrics Community



Editors:

Alain Abran, Günter Büren, Reiner Dumke, Christof Ebert, Horst Zuse

Université du Québec
École de technologie supérieure

The *SOFTWARE MEASUREMENT NEWS* can be ordered directly from the Editorial Office (address can be found below).

Editors:

Alain Abran
Professor and Director of the Research Lab. in Software Engineering Management
École de Technologie Supérieure - ETS
1100 Notre-Dame Quest,
Montréal, Quebec, H3C 1K3, Canada
***Tel.:** +1-514-396-8632, **Fax:** +1-514-396-8684*
aabran@ele.etsmtl.ca

Günter Büren
Vice Chair of the DASMA
Büren & Partner Software-Design GbR
Oedenberger Str. 155, D-90491 Nürnberg, Germany
***Tel.:** +49-911-5195511, **Fax:** +49-911-5195555*
gb@bup-nbg.de
http://www.dasma.org

Reiner Dumke
Professor on Software Engineering
University of Magdeburg, FIN/IVS
Postfach 4120, D-39016 Magdeburg, Germany
***Tel.:** +49-391-67-18664, **Fax:** +49-391-67-12810*
dumke@ivs.cs.uni-magdeburg.de

Christof Ebert
Dr.-Ing. in Computer Science
Vector Consulting GmbH
Ingersheimer Str. 24, D-70499 Stuttgart, Germany
***Tel.:** +49-711-80670-175*
christof.ebert@vector-consulting.de

Horst Zuse
Dr.-Ing. habil. in Computer Science
Technical University of Berlin, FR 5-3,
Franklinstr. 28/29, D-10587 Berlin, Germany
***Tel.:** +49-30-314-73439, **Fax:** +49-30-314-21103*
zuse@tubvm.cs.tu-berlin.de

Editorial Office: Otto-von-Guericke-University of Magdeburg, FIN/IVS, Postfach 4120, 39016 Magdeburg, Germany

Technical Editor: Dagmar Dörge

Call for Participation

Aufruf zur Teilnahme

IWSM/MetriKon/MENSURA 2010

10. – 12. November 2010 in Stuttgart
(www.metrikon.de)

– Presentations –

(see the detailed program in our Web links)

Keynotes:

Dr. Christiof Ebert: *Lean Development – Measuring and Improving Performance*

Dr. Pekka Abrahamsson: *The Art of Measurement in Agile Development*

Wai F. Hom: *Metrics @ IBM Corporation*

New Measurement Appraoches for Business-Applications:

Luigi Buglione (Engineering.IT, Italy): *Suggestions for Improving Measurement Plans: a BMP application in Italy*

Khalid T. Al-Sarayreh (University of Quebec (ETS), Montreal, Canada): *Specification and Measurement of System Configuration Non Functional Requirements*

Kosta Pandazo (IT University of Copenhagen, Danemark): *Presenting Software Metrics Indicators - A Case Study*

Monica Villavicencio (ETS Montreal, Canada): *Software Measurement in Software Engineering Education: A Comparative Analysis*

Filippo De Carli (Telecom Italia group, Italy): *YES! We measure, a practical metrics experience.*

Martin Kowalczyk (IESE Kaiserslautern, Germany): *Aligning Software-related Activities of Different Organizations with Common Business Goals*

Alain Abran (University of Quebec): *Measurement of Software Requirements Derived from System Operations Requirements*

New Approaches for Quality Assurance:

Konstantina Georgieva (Uni Magdeburg, Germany): *Applying Human Error Assessment and Reduction Technique (HEART) in the software development process*

Adam Trendowicz (IESE Kaiserslautern, Germany): *Model-based Product Quality Evaluation with Multi-Criteria Decision Analysis*

Markus Großmann (Capgemini sd&m AG, Germany): *Entwicklung eines operationalisierten Qualitätsmodells für SAP-Integrationsprojekte: Ergebnis der ersten Iteration*

Benoît Vanderose (University of Namur, Belgium): *Towards a Model-Centric Quality Assessment*

Michael Kläs (IESE Kaiserslautern, Germany): *How-To Evaluate Meta-Models for Software Quality?*

Frank Simon (SQS, Köln, Germany): *Aktives Risikomanagement mit Escrow-Services*

Ralf Russ (Siemens Corporate Technology, Germany): *Identify Quality Issues by systematic quantitative Analysis*

Harry Sneed (FH Hagenberg, Germany): *Werterhaltung von Software durch evolutionäre Qualitätssicherung*

Nguyen Duc Anh (IESE Kaiserslautern, Germany): *The Impact of Design Complexity on Software Quality - A Meta Analysis*

Harald Gruber (Universität Linz, Austria): *On the validity of benchmarking for evaluating code quality*

Olga Ormandjieva (Concordia University Montreal, Canada): *Categorical Representation of Software Services Quality and Decision-Making Using Markov Decision Process*

Improvements for Cost/Effort Estimation:

Luigi Buglione (Engineering.IT Rome, Italy): *Which COSMIC Base Functional Components are Significant in Estimating Web Application Development? - An Empirical Study*

Carlos Monsalve (ETS Montréal, Canada): *Functional Size Measurement with Business Process Models: the Business Application Domain*

Özden Özcan Top (Middle East Technical University, Turkey): *Comparison of External Software Benchmark Data Sets from Effort Prediction Perspective*

Luigi Lavazza (Università dell'Insubria, Italy) : *A Systematic Approach to the Analysis of Function Point - COSMIC Convertibility*

Jean-Marc Desharnais and Bogaziçi (University, Istanbul, Turkey): *Assessment of the Quality of the Documenbtation of Functional User Requeirements Using Criteria derived from Using COSMIC ISO 19761*

Cornelius Wille (FH Bingen, Germany*): Evidenz-basierte Klassifikation und Bewertung von Aufwandschätzmethoden*

Baris Ozkan (Middle East Technical University, Turkey): *Estimating Software Effort in a change-intensive software applications environment: An Experience Report*

Thomas Fehlmann (Euro Project Office AG Zürich, Switzerland): *From Story Points to COSMIC Function Points in Agile Development – A Six Sigma perspective*

Sophie Stern (RENAULT, France): *Estimating the Memory Size of Embedded Software Using COSMIC Function Points - An Empirical Study*

Erdir Ungan (Middle East Technical University, Turkey): *Evaluation of Reliability Improvements for COSMIC Size Measurement Results*

New Metrics Analysis and Applications:

Bernhard Peischl (Technische Universität Graz, Austria): *Integration von Prozess-, Modell- und Produktmetriken*

Luigi Buglione (Engineering.IT Rome, Italy): *Measure well, not ramdonly!*

Harry Sneed (University of Regensburg, Germany): *Design Metrics for UML Models*

Akito Monden (NAIST, Nara, Japan*): On Building a Better Program Size Measure*

J. J. Cuadrado-Gallego (Alcala University, Madrid, Spain): *A Review and Fundamental Issues of Lines of Code Metrics*

Measurement and Evaluation of modern Paradigms and Architectures:

Florian Muhß (Uni Magdeburg, Germany): *Software Engineering meets Cloud Computing - How Design for Six Sigma can help*

Andreas Schmietendorf (HWR Berlin, Germany): *COSMIC and SOA-Sizing A critical analysis and proposals for improvement potentials*

Hassan Soubra (Renault-Direction de l'Electronique, France): *Functional size measurement of real-time embedded software: procedures and practical steps*

Masateru Tsunoda (NAIST, Nara, Japan): *An Empirical Analysis of Information Technology Operations Cost*

Martin Hobelsberger (FH Regensburg, Germany): *An Experience-Based Repository of Reusable Components for an Component-Based Automotive Software System*

Sebastian Bress (Uni Magdeburg, Germany): *Kostenmetriken für virtuelle Rechnerressourcen - Eine Erweiterung des Frameworks zur Schätzung des Wertes von Cloud Computing*

Measurement and Evaluatuion in Agile Development:

André Janus (Janus–IT Consulting Stuttgart, Germany): *Qualitätssicherung in der Agilen Software-Entwicklung*

Sylvie Trudel (Pyxis Technologies Québec, Canada): *Guideline for sizing Agile projects with COSMIC*

Enrico Berardi (TRS SPA, Rome, Italy): *COSMIC-based Project Management in Agile Software Projects and Mapping onto related CMMI-DEV process areas*

Ankündigung des 5. Workshops „Bewertungsaspekte serviceorientierter Architekturen“

November 2010 Karlsruhe (Gastgeber: FZI Karlsruhe)

MOTIVATION

Moderne Integrationsarchitekturen gelten als „agiler“ Schlüssel bei der erfolgreichen Implementierung von neuen Softwarelösungen. Obwohl vielfältige technologische Ansätze für die Implementierung derartiger Lösungen zur Verfügung stehen, treten bei korrespondierenden Entwicklungs-, Einführungs- und Betriebsprojekten nach wie vor komplexe Problemstellungen auf, die einer nachhaltigen Lösung bedürfen. Die BSOA-Initiative greift die Bewertung dieser Herausforderung auf und führt dazu jährlich einen entsprechenden Workshop durch. Der kommende Workshop wird sich insbesondere mit den folgenden Themen auseinandersetzen:

- Aufwands- und Risikobetrachtungen bei Integrationsprojekten,
- Veränderte Anforderungen an eine modellgetriebene Servicekomposition,
- Implikationen bei der Auswahl marktgängiger Serviceangebote,
- Wechselwirkungen zu Themen wie Clouds oder Mashups,
- Erfahrungen beim Management von Integrationsprojekten.

Selbstverständlich geben die dargestellten Themen nur einen ausgewählten Teil möglicher Herausforderungen bei der Bewertung serviceorientierter Architekturen wieder. Dem entsprechend dienen diese der Orientierung und nicht der Einschränkung für potentielle Beiträge.

WORKSHOP-BEITRÄGE

Praktiker und Wissenschaftler, die auf dem Gebiet der Konzeption, Entwicklung und Management moderner Integrationsarchitekturen tätig sind, werden gebeten, Beiträge im doc- oder pdf-Format über die Webseite einzureichen. Der Umfang der Beiträge sollte 3000 Wörter nicht übersteigen. Die Formatierungsrichtlinien werden auf der unten genannten Webseite veröffentlicht. Angenommene Beiträge werden innerhalb eines 30-minütigen Vortrags präsentiert bzw. in Form eines Posters (innerhalb der Workshoppausen) vorgestellt. Alle angenommenen Beiträge des Workshops erscheinen in einem Tagungsband.

PROGRAMMKOMITEE

S. Aier, Universität St. Gallen
E. Dimitrov, T-Systems
T. Grawe, Advicio
M. Irtmann, IBM
G. Limböck, SAP
R. Molle, ITAB Hamburg
H. Pundt, HS Harz
J. Schuck, MATERNA GmbH
F. Balzer, CA Deutschland
R. Dumke, Universität Magdeburg
J. M. Gomez, Uni Oldenburg
B. Klöppel, T-Systems
M. Lother, Robert Bosch GmbH
S. Nakonz, Bitnologie
M. Rothaut, Deutsche Telekom
F. Victor, FH Köln
R. Zarnekow, TU Berlin
J. vom Brocke, HS Liechtenstein
M. Fiedler, Software AG
W. Greis, TPS Data & CECMG
S. Kosterski, Toll Collect
M. Mevius, FZI & HTWG Konstanz
S. Patig, Universität Bern
A. Schmietendorf, HWR Berlin
C. Wille, FH Bingen

TERMINE

12.09.2010	Einreichung von Beiträgen
04.10.2010	Annahme/Ablehnung
08.10.2010	finales Workshop-Programm
18.10.2010	Abgabe der druckreifen Beiträge
November 2010	Workshop in Darmstadt

WEBSEITE ZUM WORKSHOP

`http://ivs.cs.uni-magdeburg.de/~gi-bsoa`

Fachhochschule für Wirtschaft Berlin
Berlin School of Economics

Diplomarbeiten-Preis 2010

für studentische Arbeiten aus den Bereichen
Software-Metriken und Aufwandschätzung

Ausschreibung

DASMA e.V.
Deutschsprachige Anwendergruppe für Software-Metrik und Aufwandschätzung e.V.

Kontakt DASMA

DASMA e.V.
c/o Frau Katja Sons
Am Poth 25
40625 Düsseldorf
E-Mail: info@dasma.org
Internet: www.dasma.org

Kontakt
DASMA Diplomarbeiten-Preis
Prof. Dr.-Ing. Cornelius Wille
E-Mail: wille@fh-bingen.de
Telefon: 06721 409-257

Preiskomitee

Manfred Bundschuh
DASMA e.V.
Günter Büren
Büren & Partner Softwaredesign, Nürnberg
Prof. Dr.-Ing. habil. Reiner Dumke
Universität Magdeburg
Dr. Ing. Christof Ebert
Vector Consulting, Stuttgart
Prof. Dr. Hans-Georg Hopf
GSO-Fachhochschule Nürnberg
Dr.-Ing. Marek Leszak (Vorsitz)
Alcatel-Lucent, Nürnberg
Dr. Jürgen Münch
Fraunhofer IESE, Kaiserslautern
Stavros Pechlivanidis
IBM Deutschland GmbH, Köln
Prof. Dr. Andreas Schmietendorf
Hochschule für Wirtschaft Berlin
Manfred Seufert
MediaanABS Deutschland GmbH, Düsseldorf
Prof. Dr.-Ing. Cornelius Wille
Fachhochschule Bingen
Prof. Dr.-Ing. Horst Zuse
Technische Universität Berlin
+ angefragt

Termine

25. September 2010
Abgabeschluss
25. Oktober 2010 (spätestens)
Benachrichtigung des Preisträgers
11. November 2010
Preisverleihung auf der MetriKon 2010

Software-Metriken sind eine der Schlüsseltechniken für das Management und die Führung von Software-Entwicklungsprozessen. Die praktische Anwendbarkeit von Metriken und die Effektivität von Metrik-Programmen ist dabei immer wieder Gegenstand von engagierten Diskussionen und wissenschaftlichen Betrachtungen. Die DASMA verfolgt das Ziel, diese Diskussionen aufzugreifen und damit den Erfahrungsaustausch über die Anwendung von Metriken zu verbreiten.

Mit der Ausschreibung eines Diplomarbeiten-Preises möchte die DASMA hervorragende Arbeiten auf dem Gebiet "Software-Metriken und Aufwandschätzung" fördern.

Der DASMA Diplomarbeiten-Preis ist mit **1000 €** dotiert und wird 2010 bereits zum achten Mal verliehen.

Auswahlkriterien

Kriterien für die Bewertung der eingereichten Arbeiten sind Innovationsgehalt, praktische Umsetzbarkeit, wissenschaftliche Methodik und Klarheit der Darstellung.

Preiskomitee

Das Preiskomitee wird vom Vorstand der DASMA bestellt und setzt sich aus DASMA-Vorstandsmitgliedern, Hochschullehrern, Industrievertretern und Sponsoren des Preises zusammen. Bei Bedarf können weitere Gutachter beauftragt werden.

Themenkreis

Generell können Arbeiten zu jeglichen Themen rund um Software-Metriken und Aufwandschätzverfahren eingereicht werden. Zur Orientierung hier eine Auswahl typischer Themen:

- Software-Metriken, Vergleich von Metriken etc.
- Einführung von Metriken, Metriken als "Erfolgsfaktoren"
- Einsatz, Einführung und Erfahrungen mit Aufwandschätzverfahren und -Tools
- Einsatz von Metriken/Aufwandschätzverfahren im Zusammenhang mit Embedded Systems, System Engineering, Integration & Test, Web-Anwendungen, Data Warehouse, OO, etc.
- Quantitatives Projektmanagement, Projektcontrolling, Benchmarking
- Metriken für Prozessverbesserung und Effizienzsteigerung
- Neue Ansätze für den Einsatz von Metriken im IT-Bereich

Teilnahmebedingungen

Zugelassen sind abgeschlossene Diplom-, Master-, Magister- oder Bachelor-Arbeiten an einer deutschsprachigen Hoch- oder Fachhochschule, die sich mit dem o.g. Themenkreis befassen und im Zeitraum **August 2009 – Juli 2010** fertiggestellt worden sind. Die Einreichung muss vom **Betreuer der Arbeit** erfolgen.

Eingereicht werden müssen:

- die vollständige Arbeit in deutscher oder englischer Sprache,
- eine max. 5-seitige Kurzfassung in deutscher Sprache,
- Nachweis der Fertigstellung und erfolgten Benotung der Arbeit,
- Begründung warum diese Arbeit exzellent ist und als preiswürdig empfohlen wird,
- Begleitschreiben mit folgenden Informationen: Name, Vorname, Anschrift, Telefonnummer, E-Mail-Adresse, Titel der Arbeit, Betreuer und Hochschule

Alle Unterlagen müssen in elektronischer Form (PDF) an wille@fh-bingen.de und mleszak@alcatel-lucent.com eingereicht werden und verbleiben bei der DASMA.

Mit der Abgabe der Diplomarbeit stimmt der/die Verfasser/in einer Veröffentlichung der Arbeit (auch auszugsweise) auf der DASMA-Homepage zu. Eine weitere Publikation über andere Medien findet seitens der DASMA nicht statt.

Der/die Preisträger/in verpflichtet sich, auf der **MetriKon 2010** in Stuttgart am **11. November 2010** den Preis persönlich entgegenzunehmen und die Arbeit in einem ca. 30-minütigen Vortrag zu präsentieren. Der Vortrag und die Kurzfassung werden in den MetriKon-Tagungsband aufgenommen. Reisekosten (Bahn 2. Klasse) und Unterkunftskosten (eine Übernachtung) trägt die DASMA. Die DASMA behält sich vor, bei Nichterscheinen den Preis einzubehalten.

Die Entscheidung über die Preiszuerkennung wird unter Ausschluss des Rechtsweges getroffen. Eine Aufteilung des Preises auf mehrere Preisträger ist gegebenenfalls möglich. Das Preiskomitee kann von einer Preiszuerkennung absehen, falls keine geeigneten Arbeiten eingereicht werden.

Weitere Informationen

Alle Informationen zur **DASMA e.V.**, zur **MetriKon 2010** und zum **DASMA Diplomarbeiten-Preis** finden Sie unter www.dasma.org. Im Mitgliederbereich sind **alle eingereichten Diplomarbeiten** der vorjährigen MetriKon Events zugänglich.

COSMIC News

www.cosmicon.com

April 2010 Volume.6 Issue.1

In this Edition:

CSC reports successful use of COSMIC in India

CSC, one of the world's largest IT services companies, cites **'ease of use, estimation accuracy and client appreciation'** as the major benefits of adopting the COSMIC method, which it introduced in 2006

Dr. Somasundaram Muralidharan, director of Quality and Technical Training in India, said that CSC, a long-time user of IFPUG FPA, is now using COSMIC for development, enhancement and maintenance projects for sizing Java and main-frame software.

COSMIC, along with IFPUG FPA, is mandated for use in one of the CSC business groups Already over 30 projects use the method with many professionals able to interpret it properly and consistently. Project managers and SMEs find COSMIC an easier sizing technique.

Over 300 professionals in CSC have been trained in the method in India and 100 senior professionals have gained certification. The policy of encouraging people to become certified aims to ensure broad use of the method. CSC has more than 15 trainers on COSMIC at four locations in India.

Dr Muralidharan praised the COSMIC manual as providing '**clear, well-defined principles and rules which are easy to understand and implement'.** He would however like to see 'more case studies and experience summaries shared across the community'.

CSC uses the method for developing prediction models at the project level as well as at the organization level, resulting in **'the estimated effort being very close to the actual effort'**. Some CSC clients have also expressed appreciation of the method.

CSC works in many industries including financial services, healthcare, automotive, energy, manufacturing and the public sector. Globally, it has 92,000 employees, of which16,000 are in India. 2009 revenue was $16.0 billion. In 2010, CSC was recognized by FORTUNE magazine as one of the World's Most Admired Companies for IT Services.

IEEE Study Group on a 'Requirements Definition Language' includes COSMIC

The IEEE of the USA, probably the world's leading professional body for all aspects of computing, has set up a study group to define a Requirements Definition Language that will enable automatic measurement of functional sizes according to the IFPUG and COSMIC methods.

We are particularly pleased that the Study Group decided to include the COSMIC method within its scope without any reference to us. This is a good indication that some people 'out there' think it is important.

The Study Group operates 'virtually' over the web. Alain Abran, Luigi Buglione and Gerhard Ungerer have joined this group.

This development complements work already underway in universities in Canada, France, Poland and Spain to automate COSMIC sizing from requirements specified in UML in commercial CASE tools.

Renault France adopts COSMIC for embedded software

Use of COSMIC to construct size/cost models for embedded software in the auto industry

At the IWSM Conference in Amsterdam last November, Sophie Stern of Renault presented a paper on her company's experience of using the COSMIC method to help control the price/ performance of its many suppliers of 'Electronic Control Units' (ECU's).

Many functions of a modern car such as Multimedia, Connectivity with the outside world, Advanced Driving Assist Systems, etc are controlled by ECU's which are hardware devices containing embedded software.

ECU's account for an important and increasing cost of developing a car. Hence Renault's embedded software group worked on metrics to control the cost of the software content of an ECU.

Now this group measures the functional size of software in ECU's and uses the information to construct size/cost models. These are used by Purchasing and Cost Analysis departments to predict the cost of new requirements and to compare them with suppliers' prices.

This sounds straightforward, but to achieve the benefits now being obtained, the software group has had to work with Purchasing and Supplier staff to gain their understanding and to change processes.

For more, download Sophie Stern's paper from www.cosmicon.com

Renault is one of the world's largest car and truck manufacturers, producing over 2.3 million units with revenues of over 33 Billion Euros (2009).

COSMIC to size 'Non-Functional' Requirements of Space Systems?

'Non-functional' requirements in Space Systems can be sized using COSMIC

The European Space Agency and IEEE publish standards for real-time and embedded software in space systems which define various 'non-functional' requirements, such as for interfaces, maintainability, etc. In fact these requirements although described as 'non-functional' are in many cases implemented in software.

For example the 'interface' requirements define what is needed to allow:

- a human user to interact with the system
- communication via particular network protocols, for example
- communication with other software that is not within the system to be designed e.g. operating system, files, database management system, or other applications software
- support for the hardware and the specific hardware configuration by the system (i.e. logical structure, physical address, and the expected behavior).

Khalid Al-Sarayreh and Alain Abran have started to produce a series of papers on how to size these various 'non-functional' requirements standards.

For more, contact kahlid.al-sarayreh.1@ens.etsmtl.ca

COSMIC sizing for Agile methods

Who needs 'Story Points'?

Sylvie Trudel and Luigi Buglione have started to develop a Guideline for the use of the COSMIC method for sizing and estimating in software projects following 'agile' life-cycles.

In the interim, Grant Rule has published a short note on 'Sizing Agile Stories using COSMIC'. His simple approach avoids the use of non-standard 'Story Points', thus enabling proper use of standard functional size measurements for performance measurement and estimating. Find this paper on www.cosmicon.com, under Downloads.

COSMIC makes great progress in many areas

COSMIC Progress:

- *Benchmark Data*
- *Web Services*
- *Translations*
- *Users*
- *Guidelines*
- *Tools*

In the nine months since the last issue of 'COSMIC News', we have made tremendous progress in providing all the support facilities and services needed by COSMIC method users.

Benchmarking, We now have comprehensive benchmark data (productivity, speed, etc) obtained from COSMIC-measured new development and enhancement projects submitted to the International Software Benchmarking Standards Group.

Visit www.isbsg.org to get the report 'The Performance of Real-Time, Infrastructure & Business Applications Software Projects: an Analysis of COSMIC-measured Projects in the ISBSG Database'.

The raw data on over 350 projects is also available which you can analyse for your own purposes.

The 'COSMICON' website is now fully operational with much more information available than hitherto. DO VISIT www.cosmicon.com.

Translations. Translations of the Measurement Manual, are now available in Arabic, Chinese, Dutch, French, Japanese and Spanish. We expect translations into German, Italian, Polish and Turkish to become available in 2010.

Method Users On the 'COSMICON' website you will find a list of 40 major organizations from 17 countries in the private and public sectors that use the method to size software in the business application and real-time domain. N.B. half these users are software houses. (Do they know something that their customers have not yet recognised?)

Guidelines. The purpose of a Guideline is to supplement the principles and rules in the Measurement Manual with additional guidance and examples for specific domains.

In 2009 we published a 'Guideline on sizing Data Warehouse Software.' Now we have published a 'Guideline on sizing SOA Software'. Work has started on Guidelines for sizing Real-time System Software and for using COSMIC in an Agile environment. For more see www.cosmicon.com.

Note that for all these Guidelines on how to apply the method to measure software from specific domains, we do not need any new Principles or Rules. The underlying measurement method is stable and hence 'future-proof'.

Tools Simple tools for recording and reporting COSMIC measurements are now available on 'cosmicon'.

Forthcoming Books featuring COSMIC

'Software Metrics and Software Metrology' by Alain Abran will be published mid-year.

'COSMIC Function Points: Theory and Advanced Practices' by Reiner Dumke and Alain Abran, to be published Sept 25th (re-prints of key conference and research papers.)

'Project Estimation Handbook' A new edition to be published by the ISBSG by mid-September, contains a new chapter and case study on COSMIC sizing. See www.isbsg.org

Free Offer of MeterIT-COSMIC to students

MeterIT-COSMIC, a refined tool for capturing COSMIC size measurements is offered free-of-charge to students for the duration of their academic projects. For information, (supervisors, please) contact sales@telmaco.co.uk."

The COSMIC organization

The COSMIC organization is structured into two different bodies: the International Advisory Committee (IAC) of 22 members from 15 countries and the Measurement Practices Committee (MPC).

The COSMICON web-site, **www.cosmicon.com**, is kept up to date and describes the COSMIC organization. It also provides complete background data on functional size measurement, FSM methods, etc.

Further Information

If you have any questions or require further information on COSMIC, please contact your national representative on the COSMIC International Advisory Committee (see www.cosmicon.com, IAC). If you would like to publish an article in this newsletter relating your experience with COSMIC, please forward a draft to the editor via cr.symons@btinternet.com

Metrics Support in Industrial CASE Tools

Hashem Yazbek

University of Magdeburg, Dept. of Computer Science, P.O. Box 4120,
D-39016 Magdeburg, Germany,
hashem@ivs.cs.uni-magdeburg.de

Abstract. *The benefits of software measurement and its importance to the industry are generally unquestioned. A wide variety of powerful software measurement programs are available to customers and to the software industry as a whole. However, metrics and metrics tools are still pretty unused in most software companies. Software measurement is beneficial and relatively simple, but not all benefits of measurement being realized.*
One reason is that many senior engineers and managers are still unaware of the software measurement and its benefits or they have misconception about the effective use of software measurement and they think that software measurement requires special training.
Another reason is that, software measurement is still not recognized as part of the software development process and therefore considered as time-consuming and inevitably delays the completion of a software system. Therefore the company may have to sacrifice software measurement activities if the project is under-budgeted or planned with an unrealistic delivery schedule.
To bear in mind that good metrics tools are still expensive for small and middle companies.
This paper discusses the measurement support in the context of an integrated CASE (I-CASE) environment. For example, how efficient and mature are the existing metric tools? What are the features of a metric extension that enable product and process improvement?
Furthermore, we have considered the software development processes using these tools in three industrial environments in the south of Germany that we don't describe explicitly because of confidential reasons.

Keywords: *CASE Tools, CAME Tools, Quality Assurance, Metrics Tools, Metrics, Metric Repository.*

1 Introduction

Today, CASE-Tools and Software metrics tools play an important role in Software development. It is not possible today to create large software systems without computer support. CASE-Tools are a set of software that supports software process to create high-quality software products.

On the other hand, software companies created in their organization an individual set of activities that will help ensure that every software product exhibits high quality. One of these activities is software measurement. Software measurement helps us to understand more about software products and software processes.

Within the literature we found over 2000 software metrics. With this large amount of metrics it is impossible to evaluate the software quality manually, particularly for large projects. For this reason, tool supports is necessary. There exist many metrics tools

which realize and support the software measurement process and implement many different metrics [Dumke 2003] and [Testingfaqs 2010].

2 Software Measurement Tools

The Tools that support the software measurement are defined as **C**omputer **A**ssisted Software **M**easurement and **E**valuation Tools (CAME tools) [Dumke 1996] [Ebert 2007]. The following figure shows a classification of CAME-Tools by Dumke.

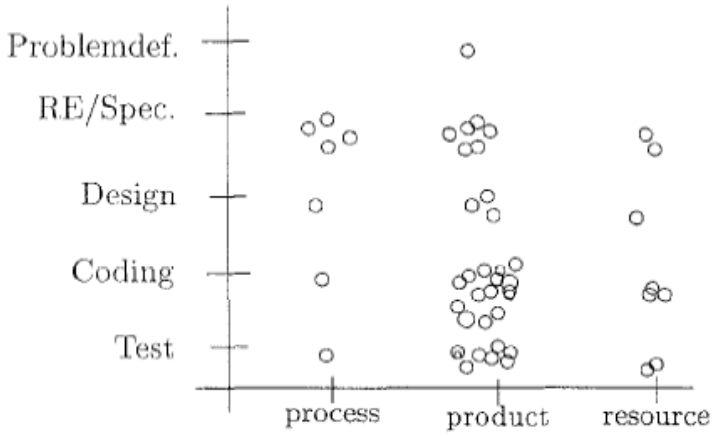


Figure 1: CAME-Tools Classification [Dumke 1996]

The classification of the CAME-Tools results from their software life cycle in relation to the software components aspects referring to the process, the product and the resource.

On the other side, CAME-Tools can be classified according to the degree of integration in development environment such as integrated forms, external coupling forms and stand-alone metrics tools [Ebert 2005].

2.1 Product Quality Measurement Tools

Software include documentation are treated here as an artifact. The aim of these tools is to measure and predict attributes such as complexity, reliability, maintainability etc.

A large amount of software metrics have been developed and a wide variety of free and commercial metrics tools exist to implement and interpret these metrics from measured artifacts. This variety of tools allows users to select the tool best suited, e.g. depending on its price or depending on its supported programming language. Hence, the question was leading: Are all existing tools reliable? In other words, give different tools the same measured value for the same metric?

The response is a resounding “no”, these tools are not reliable. Existing software metric tools interpret and implement the definitions of software metrics differently.

One experiment was by Lincke et al. [Lincke 2008]. They inspected only java metrics tools on several java projects and showed that different metrics tools give different results for the same metrics on the same input.

For making simple measurements for other programming languages (Java, C#, C++, PHP, and Delphi) and comparing the measured values we constructed a small project in different language and calculated metrics value using other set of free and Trial Editions of metrics tools which collect a common subset of metrics. The inspected tools were:

- **CCCC:** It is a tool which analyzes C, C++ and Java files and generates a report in HTML and XML. Metrics supported are Lines of code (LOC), Lines of Comment (COM), Number of modules (NOM), McCabe's Cyclomatic Complexity (MVG), LOC/COM, MVG/COM, Fan-out, Fan-in, Rejected Lines, Weighted Methods per Class and metrics proposed by Henry&Kafura. [CCCC 2010]
- **JMetric:** It is a metrics collection and analysis tool. Metrics supported include Lines of Code, Number of Classes, Number of Packages, Number of Methods, LCOM, and Cyclomatic Complexity [Jmetric 2010].
- **CodeAnalyzer:** It is a java metrics application for C, C++, java, HTML and assembly and generates a report in HTML and XML. Metrics supported are Total Files, Total Lines, Code Lines, Whitespace Lines, Comment Lines, Average Line Length, Code Lines/File, Comment Lines/File, Code/Whitespace Ratio, Code/Comments Ratio, Code/(Comments + Whitespace) Ratio [CodeAnalyzer 2010].
- **Dependometer:** It is available for projects written in Java, C++ or C#. The calculated metrics are Afferent coupling, Efferent coupling, Instability, Number of classes, Number of abstract classes and much more [Dependometer 2010].
- **Krakatau Essential PM:** It is a Multilanguage project tool and applies Ada, ASP.NET, C/C++, C#, IDL, Java, JSP, Oracle SQL, Perl, PHP, VB/VB.net, VHDL and XML. It includes a wide range of source code metrics like Number of Files, Halstead size metrics and numerous LOC metrics. Krakatau generates reports in HTML, CSV and XML [Krakatau 2010].

Our measurements showed that the measured values are tool depended. That means different tools deliver different values for the same metric.

In this time showed Breuker et al. [Breuker 2009] in their own similar experiment that a reason for these variations in measured values between different tools is that tools use different definitions of the metrics, different from other tools and / or the literature. However, that does not mean that all metric tools don’t have implementation errors (see also [Breuker 2009]).

In fact, sometimes it was not clear what the metrics are measured or what definition of metrics has been used in the tool. For example, the shortcuts from metrics in some tools do not correspond to known acronyms. The metrics DIT, NOC by Chidamber & Kemerer are described for example in some tools as DOIH, NOCC.

In addition, from an industrial point of view it is not clear what to do with some language-specific technologies like *LINQ* (Language Integrated Query) from Microsoft. It is also not clear what to do with some technologies that provided from extern frameworks. For example, *Dependency Injection* that implemented in the framework spring. This is a technique for initialization of objects separately from the code that uses it. How will be in this case the metric LOC implemented?

Anyway, to be on the safe side, just one metric tool should be used for measuring in the company.

2.2 Process Quality Measurement Tools

Beside the measurement tools for product quality additional tools has been developed to measure the software development process itself. The aim of these tools is to understand the process context and to improve the process.

2.2.1 CMM Quest

A common example is the CMMI assessment tool from CMM Quest [CMM Quest 2010]. CMM Quest supports the appraiser in each appraisal phase, from planning, preparation and execution up to the evaluation of results and preparation of the appraisal presentation.

The appraisal phases are separated in CMM Quest in four steps:

1. **Prepare:** Preparation of the appraisal. This includes selected process areas and the chosen maximum capability level (for all selected process areas or for each process area individually). The order of the selected process areas can be arranged as desired. General questions can be selected (i.e. Information in regards: company data, employees, projects, appraisal team, etc).
2. **Fill in:** Performance of the appraisal. During an appraisal evidence and notes are documented and rated. Rating is based on “base and generic” practices and performed within the fill in pane.
3. **Analyze:** Graphical analysis of the appraisal results. To analyze appraisal data various charts are available within the analysis pane.
4. **Reports:** To create an appraisal report various report types (e. g. HTML or MS Word format) can be used within the reports pane.

Results of an appraisal are stored in an appraisal file. This file contains all selected process areas of the appraisal model as well as their ratings and notes. [CMM Quest 2010]

2.2.2 SPiCE-Lite

Another common example for suchlike tools is the SPiCE-Lite. It is a self assessment tool for performing self-analysis and evaluation in small software organization. SPiCE-Lite also provides outstanding cost/usage analysis as well differential analysis facilities, in which process quality can be compared before and after methods and tools have been introduced [SPiCE-Lite 2010].

The SPiCE-Lite software consists of two parts: ‘Fill in’ and ‘charts’. The ‘Fill in’ part covers process questions each relating to at least one process category. Together these process categories encompass the entire software development process. Up to nine process attributes are assigned to each of these processes, to which the user can assign a score from 0 to 100%.

As general validation suchlike tools for process measurement we can say that the License Agreements suchlike tools are still expensive. Moreover our experiences showed that the completion of the questions can be needed four to five hours. Software engineers sometimes consider these tools to be bureaucratic and time consuming. Although they usually agree about the general need for support of suchlike tools, engineers often find good reasons why such filling is not necessary appropriate to their particular project.

3 Software Measurement Tools Situation

Software measurement helps us to determine whether software products and software processes work against established standards. To measure is simple. But the fundamental difficulty in measurement process is knowing what metrics will be collected, when in the process, how it will be done, who will do it and what the meaning of results.

There are many approaches which tried to replay to these or some of these questions. An example is the E4 approach proposed by Ebert and Dumke in [Ebert 2007]. The **E4-measurement process** model based on the standard ISO 15939 and extends the GQM-Paradigm. This Approach contains four essential steps:

- Establish: identifying improvement goals and establishing objectives and measurement activities that should be achieved
- Extract: extracting the right information for the established need, e.g. the number of defects are found after the phase where they created
- Evaluate: analyzing and evaluating the information, e.g. cost benefits, business case, usefulness of the results from projects, and market readiness
- Execute: executing corrective actions and provide feedbacks and improvements based on the analyzed measurement data.

The E4-meaurement process is a management paradigm and can also be presented as closed control loop. The four steps of the E4-meaurement process should be introduced in this order.

Software measurement doesn’t only improve the quality of software product and process, but it will also become a helpful tool in training junior engineers of the team. However, measurement tools are not capable of determining the correctness of business logic. To verify correctness other technologies are required, e.g. human inspection and unit tests. The measurement tools therefore allows to spent time on this critical issue of suchlike technologies.

Our experiences allow the following major issues as a result of applying measurement tools in industrial environments:

- Chasm between research in academia and the use of software metrics in industry
- Expensive tools evaluation
- Platform dependence
- Measurement needs to be done with clear objectives in mind
- Different developers means different programming style
- Poorly handling of the measurement results - expert knowledge from user is required
- Poorly reliability of metric tools - different metrics tools give different results for the same metrics on the same project
- Poorly usability of metric tools – most of these tools require tedious from filling and work recording, another tools are ambiguous to understand

Moreover it is still difficult to replay some questions like:

- Which measurement tools support the used programming language?
- Which measurement tools apply the development process?
- Which measurement tools improve process quality and which improve product quality?
- Is the measurement tool combinable with the used CASE-tool?
- How many metrics and measurement tools are necessary?
- If more than one measurement tool are needed, can be combined the selected measurement tools?
- Another problem is the long-term dependency of tool manufacturers, especially when they disappear from the market.

These and other questions require answers and investigations for the successful use of measurement tools in software sector.

4 CASE Tool Based Software Processes

One approach to improve the situation described above and provide the benefits of measurement tools is the integration of metrics in the computer aided software engineering (CASE) tools.

We can improve this situation by a meaningful quality measurement implementation as shown in Figure 2:

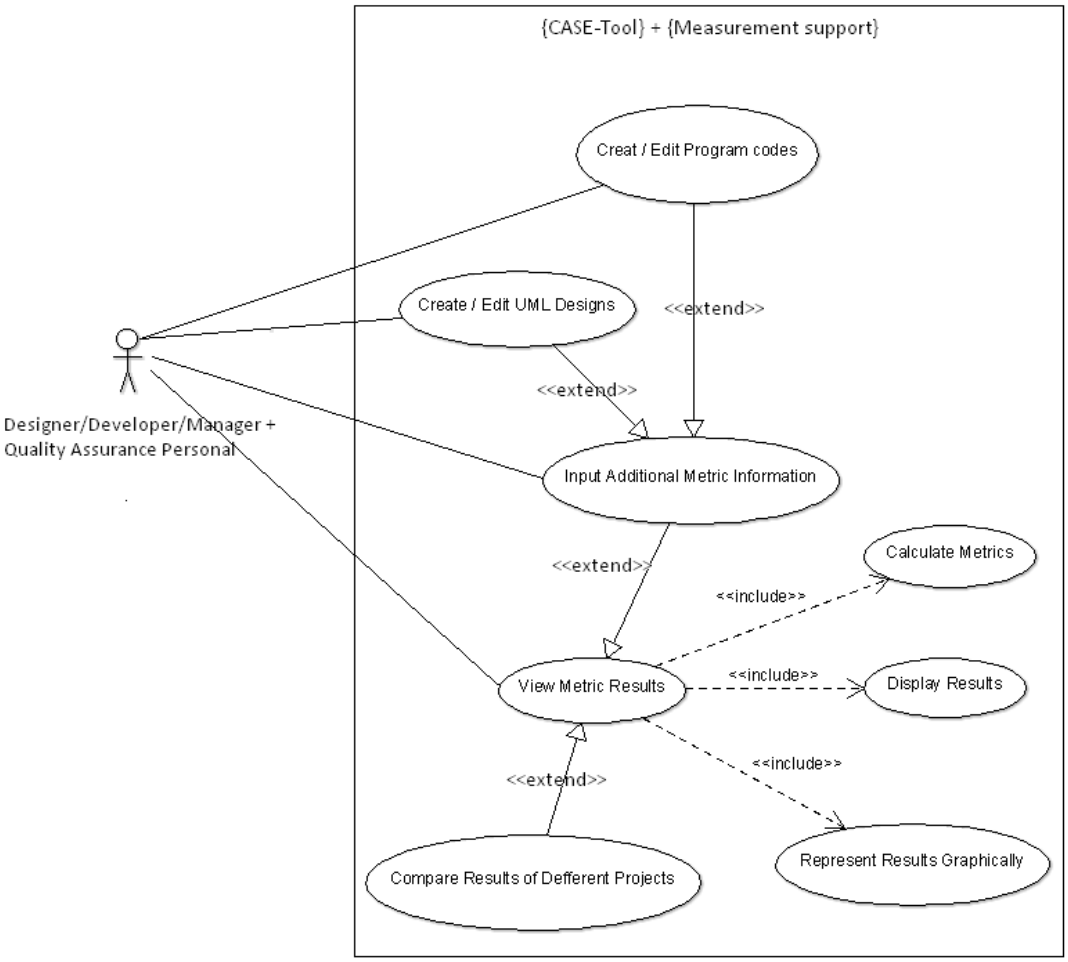


Figure 2: Use Case Diagram of CASE-based Software measurement

For analysis the quality of software product and process it is important to view metric results in a variety of ways.

Nowadays a large body of powerful CASE Tools is emerging. Unfortunately, only a handful of CASE tool suppliers thought of this solution and have built some metrics into their tool or provided it as plug-in. However there does not yet exist a standard scheme for integration metrics into the CASE-tools, especially when the tool supports a few activities in the information systems life cycle. So the provided solutions are still not really helpful.

The inspected tools are the following:

- Borland Together product family [Together 2010]
  - Together 2008
  - Together 2006 Release 2 for Eclipse
  - Together 2006 for Visual Studio
  - ControlCenter 6.2

- Microsoft Visual Studio 2010 product family [VS 2010]
  - Visual Studio 2010 Premium
  - Visual Studio 2010 Ultimate
- Enterprise Architect Version 8.0 [EA 2010]
- Metrics Eclipse Plug-in Version 1.3.6 [Metrics 2010]
- metricsOne Rational Rose Plug-in [metricsOne 2004]
- Embarcadero RAD Studio 2010 (Delphi and C++Builder) [RS 2010]

We limited our selection above on CASE tools which are currently provided metrics at the same time in their tools. The metrics are either integrated into the tool, or available as a plug-in in a separate component.

5 CASE-Based Test Scenario

To present information on how the quality assurance metrics available in the inspected CASE-Tools and its results, we have constructed here a project for an airline agency with different programming language (Java, C++, C# and Delphi) based on UML approach. The research prototype is a simple example (about 32000 object oriented LOC in ca. 70 classes) that nevertheless contains all important phases and structure elements of a real IT project. This prototype was important not only for manipulating and comparing the implemented metrics, but also for inspecting their suggested limits and feedbacks.

As a minimum goal, the tools should first be able to use its metrics and show the results.

6 Appraisal of CASE Tool Evaluation Results

In our SQA concept [Yazbek 2010] we established fifteen criteria such as metrics coverage, metrics providing, metrics suites, metrics customizing, metrics extending, metrics feedback, metrics filtering, sorting results, metrics visualization, saving and loading metrics results, comparing metrics results, printing results, exporting results, copying metrics results to the clipboard and metrics verification.

In due consideration of this SQA concept we get the following Figure 3:

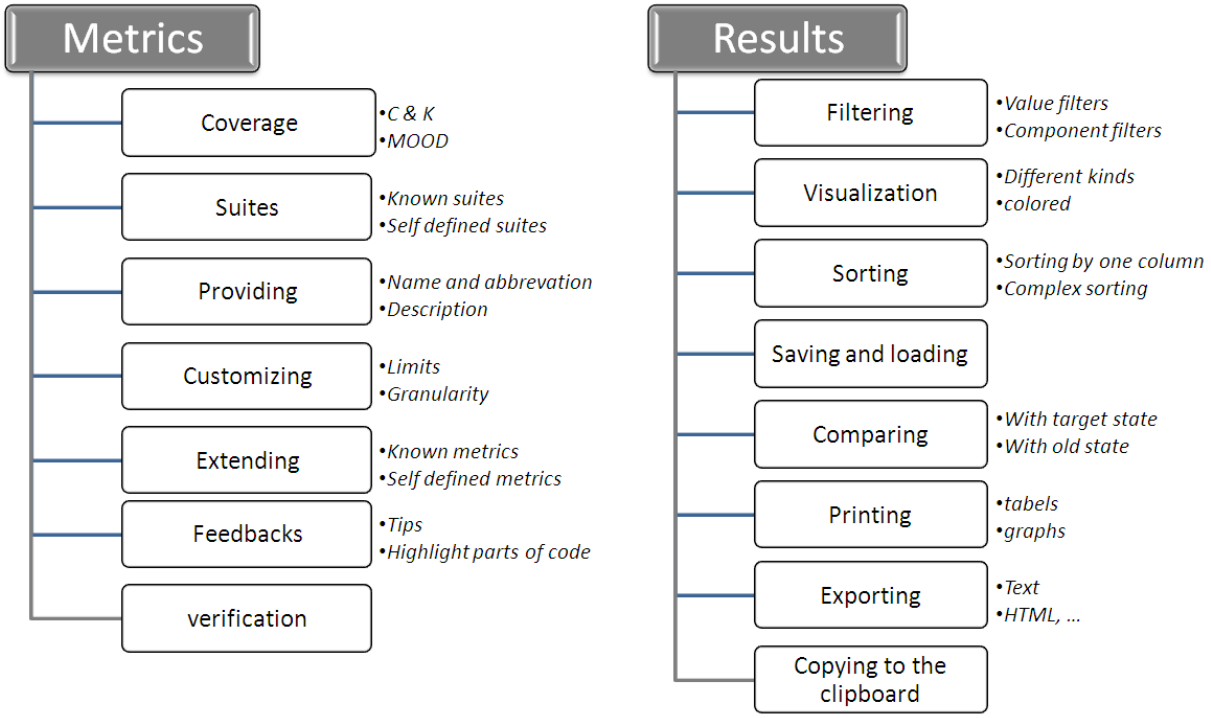


Figure 3: The Conept of QA for Metrics in CASE Tools [Yazbek 2010]

That leads to a framework as a measurement based concept for evaluate of the QA level of CASE-based software development in practical situation. The general framework intention is to evaluate and improve the CASE tool based measurement processes themselves. This framework is for team members involved in software product and process development and for members who wants to improve his development through the effective use of metrics.

6.1 Framework Fundamental Principles

All in all we also note the following principles to development the concept described above in industrial CASE-Tools:

- The QA support should provide a reusable framework for software measurement application
- The QA support should not only be limited to product assessments, but also includes resources and process data
- Easy to learn and to use
- Expandable and Supports existing metrics
- Automated: We don't consider the means of "automated" that metrics have to be collected manually. Instead, we mean:
  - metric analysis
  - metric visualizing
  - metric reporting
  - documentation-generation
  - metric prediction

- Portable: the QA support should be able to transfer from one environment to another.
- Flexible: users can customize the existing baseline set and add their own sets of metrics
- Low cost

6.2 CASE-based SQA Phases

It is very important to identify trends in the results in measuring over a period of time. In order to improve a given industrial CASE-based measurement forms we define the following Framework steps:

- Analyze the current measurement situation of the CASE infrastructure
- Identify the direction(s) of measurement improvement
- Determine the measurement infrastructure artifacts
- Implement the CASE-based measurement extensions
- Evaluate the new measurement situation/process

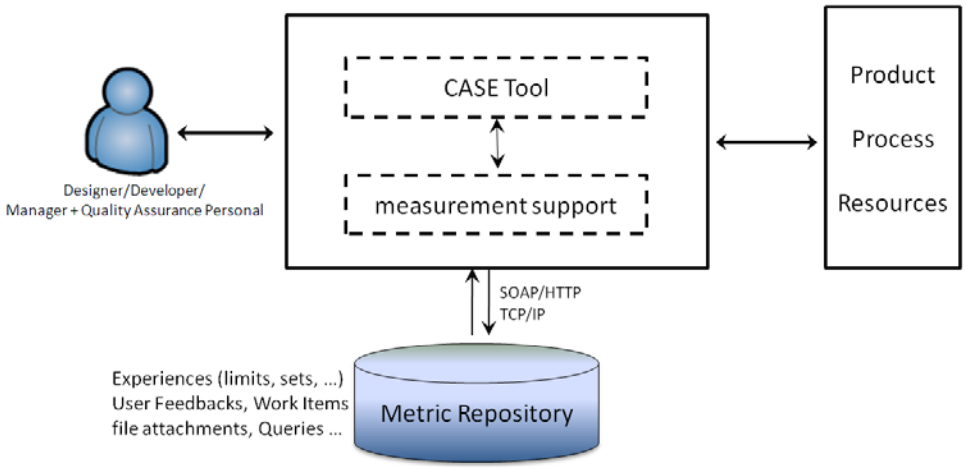


Figure 4: metrics tools integrated into CASE-Tool

The Metric Repository is to storing project experiences (e. g. limits and metrics sets), and to export metrics results for sharing them with other team members or comparing them later with other results. Product owners and team members can also create team work items (Bugs, Tasks, issues, etc.) that are based on measuring results to track and monitor the development of a product and its features. For Example the team can create tasks to track work hours that are needed to implement a requirement or other area of work.

Due to this approach the tool is able to storing measuring data from different workplaces in a common data base. Team members can display and report different types of data. In order to simplify the data handle our framework present universal approaches to use like:

- XML-based Interfaces
- Flexible and dynamic appropriation (e.g. as service orchestration)
- Simple communication technologies (SOAP, HTTP, TCP/IP, etc.)

In addition to the simple handle, we consider the pervasive spread of these technologies.

The application and validation of the steps and phases described above are the contents of the next chapter.

7 Framework Application and Validation

Our first evaluation summarizes the general framework steps in their fulfilling characteristics. We use the evaluation as “+” for “fulfilled”, “-“ as “not fulfilled” and “o” as “part fulfilled“. The following table shows this kind of metrics-based evaluation of CASE tool based measurement support.

Tool	Metrics coverage	Metrics providing	Metrics suites	Metrics customizing	Metrics extending	Metrics feedback	Metrics verification
Together	o	o	+	+	+	+	+
Visual Studio	-	o	-	-	-	+	-
Enterprise Architect	-	o	-	+	-	-	-
Metrics Eclipse Plug-in	o	o	-	+	-	+	+
metricsOne	+	o	o	+	-	+	-
RAD Studio	+	o	+	+	+	+	+

Tool	Results filtering	Results visualization	Results sorting	Results saving	Results comparing	Results printing	Results exporting	Results copying
Together	+	o	+	+	+	+	+	-
Visual Studio	+	-	+	o	-	-	+	+
Enterprise Architect	-	-	-	-	-	-	+	-
Metrics Eclipse Plug-in	-	o	+	-	-	-	+	-
metricsOne	+	o	+	+	+	+	+	+
RAD Studio	+	o	+	+	o	+	+	-

Table 1 - Overview of metrics concept

As we see, the described CASE tools meet different requirements at measuring in different levels and ways.

7.1 Together Measurement Level

Borland Together is well-known in practice especially for object-oriented system development using current OO languages such C++ and Java. Together provides metrics for object-oriented code and design as Chidamber & Kemerer, the MOOD-metrics from Abreu, and the McCabe and Halstead metrics. Two metrics are for the documentation (*Comment Ratio* und *True Comment Ratio*). Managers can extend

this QA module by creating their own metrics if they have more specific needs. Together presents the results as graphic output (Bar Graph, Distribution Graph or Kiviat Graph) and provides tips for interpreting the results, but it is up to the engineers to examine these results (see [Together 2010] and [Yazbek 2010]).

7.2 Metrication in Visual Studio

Microsoft Visual Studio is well-used in the .NET developer community world-wide and support the visual programming mainly. Visual Studio provides five basic object-oriented code metrics - Lines of Code, Class Coupling, Depth of Inheritance, Cyclomatic Complexity und Maintainability Index. These metrics are provided without shortcuts and without info about them in the tool. Engineers con not define limitations for these metrics and cannot customize them. It is also not possible to extend this QA module if engineers have more specific needs. There is also no possibility to visualize the results in the tool itself. (see [VS 2010] and [Yazbek 2010]).

7.3 Measurement in Enterprise Architect

Enterprise Architect was used in the enterprise application integration community (EAI) and prefers the building of infrastructures of business applications. It supports Round-trip engineering in Java, C#, C++, ActionScript, Delphi, PHP, Python, Visual Basic and Visual Basic.NET. Engineers and managers can measure risk and effort with this tool. They can also use this EA to estimate the size of a project, and assign resources to elements. Project estimation is based on Karner's Use Case Points Method, which is based on these two metrics: EWE (Estimated Work Effort) and EC (Estimated Cost). However, before estimating project size, the following values must be calibrated: TCF (Technical Complexity Factors), ECF (Environment Complexity Factors), Default Hour Rate, and Use case complexity. The project timescale will be then estimated as following:

EWE = Duration * sum of Complexity * TCF * ECF

EC = EWE * Default Hour Rate

The results can be exported to a RTF file. There is also no way to represent the results graphically, and the results are in no way comparable with other measurements, so that an improvement in the project is not visible (see also [EA 2010] and [Yazbek 2010]).

7.4 Metrics Eclipse Plug-in

This Plug-in is from an IBM initiative for education and training was more and more used in practice. It provides 23 object-oriented metrics for Java as LOC, NOC, McCabe Cyclomatic Complexity and the metrics from Chidamber & Kemerer. In the plug-in there is no description from the metrics and there are no tips on how to interpret the results.

The results are displayed as a simple table and there is no way to sort, save, print and compare the results. Engineers can define minimum and maximum limits for

each metric. Out-of-range and in-range results will be colored in the table. Engineers can visualize the metric results only in a Package Dependency Graph (see also [Metrics 2010] and [Yazbek 2010]).

7.5 MetricsOne Measurement Level

MetricsOne is one of the well-known metrics extensions for the UML-based software development based on the Rational Unified Process (RUP) kind of development. MetricsOne provides Class metrics, Use Case metrics, Operations metrics, and Packages metrics. Limits for metrics are defined in the tool and engineers can customize it. Microsoft Excel must be installed on the same machine where MetricOne are runs because the metric results will be sent to the Excel application. Out-of-range results will appear in red. There is also no way to visualize the results in the tool itself. Engineers should do that in Excel (see also [metricsOne 2004] and [Yazbek 2010]).

7.6 Metrication in Embarcadero RAD Studio 2010

RAD Studio was mainly used in the non IT area for scientific and government applications. RAD Studio is an integrated CASE-Tool for creating Windows applications and supports UML modeling and Round-trip engineering for the programming languages C++ and Delphi. RAD Studio provides 89 object-oriented code metrics for C++ and Delphi as Chidamber&Kemerer metrics, MOOD metrics, McCabe-metrics und Halstead metrics. For the documentation, the metrics Comment Ratio (CR) and True Comment Ratio (TCR) are available. Tips on how the metrics and its results are used can be found in the tool. However, it is up to engineers to determine whether the results are acceptable. RAD Studio can also demonstrate metrics results graphically in bar charts and Kiviat charts. For each metric there are settings for options such as limits and granularity. In RAD Studio it is possible to sort, filter and export the results (see also [RS 2010] and [Yazbek 2010]).

8 Conclusions and Future Work

Due to these results we come to the conclusion that software using the inspected tools can be quickly measured during design, implementing, testing and maintenance after every significant change, but the measurement support in these tools is introduced as single concepts and is not arranged within the whole process of software development. This may be due to the missing of QA concept using CASE tools.

Further works includes the prototypical implementation of the CASE tool extension and their practical application and the investigation of further kinds of CASE tool including modern infrastructures and services.

The consideration of the CASE-based software development from a process point of view should also improve the CASE tool situation themselves.

References

[Breuker 2009]: Alain Abran et al.: Software Process and Product Measurement, INDUSTRIAL TRACK PAPERS. Reliability of software metrics tools, PP. 10-22. Amsterdam, Netherlands. November 2009

[CCCC 2010]: http://sourceforge.net/projects/cccc/ (accessed 2010-7)

[CMM Quest 2010]: http://www.cmm-quest.com/default.htm (accessed 07-2010)

[CodeAnalyzer 2010]: http://www.codeanalyzer.teel.ws/ (accessed 2010-07)

[Dependometer 2010]: http://source.valtech.com/display/dpm/Dependometer (accessed 2010-07)

[Dumke 1996]: Reiner R. Dumke: CAME tools-lessons learned. Proceedings of the Fourth International Symposium on Assessment of Software Tools (SAST '96) IEEE Computer Society Washington, DC, USA 1996

[Dumke 2003]: Reiner Dumke: Software Engineering. Vieweg, 2003

[EA 2010]: http://www.sparxsystems.de/ (accessed 2010-07)

[Ebert 2005]: Christof Ebert, Reiner Dumke, Manfred Bundschuh and Andreas Schmietendorf: Best Practices in Software Measurement. Springer-Verlag Berlin Heidelberg 2005

[Ebert 2007]: Christof Ebert, Reiner Dumke: Software Measurement - Establish, Extract, Evaluate, Execute. Springer; 1 edition (August 30, 2007)

[Jmetric 2010]: http://www.it.swin.edu.au/projects/jmetric/products/jmetric/default.htm (accessed 2010-07)

[Krakatau 2010]: http://www.powersoftware.com/ (accessed 2010-07)

[Lincke 2008]: Rüdiger Lincke, Jonas Lundberg and Welf Löwe: Comparing Software Metrics Tools. Proceedings of the 2008 international symposium on Software testing and analysis 2008, Seattle, WA, USA July 20 - 24, 2008 Pages 131-142

[Metrics 2010]: http://metrics.sourceforge.net/ (accessed 2010-07)

[metricsOne 2004]: metricsOne Rational Rose Plug-in: Tools and associated documentation are available at the University Magdeburg

[RS 2010]: http://www.embarcadero.com/products/rad-studio (accessed 2010-07)

[SPiCE-Lite 2010]: http://www.spicelite.com/ (accessed 2010-07)

[Testingfaqs 2010]: http://www.testingfaqs.org/t-static.html (accessed 2010-07)

[Together 2010]: http://techpubs.borland.com/together/ (accessed 2010-07)

[VS 2010]: http://www.microsoft.com/visualstudio/en-us (accessed 2010-07)

[Yazbek 2010]: A Conept of Quality Assurance for Metrics in CASE Tools. Software Engineering Notes, Sept. 2010, p. 32

Schmietendorf, A.:

5. Hochschul-Roundtable der CECMG/DASMA

Industrielle und gesellschaftliche Herausforderungen beim flexiblen Sourcing von IT-Projekten/-Dienstleistungen

Shaker Verlag, Aachen, 2010 (82 Seiten)
ISBN 978-3-8322-8940-9

Das Buch fasst die einzelnen Diskussionsbeiträge vornehmlich aus dem industriellen Umfeld zusammen und stellt auch aktuelle Forschungsansätze zu diesem Themengebiet vor.

Dumke, R.; Abran, A.:

Cosmic Function Points: Theory and Advanced Practices

CRC Press Taylor & Francis Group, 2010 (328 Seiten)
ISBN-10: 1-4398-4486-0
ISBN-13: 978-1-4398-4486-1

This book has the following characteristics: the theme is about a new software size estimation method including their scientific and practical background; the chapters are based on papers, that would be published in our conference proceedings during the last six years; the authors are wellknown participants of the international software measurement community (see e. g. COSMIC, IFPUG etc.) and the book content is structured in the main problems of building new measurement or estimation methods in general and should be interesting for the software quality.

Dumke, R.; Mencke, S.; Wille, C.:

Quality Assurance of Agent-Based and Self-Managed Systems

CRC Press Taylor & Francis Group, 2010 (154 Seiten)
ISBN 978-1-4398-1266-2

The challenges in implementing intelligent and autonomous software systems remain to be the development of self-adapting systems, self-healing applications. Corporate global creation, and collaborated robotic teams. With software agent technology widely recognized as a key approach in implementing such global infrastructure, the importance of the role of quality assurance of agent-based systems and system development is growing daily.

Based on the author's more than 15 years of experience in software agent technology, **Quality Assurance of Agent-Based and Self-Managed Systems** presents the basic principles and structures of agent technology. It covers the main quality issues of software system development and provides examples of agent measurement and evaluation. The authors focus on software agent systems and multiagent systems (MAS) and discuss the determination of quality properties. They also explain different techniques and approaches used to evaluate the development of MAS. The final chapter summarizes quality assurance approaches for agent-based systems and discusses some open problems and future directions.

Although often complex and difficult to manage, the applications for software agent systems in essential life systems in crease every day. Since the quality of the agent-based self-managing systems is a central point of software risk; analyzing, evaluating, and improving the quality measurement situation will always be a concern when developing these systems. With more than 60 illustrations and 20 tables, this book builds a foundation in quality and quality for agent-based technology.

Humphrey, W.S.; Thomas, W.R.:

Reflections on Management

How to Manage Your Software Projects, Your Teams, Your Boss, and Yourself

Addison-Wesley, 2010 (288 Seiten)
ISBN-10: 0-321-71153-X
ISBN-13: 978-0-321-71153-3

This book, drawn from Humphreys books, articles, and columns, comprises a collection of advice, stories, and hard-earned wisdom, rather than specific instruction on how to implement the PSP or TSP (which are thoroughly covered in Humphreys book on those specific subjects). What emerges for the reader is an understanding that successful software project management is a journey with many obstacles. To succeed, engineers must manage more than their projects. They must use their own experience and that of their teams to first understand and then plan the project ahead. They must influence their teams' attitudes and methods for doing disciplined work. And they must persuade their bosses to set aside ill-informed notions of schedules and resource commitments and look instead at hard, historical data.

The essays in Part I provide insights on types of plans and the planning process. Part II covers team building and motivation. Part III describes how to work with your managers and persuade them to use best practices. And Part IV examines your personal responsibilities, commitments, and processes.

These essays shine a light on the challenges inherent in software development and can set engineers on the road to understanding how to succeed. And while Humphreys particular expertise is software, practitioners in every field of business will benefit from the wisdom and advice contained here.

Abran, A.; Braungarten, R.; Dumke, R.R.; Cuadrado-Gallego, J. J.; Brunekreef, J.:

Software Process and Product Measurement

International Conferences IWSM 2009 and Mensura 2009

Amsterdam, The Netherlands, November 4-6, 2009

Springer-Verlag, 2009 (346 Seiten)
LNCS 5891, ISBN 978-3-642-05414-3

Since 1990 the International Workshop on Software Measurement (IWSM) has been celebrated annually alternating between Montréal (Canada) and various cities across Germany. The Montréal editions have been organized by the Software Engineering Research Laboratory (GELOG) of the École de technologie supérieure – Université Québec, which is directed by Prof. Alain Abran. The German editions have been organized jointly by the Software Measurement Laboratory (SMLAB) of the Otto von

Guericke University Magdeburg (Germany), which is directed by Prof. Reiner R. Dumke.

Since 2006 the Spanish software measurement community directed by Prof. Juan J. Cuadrado-Gallego as MENSURA participate at our conferences also.

This volume comprises the proceedings of IWSM / Mensura 2009 and consists of the final papers presented at these joint events. Each one of these papers has been thoroughly revised and extended in order to be accepted for publication.

Büren, G.; Dumke, R.:

MetriKon 2009 – Praxis der Software-Messung

Tagungsband des DASMA Software Metrik Kongresses

19.-20. November 2009, Kaiserslautern

Shaker Verlag, Aachen, 2009 (320 Seiten)
ISBN 978-3-8322-8649-1

The book includes the proceedings of the DASMA Metric Conference MetriKon 2009 held in Kaiserslautern in November 2009, which constitute a collection of theoretical studies in the field of software measurement and case reports on the application of software metrics in companies and universities.

The contents are described by the listing of the paper abstracts in this Measurement News.

Schmietendorf, A.; Fiedler, M.; Dumke, R.R.:

BSOA 2009

4. Workshop Bewertungsaspekte serviceorientierter Architekturen

18. November 2009, Darmstadt

Shaker Verlag, Aachen, 2009 (142 Seiten)
ISBN 978-3-8322-8551-7

Moderne Integrationsarchitekturen bestimmen nach wie vor die Art und Weise, wie softwaretechnische Lösungen im Kontext industrieller Problemstellungen umzusetzen sind. Immer stärker wird den Verantwortlichen dabei bewusst, dass der Erfolg unternehmensweit genutzter Integrationsarchitekturen weniger von einzukaufenden Produkten abhängt, als vielmehr von prozess- und organisationsbezogenen Aspekten beeinflusst wird. Die Implementierung einer serviceorientierten Architektur ist als Strategie zu verstehen, wobei die Vision einer geschäftsprozesskonformen Ausrichtung der IT-Landschaft verfolgt wird. In diesem Kontext werden vielfältige Bewertungsansätze benötigt, die den gesamten

Lebenszyklus eines Informationssystems erfassen können. Die BSOA-Initiative widmet sich seit mehr als 3 Jahren diesen Fragen.

Aus der Vielzahl an eingereichten Beiträgen konnte durch das Programmkomitee eine anspruchsvolle Agenda zusammengestellt werden. Ausgewählt wurden 6 Beiträge für eine Präsentation während der Workshopsitzungen und 4 Beiträge für Posterpräsentationen während der Pausenzeiten. Dazu kommen noch die beiden Keynote-Vorträge, die das vorliegende Buch in exzellenter Weise ergänzen und prägen.

Schmietendorf, A.:

Aufwandsschätzung bei Projekten nach dem eXtreme Programming-Paradigma

Shaker Verlag, Aachen, 2009 (643 Seiten)
ISBN 978-3-8322-8560-9

In diesem Buch wird eine monografische Darstellung grundlegender Aufwandsschätzmethoden im Umfeld von Ansätzen der agilen Software-Entwicklung diskutiert.

Schneider, K.:

Experience and Knowledge Management in Software Engineering

Springer-Verlag, 2009, XVI, Hardcover (235 Seiten)
ISBN 978-3-540-95879-6

Nowadays, there is software everywhere in our life. It controls cars, airplanes, factories, medical implants. Without software, banking, logistics and transportation, media, and even scientific research would not function in the accustomed way. Building and maintaining software is a knowledge-intensive endeavour and requires that specific experiences are handled successfully. However, neither knowledge nor experience can be collected, stored, and shipped like physical goods, instead these delicate resources require dedicated techniques. Knowledge and experience are often called company assets, yet this is only part of the truth: it is only software engineers and other creative employees who will effectively exploit an organisation's knowledge and experience.

Kurt Schneider’s textbook is written for those who want to make better use of their own knowledge and experience – either personally or within their group or company. Everyone related to software development will benefit from his detailed explanations and case studies: project managers, software engineers, quality assurance responsibles, and knowledge managers. His presentation is based on years of both practical experience, with companies such as Boeing, Daimler, and Nokia, and research in renowned environments, such as the Fraunhofer Institute. Each chapter

is self-contained, it clearly states its learning objectives, gives in-depth presentations, shows the techniques' practical relevance in application scenarios, lists detailed references for further reading, and is finally completed by exercises that review the material presented and also challenge further, critical examinations. The overall result is a textbook that is equally suitable as a personal resource for self-directed learning and as the basis for a one-semester course on software engineering and knowledge management.

Preprints/Technical Reports:

Farooq, A.; Dumke, R. R.: *Evaluation Approaches in Software Testing.* University of Magdeburg 2008

Richter, K.; Dumke R. R.: *The Causal-Based Software Process Modelling.* University of Magdeburg 2008

Dumke, R.; Kunz, M.; Farooq, A.; Georgieva, K.; Hegewald, H.: *Formal Modelling of Software Measurement Levels of Paradigm-Based Approches.* University of Magdeburg 2008

see as pdf files:

http://ivs.cs.uni-magdeburg.de/sw-eng/agruppe/forschung/Preprints.shtml

SWQD 2010:

Software Quality Days
January 19-21, 2010, Wien, Austria
see: **http://www.software-quality-days.at/**

WOSP 2010:

7th International Workshop on Software & Performance
January 28-30, San Jose, CA, 2010
see: **http://www.inf.pucrs.br/wosp/**

IASTED SE 2010:

IASTED International Conference on Software Engineering 2010
February 16-18, 2010, Innsbruck, Austria
see: **http://www.iasted.org/conferences/home-677.html**

SQMB 2010:

3. Workshop zur Software-Qualitätsmodellierung und -bewertung
February 22, 2010, Paderborn, Germany
see: **http://sqmb.informatik.tu-muenchen.de/2010/**

SSE 2010:

3th International Workshop on Social Software Engineering
February 24, 2010, Paderborn, Germany
see: **http://www1.cs.tum.edu/static/sse10/**

CSMR 2010:

14th European Conference on Software Maintenance and Reengineering
March 15-18, 2010, Madrid, Spain
see: **http://www.sait.escet.urjc.es/csmr2010/**

SEPG 2010:

22th Software Engineering Process Group Conference
March 22-25, 2010, Savannah, Georgia, USA
see: **http://www.sei.cmu.edu/sepg/na/2010/index.cfm**

ASWEC 2010:

21th Australien Software Engineering Conference
April 6-9, 2010, Auckland, New Zealand
see: **http://aswec2010.massey.ac.nz/**

ICEME 2010:

International Conference on Engineering and Meta-Engineering
April 6-9, 2010, Orando, FL, USA
see: **http://www.iiis2010.org/iceme/website/default.asp?vc=32**

EASE 2010:

International Conference on Empirical Assessment in Software Engineering
April 12-13, 2010, Keele University, UK
see: **http://www.scm.keele.ac.uk/ease/**

STAREAST 2010:

Software Testing Analysis & Review Conference
April 25-30, 2010, Orlando, FL, USA
see: **http://www.sqe.com/stareast/**

SQS iqnite 2010:

Software Quality Systems Conference
April 28-30, 2010, Düsseldorf, Germany
see: **http://www.iqnite-conferences.com/de/index.html**

ICSE 2010:

International Conference on Software Engineering
May 2-8, 2010, Cape Town, South Africa
see: **http://www.sbs.co.za/ICSE2010/**

PSQT 2010:

International Conference on Practical Software Quality & Testing
West: May 10-14, 2010, Las Vegas; North: Sept. 13-17, 20010, Minneapolis, USA
see: **http://www.psqtconference.com**

SPICE 2010:

SPICE Conference
May 18-20, 2010, Pisa, Italy
see: **http://www.spiceconference.com/**

SERA 2010:

8th ACIS Conference on Software Engineering
May 24-26, 2010, Montreal, Canada
see: **http://users.encs.concordia.ca/~sera2010/**

XP 2010:

11th International Conference on Agile Software Development
June 1-4, 2010, Trondheim, Norway
see: **http://xp2010.org/**

SMEF 2010:

Software Measurement European Forum
June 10-11, 2010, Rome, Italy
see: **http://www.dpo.it/smef2010.htm**

PROFES 2010:

10th International Conference on Product Focused Software Process Improvement
June 21-23, 2010, Limerick, Ireland
see: **http://www.lero.ie/profes2010/**

SEPG Europe 2010:

Software Engineering Process Group Conference
June 28- July 1, 2010, Porto, Portugal
see: **http://www.sei.cmu.edu/sepg/europe/2010/index.cfm**

REFSQ 2010:

16th International Working Conference on Reuquirements Engineering: Foundation for Software Quality
June 30 - July 2, 2010, Essen, Germany
see: **http://www.sse.uni-due.de/refsq/2010/**

ICPC 2010:

18th International Conference on Program Comprehension
June 30 - July 2, 2010, Braga, Portugal
see: **http://icpc2010.di.uminho.pt/**

ICWE 2010:

International Conference on Web Engineering
July 5-9, 2010, Vienna, Austria
see: **http://icwe2010.webengineering.org/**

UKPEW 2010:

23th Annual United Kingdom Workshop on Performance Engineering
July 8-9, 2010, Warwick, UK
see: **http://www2.warwick.ac.uk/fac/sci/dcs/research/hpsg/events/ukpew2010/**

ICSP 2010:

International Conference on Software Process
July 8-9, 2010, Paderborn, Germany
see: **http://icsp10.upb.de/**

SETP 2010:

International Conference on Software Engineering Theory and Practice
July 12-14, 2010, Orlando, FL, USA
see: **http://www.promoteresearch.org/2010/setp/index.html**

ISSTA 2010:

International Symposium on Software Testing and Analysis
July 12-16, 2010, Trento, Italy
see: **http://selab.fbk.eu/issta2010/**

QSIC 2010:

10th International Conference on Software Quality
July 14-15, 2010, Zhangjiajie, China
see: **http://www.nudt.edu.cn/qsic2010/**

ENASE 2010:

5th International Conference on Evaluation of Novel Approaches to Software Engineering
July 22-24, 2010, Athens, Greece
see: **http://www.enase.org/**

ICGSE 2010:

International Conference on Global Software Engineering
August 23-26, 2010, Princeton, NJ, USA
see: **http://www.icgse.org/**

ASQT 2010:

Arbeitskonferenz Softwarequalität und Test
September 8-10, 2010, Klagenfurt, Austria
see: **http://www.asqt.org/**

QEST 2010:

5rd International Conference on Quantitative Evaluation of SysTems
September 15-18, 2010, Williamsburg, Virginia, USA
see: **http://www.qest.org/**

CONQUEST 2010:

11. International Conference on Software Quality
September , 2010, Nuremberg, Germany
see: **http://www.conquest-conference.org/**

UKSMA 2010:

20th Annual UKSMA Conference - Managing your Software (through Measurement)
October , 2010, London, UK
see: **http://www.uksma.co.uk/**

ESEM 2010:

International Symposium on Empirical Software Engineering & Measurement
October 16-17, 2009, Bolzano-Bozen, Italy
see: **http://esem2010.case.unibz.it/**

IWSM/Mensura/MetriKon 2010:

Common international Conference on Software Measurement
November , 2010, , Germany
see: **http://www.smlab.de/conferences.html**

BSOA 2010:

3. Workshop Bewertungsaspekte service-orientierte Architekturen
November , 2010, FZI, Karlsruhe, Germany
see: **http://ivs.cs.uni-magdeburg.de/~gi-bsoa/**

see also: OOIS, ECOOP and ESEC European Conferences

Other Information Sources and Related Topics

- **http://rbse.jsc.nasa.gov/virt-lib/soft-eng.html**
  Software Engineering Virtual Library in Houston

- **http://www.mccabe.com/**
  McCabe & Associates. Commercial site offering products and services for software developers (i. e. Y2K, Testing or Quality Assurance)

- **http://www.sei.cmu.edu/**
  Software Engineering Institute of the U. S. Department of Defence at Carnegie Mellon University. Main objective of the Institute is to identify and promote successful software development practices.
  Exhaustive list of publications available for download.

- **http://dxsting.cern.ch/sting/sting.html**
  Software Technology Interest Group at CERN: their WEB-service is currently limited (due to "various reconfigurations") to a list of links to other information sources.

- **http://www.spr.com/index.htm**
  Software Productivity Research, Capers Jones. A commercial site offering products and services mainly for software estimation and planning.

- **http://www.qucis.queensu.ca/Software-Engineering/**
  This site hosts the World-Wide Web archives for the USENET usegroup comp.software-eng. Some links to other information sources are also provided.

- **http://www.esi.es/**
  The European Software Institute, Spain

- **http://www.lrgl.uqam.ca/**
  Software Engineering Management Research Laboratory at the University of Quebec, Montreal. Site offers research reports for download. One key focus area is the analysis and extension of the Function Point method.

- **http://www.SoftwareMetrics.com/**
  Homepage of Longstreet Consulting. Offers products and services and some general information on Function Point Analysis.

- **http://www.utexas.edu/coe/sqi/**
  Software Quality Institute of the University of Texas at Austin. Offers comprehensive general information sources on software quality issues.

- **http://wwwtrese.cs.utwente.nl/~vdberg/thesis.htm**
  Klaas van den Berg: Software Measurement and Functional Programming (PhD thesis)

- **http://divcom.otago.ac.nz:800/com/infosci/smrl/home.htm**
  The Software Metrics Research Laboratory at the University of Otago (New Zealand).
- **http://ivs.cs.uni-magdeburg.de/sw-eng/us/**

Homepage of the Software Measurement Laboratory at the University of Magdeburg.

- **http://www.cs.tu-berlin.de/~zuse/**
  Homepage of Dr. Horst Zuse

- **http://dec.bournemouth.ac.uk/ESERG/bibliography.html**
  Annotaded bibliography on Object-Oriented Metrics

- **http://www.iso.ch/9000e/forum.html**
  The ISO 9000 Forum aims to facilitate communication between newcomers to Quality Management and those who have already made the journey have experience to draw on and advice to share.

- **http://www.qa-inc.com/**
  Quality America, Inc's Home Page offers tools and services for quality improvement. Some articles for download are available.

- **http://www.quality.org/qc/**
  Exhaustive set of online quality resources, not limited to software quality issues

- **http://freedom.larc.nasa.gov/spqr/spqr.html**
  Software Productivity, Quality, and Reliability N-Team

- **http://www.qsm.com/**
  Homepage of the Quantitative Software Management (QSM) in the Netherlands

- **http://www.iese.fhg.de/**
  Homepage of the Fraunhofer Institute for Experimental Software Engineering (IESE) in Kaiserslautern, Germany

- **http://www.highq.be/quality/besma.htm**
  Homepage of the Belgian Software Metrics Association (BeSMA) in Keebergen, Belgium

- **http://www.cetus-links.org/oo_metrics.html**
  Homepage of Manfred Schneider on Objects and Components

- **http://dec.bournemouth.ac.uk/ESERG/bibliography.html**
  An annotated bibliography of object-oriented metrics of the Empirical Software Engineering Research Group (ESERG) of the Bournemouth University, UK

News Groups

- **news:comp.software-eng**
- **news:comp.software.testing**
- **news:comp.software.measurement**

Software Measurement Associations

- **http://www.dasma.org**
  DASMA Deutsche Anwendergruppe für SW Metrik und Aufwands-schätzung e.V.

- **http://www.aemes.fi.upm.es**
  AEMES Association Espanola de Metricas del Software

- **http://www.cosmicon.com**
  COSMIC Common Software Measurement International Consortium

- **http://www.esi.es**
  ESI European Software Engineering Institute in Bilbao, Spain

- **http://www.mai-net.org/**
  Network (MAIN) Metrics Associations International

- **http://www.sttf.fi**
  FiSMA Finnish Software Metrics Association

- **http://www.iese.fhg.de**
  IESE Fraunhofer Einrichtung für Experimentelles Software Engineering

- **http://www.isbsg.org.au**
  ISBSG International Software Benchmarking Standards Group, Australia

- **http://www.nesma.nl**
  NESMA Netherlands Software Metrics Association

- **http://www.sei.cmu.edu/**
  SEI Software Engineering Institute Pittsburgh

- **http://www.spr.com/**
  SPR Software Productivity Research by Capers Jones

- **http://fdd.gsfc.nasa.gov/seltext.html**
  SEL Software Engineering Laboratory - NASA-Homepage

- **http://www.vrz.net/stev**
  STEV Vereinigung für Software-Qualitätsmanagement Österreichs

- **http://www.sqs.de**
  SQS Gesellschaft für Software-Qualitätssicherung, Germany

- **http://www.ti.kviv.be**
  TI/KVIV Belgish Genootschap voor Software Metrics

- **http://www.uksma.co.uk**
  UKSMA United Kingdom Software Metrics Association

Software Metrics Tools (Overviews and Vendors)

Tool Listings

- **http://www.cs.umd.edu/users/cml/resources/cmetrics/**
  C/C++ Metrics Tools by Christopher Lott

- **http://mdmetric.com/**
  Maryland Metrics Tools

- **http://cutter.com/itgroup/reports/function.html**
  Function Point Tools by Carol Dekkers

- **http://user.cs.tu-berlin.de/~fetcke/measurement/products.html**
  Tool overview by Thomas Fetcke

- **http://zing.ncsl.nist.gov/WebTools/tech.html**
  An Overview about Web Metrics Tools

Tool Vendors

- **http://www.mccabe.com**
  McCabe & Associates

- **http://www.scitools.com**
  Scientific Toolworks Inc.

- **http://zing.ncsl.nist.gov/webmet/**
  Web Metrics

- **http://www.globalintegrity.com/csheets/metself.html**
  Global Integrity

- **http://www.spr.com/**
  Software Productivity Research (SPR)

- **http://jmetric.it.swin.edu.au/products/jmetric/**
  JMetric

- **http://www.imagix.com/products/metrics.html**
  Imagix Power Software

- **http://www.verilogusa.com/home.htm**
  VERILOG (LOGISCOPE)

- **http://www.qsm.com/**
  QSM

SOFTWARE MEASUREMENT NEWS

VOLUME 15 **2010** **NUMBER 2**

CONTENTS

ISSN 1867-9196