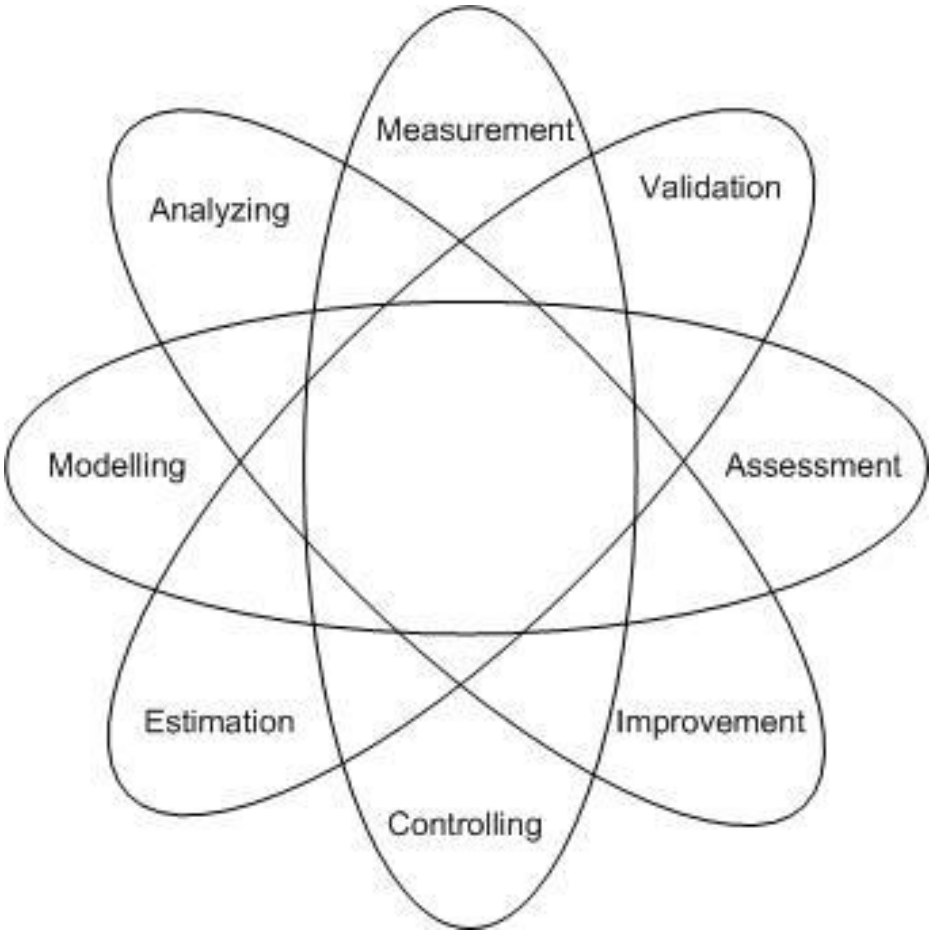
DASMA Volume 21, Number 1, February 2016

Software Measurement News

Journal of the Software Metrics Community



Editors:

Alain Abran, Manfred Seufert, Reiner Dumke, Christof Ebert, Cornelius Wille

Université du Québec
École de technologie supérieure
COSMIC
SM L@b

CONTENTS

Announcements **3**

Conference Report **9**

COSMIC Information **19**

Position Paper **25**

Christophe Commeyne, Alain Abran, Rachida Djouab:
Effort Estimation with Story Points and COSMIC Function Points - An Industry Case Study

New Books on Software Measurement **37**

Conferences Addressing Measurement Issues **41**

Metrics in the World-Wide Web **47**

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*
alain.abran@etsmtl.ca

Manfred Seufert
Chair of the DASMA, MediaanABS Deutschland GmbH
Franz-Rennefeld-Weg 2, D-40472 Düsseldorf, ***Tel.****: +49 211 250 510 0*
manfred.seufert@mediaan.com

Reiner Dumke
Professor on Software Engineering, University of Magdeburg, FIN/IKS
Postfach 4120, D-39016 Magdeburg, Germany, ***Tel.:*** *+49-391-67-52812*
dumke@ivs.cs.uni-magdeburg.de, http://www.smlab.de

Christof Ebert
Dr.-Ing. in Computer Science, Vector Consulting Services GmbH
Ingersheimer Str. 20, D-70499 Stuttgart, Germany, ***Tel.:*** *+49-711-80670-1525*
christof.ebert@vector.com

Cornelius Wille
Professor on Software Engineering, University of Applied Sciences Bingen
Berlinstr. 109, D-55411 Bingen am Rhein, Germany,
Tel.: *+49-6721-409-257,* ***Fax:*** *+49-6721-409-158*
wille@fh-bingen.de

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

Technical Editor: Dagmar Dörge

The journal is published in one volume per year consisting of two numbers. All rights reserved (including those of translation into foreign languages). No part of this issues may be reproduced in any form, by photoprint, microfilm or any other means, nor transmitted or translated into a machine language, without written permission from the publisher.

© 2016 by Otto-von-Guericke-University of Magdeburg. Printed in Germany

About the conference

IWSM Mensura is the premier international conference on measurement and data analytics. Each year practitioners and researchers from all over the world meet to discuss practical challenges and solutions in the field of software and IT measurement and data analytics.

On **October 5-7, 2016** the IWSM Mensura conference will be held in **Berlin, Germany**. The conference venue will be at the Berlin School of Economics, Campus Lichtenberg. More information on the conference can be found on the website**:** **http://www.iwsm-mensura.org**.

Theme & scope

Software and IT measurement are keys for successfully managing and controlling software development projects. Data analytics and measurement are essential for both business and engineering. They enrich scientific and technical knowledge regarding both the practice of software development and empirical research in software technology. The conference focuses on all aspects of software measurement and data analytics.

This year focus is the **Value of Data**, i.e. how to maximize the value for an organization from making use of data from their software applications and systems. The trend towards digitization also dramatically increases the amount of data that becomes available. The value of a company is increasingly hidden in its data and can only be exploited fully if these are used efficiently along the entire value chain. Big data becomes an important keyword to deal with. The conference also focuses on novel approaches and innovative ideas on how to optimize existing products and processes making use of data as well as using Big Data as an enabler for new application cases.

Topics of interest

We encourage submissions in any field of software measurement, including, but not limited to:

- Practical measurement applications
- Data analytics in practice, e.g. Enterprise embedded solutions
- Usage of big data analytics for improving products and processes
- Quantitative and qualitative methods for software measurement
- Measurement processes and resources, e.g. agile or model-driven
- Empirical case studies
- System and software engineering measurement
- IT and project cost and effort estimation, e.g., cost, effort, defects
- Functional size measurement
- Data analytics and measurement in novel areas, e.g. ECU's or web services
- Measures for Cognitive Computing

Conference language

The language for the conference, workshops and special sessions is English.

Full and Short Papers

Papers will undergo a strict peer review process. They must be submitted for review by the Program Committee through the EasyChair conference system via:

www.easychair.org/conferences/?conf=iwsmmensura2016

- Full papers (8 to 14 pages) or
- Short papers (3 to 6 pages)

Papers should not have already been published elsewhere, nor should they have been submitted to a journal or to another conference. At least one among the authors of each paper accepted must register for the conference and commit to paper presentation. All papers submitted must follow the IEEE CPS format (US letter format).
Accepted and presented papers (full papers and short papers) will be included in the conference proceedings, which will be submitted to the IEEE Computer Society Digital Library (CSDL), and IEEE Xplore. A selection of the accepted papers will be invited to re-submit extended versions for consideration for publication in a Journal.

Workshop proposals

The main idea of the workshops is to bring both practitioners and researchers together to exchange ideas on particular topics of importance. Workshop proposals should be described on two pages maximum and submitted directly to the Program Chair via program@iwsm-mensura.org.

Industry Presentation proposals

The main idea of industry presentations is to share experiences, challenges, solution approaches regarding the topics of interest for the conference from a practitioner's point of view in order to stimulate discussions with researchers and other practitioners. For presentation proposals, a title and abstract (at most half page) should be submitted directly to the Program Chair via program@iwsm-mensura.org.

Important dates

	Full Papers	Short Papers	Workshop / Industry Presentation
Submission	**April 17th, 2016**	April 17th, 2016	May 8th, 2016
Notification of acceptance	**June 12th, 2016**	June 12th, 2016	June 12th, 2016
Final version	**July 3rd, 2016**	July 3rd, 2016	August 28th, 2016

Contact information

General Chair	Christof Ebert (Vector and GI, Germany)
Program Co-Chairs	Jens Heidrich (Fraunhofer IESE and GI, Germany) Frank Vogelezang (Ordina, Nesma, and COSMIC, The Netherlands and Canada)
Financing Chair	Manfred Seufert (Mediaan and DASMA, Germany)
Local Chair	Andreas Schmietendorf (HWR Berlin, OvG-Universität Magdeburg, Germany)

If you have any questions with regards to this call for papers you may contact the Program Chairs via program@iwsm-mensura.org.

Big Data - Erfolgreiche Data Science Initiativen

(Qualitative und quantitative Bewertung)

14.04.2016 (09:00 bis 17:00 Uhr) Hamburg

Die Beschäftigung mit Big Data-Lösungen kann in vielen Bereichen beobachtet werden. Zumeist werden Nutzenspotentiale und Risiken dabei sehr allgemein und wenig messbar angeben. Zunehmend finden sich daher auch kritische Stimmen, die sich mit überzogenen Erwartungen und limitierenden Rahmenbedingungen auseinandersetzen. Mit Hilfe des Workshops sollen Erfolgskriterien für derartige Projekte herausgearbeitet werden. Im Mittelpunkt der interaktiv gestalteten Veranstaltung sollen im industriellen, wissenschaftlichen und öffentlichen Umfeld gewonnene Erfahrungen stehen.

Keynote im Rahmen der ECC-Tagung (09:15 Uhr)

Dr. Wolfgang Hildesheim, IBM Watson Group Leader DACH

Management von Cognitive Computing Systemen
- *Diversifizierte Datenquellen*
- *Cloud-basierte Lösungen*

Session 1 - Einführung (10:00 Uhr):

Prof. Dr. Andreas Schmietendorf (HWR Berlin/Uni Magdeburg)

Big Data – Einsatzpotentiale heben, aber wie?
- *Vielfältige Anwendungsszenarien*
- *Erfolgskriterien für Data Science*

Session 2 – Erfahrungsbericht (10:45 bis 12:00 Uhr):

Dr. Robert Neumann, Jan Hentschel Ultra Tendency UG

Hadoop als Treiber organisatorischer Veränderungen
- *Strategie, Governance und Management*
- *Bewertung von Aufwand, Nutzen und Risiken*

Session 3 – Erfahrungsbericht (13:15 bis 14:00 Uhr):

Frederik Kramer, initOS GmbH & Co. KG

Big Data zur Unterstützung des Deployment-Prozesses
- *Suche von Mustern und Anomalien*
- *Automatisierte Analyseszenarien*

Keynote im Rahmen der ECC-Tagung (14:00 Uhr)

Holger Fritzinger, Vice President Mobile Solutions & Innovation, SAP AG

Innovationen kritisch hinterfragen
- *Architekturen im Mobile Computing*
- *Auswirkungen auf Data Science*

Session 4 – Impulsvorträge (15:15 bis 16:45 Uhr):

Dr. Thomas Koch, Die Schweizerische Post AG:
Pragmatismus als Erfolgskriterium (Kundenstory)

Jochen Jörg, MarkLogic: *Datenstrategie Big Data (Kundenstory)*

Marcus Zieger, Zalando: *Performance als Enabler (Kundenstory)*

Wolfgang Schwab, SAS: *Big Data im Kampagnencontrolling (Kundenstory)*

Session 5 (16:45 bis 17:15 Uhr):

Markus Bauer, UFD AG, Andreas Schmietendorf, HWR Berlin

Moderierte Abschlussdiskussion

Die korrespondierenden Vorträge der Referenten werden den Teilnehmern über die Webseite der ceCMG zur Verfügung gestellt. Ergebnisse entsprechender Diskussionsrunden werden zeitnah im Internet publiziert. Änderungen am Programm sind unter Vorbehalt möglich. Für Verpflegung vor Ort wird gesorgt. Für die Teilnahme an der Veranstaltung ist eine kostenpflichtige Anmeldung zur Enterprise Computing Conference (ECC 2016) erforderlich. Für Mitglieder der ceCMG-, DASMA-, GI- und ASQF gilt eine reduzierte Teilnahmegebühr. Über die Teilnahmegebühr erhalten Sie eine Rechnung der ceCMG e.V. (Central Europe Computer Measurement Group).

Veranstaltungsort: *Lindner Park-Hotel Hagenbeck Hamburg*

Weiteren Informationen und Anmeldung unter: http://www.cecmg.de

Kontakt: Susanne Mund – sekretariat@cecmg.de

Mobile Computing & API Management Workshop

Mobile Apps und APIs sind die Themen des Jahres 2016. Im Konferenz-Workshop "Mobile Computing & API Management" erfahren Sie, was Sie für fundierte Entscheidungen wissen müssen.

Kunden und Mitarbeiter erwarten hochwertige mobile Dienste. Jede Organisation muss sich daher den Herausforderungen der mobilen Welt stellen. Dazu zählen Fragestellungen der **Strategie**, des Geschäftsmodells und der zugrundeliegenden **Technologien**. Wir teilen unsere Erfahrung mit Ihnen, damit Sie wissen, worauf es ankommt!

Technologien und Geschäftsmodelle sind auch die Anknüpfungspunkte zum **API Management**. Die strukturierte Bereitstellung von Daten und Diensten in Form von programmierbaren Schnittstellen entwickelt sich im Rahmen der **API Economy** und aufgrund von Gesetzesänderungen zu einer wesentlichen neuen Anforderung an IT-Systeme. Aber wie gestaltet man sichere APIs und warum sollte man seine Anwendungen überhaupt erst öffnen?

Gemeinsam mit unseren Fach-Experten und Ihnen werden wir diese und weitere Themen in **Vorträgen, Fallstudien und Diskussionsrunden** kritisch hinterfragen und bewerten.

Agenda

Alle Programmpunkte enthalten ca. 10-15 Minuten Diskussionsanteil. Ihre Reihenfolge und Themen können sich noch geringfügig ändern. Die Keynotes finden im Plenum der ECC 2016 statt.

09:15- 10:00 **Uhr**	**KEYNOTE** **Management von Cognitive-Computing-Systemen** Dr. Wolfgang Hildesheim, IBM Watson-Group
10:15- 10:30 **Uhr**	Einführung in den Workshop
10:30- 11:00 **Uhr**	**VORTRAG** **Mobile Computing und APIs – Wie die Faust auf's Auge** (30 min) Prof. Dr. Cornelius Wille
11:00- 11:30 **Uhr**	**ERFAHRUNGSBERICHT** **Von SOA zum API Management bei Energielieferanten** (30 min) Dr. Florian Marquardt
11:30- 12:00 **Uhr**	**VORTRAG** **API-Management: Viel mehr Denke als Technik!** (30 min) Dr. Frank Simon
12:00 Uhr	**Pause** (15 min)
12:15- 13:00 **Uhr**	**VORTRAG** **Mobile Apps und APIs mit IBM Bluemix** (45 min)
13:00 Uhr	**Mittagspause** (1 h)
14:00 - 14:45 **Uhr**	**KEYNOTE** **Innovationen kritisch hinterfragt** Holger Fritzinger, Vice President Mobile Solutions & Innovation, SAP AG
14:45 Uhr	**Kaffeepause** (15 min)
15:00- 15:45 **Uhr**	**DEMO** **Apps für Alle: Möglichkeiten und Grenzen mobiler Technologien** (45 min) Sandro Hartenstein
15:45- 16:30 **Uhr**	**FRAGEN UND ANTWORTEN** **Mobile Computing & API Management** (45 min) Moderation: André Nitze
16:30 Uhr	Zusammenfassung und Abschluss

Anmeldung

Melden Sie sich also gleich zum Workshop "Mobile Computing und API Management" am 14.04.2016 im Rahmen der zweitägigen Enterprise Computing Conference 2016 in Hamburg an! Für die Teilnahme an der Veranstaltung ist eine kostenpflichtige Anmeldung zur Enterprise Computing Conference (ECC 2016) erforderlich. Für Mitglieder der ceCMG, DASMA, GI und ASQF gilt eine reduzierte Teilnahmegebühr. Über die Teilnahmegebühr erhalten Sie eine Rechnung des ceCMG e.V. (Central Europe Computer Measurement Group).

Veranstaltungsort:

LINDNER Park-Hotel, Hamburg, Hagenbeckstraße 150 , 22527 Hamburg

Weitere Informationen und Anmeldung zum Workshop unter:

https://mobile-quality-research.org/workshop/

Kontakt: Susanne Mund – sekretariat@cecmg.de, André Nitze – andre.nitze@hwr-berlin.de

MetriKon 2015, 5.-6.11.2015 Köln

Seufert, M.; Ebert, C, Fehlmann, T.; Pechlivanidis, S.; Dumke, R. R.:

MetriKon 2015 - Praxis der Softwaremessung
Tagungsband des DASMA Software Metrik Kongresses
5. - 6. November 2015, IBM, Köln

Shaker Verlag, Aachen, 2015 (272 Seiten)

Konferenzinhalt

Hauptvorträge:

- **Pekka Forselius, 4SUM partner & FISMA, Finnland:** *Triangle Benchmarking - a new and easy way to realize your performance*
- **Rini van Solingen, TU Delft, Niederlanden:** *Value Measurement in Agile: Where are the numbers?*
- ***Stefan Riedel, IBM Köln:*** Die nächste Epoche hat längst schon begonnen

Konferenzbeiträge:

A. Deuter, J. Dreyer: Reversed-GQM: Ein Ansatz zur Wiederverwendung von Kennzahlen

Fabienne Auer et al.: Ausarbeitung von Kennzahlen zur Messung und Analyse der Cost of Non-Quality

R. Dumke: COSMIC Function Points: Erweiterungen und Trends

T. Liedtke: Wie gut schätzen Schätzer?

S. Herden et al.: Einführung eines Frameworks für die Bewertung eines Industrie 4.0 Protfolios

A. Vasileva, D. Schmedding: Integration von Qualitätsaspekten in einen Entwicklungsprozess

S. Hartenstein, H. Könnecke: Metrics for Evaluation of Trustworthiness-By-Design Software Development Processes

T. Fehlmann, E. Kranich: Prioritizing Functional and Nonfunctional Requirements in an Agile Software Development Environment

A. Fiegler et al.: Qualitätsbemessung von ITIL Prozessen in Cloud-Systemen am Beispiel der Lern- und Entropierate

J. Hentschel et al.: Towards Optimal Server License Balancing in a Virtual Server

P. Jansen: Field Defect Predictability

A. Schmietendorf: "API economy" Ergebnisbericht einer empirischen Untersuchung von API-Serviceverzeichnissen und Serviceangeboten

C. Ebert: Benchmarking - Experiences and Guidelines for Improvement

K. Wille et al.: Benutzbarkeit und Metriken unter dem Aspekt mobiler Anwendungen

A. Nitze: Prozessqualitätsmetriken zur Risiko-gebtriebenen Entwicklung mobiler Applikationen

R. Neumann et al.: Efficiency of Scalable Tile Rendering Based on Apache Hadoop

L. Guzman et al.: Setting Up a Research Software Factory in the Oil and Gas Domain

A. Schmietendorf: Wie kann die Software-Messung von den Möglichkeiten einer Big Data Lösung profitieren?

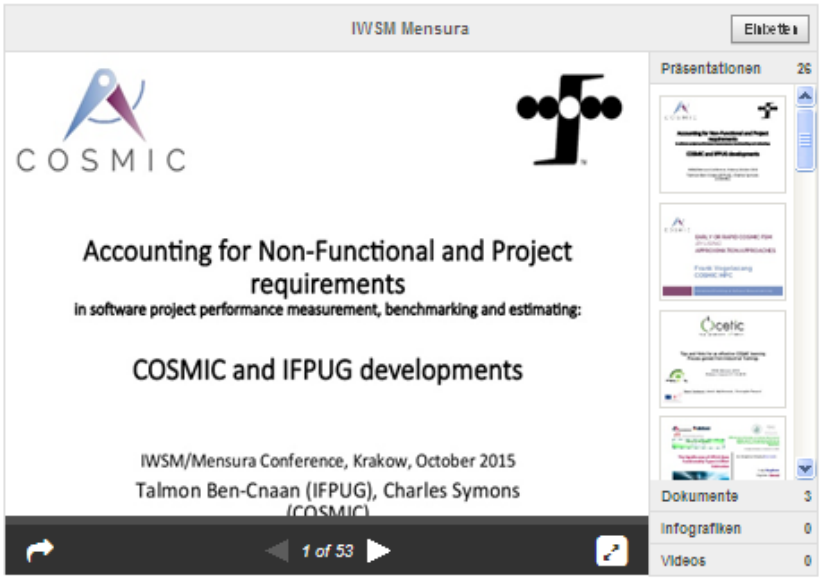
F. Vogelezang: Best Practices in Software Cost Estimation

C. Gencel, L. F. Sanz: A Novel Decision Making Platform for Managing Tradeoffs between Quality, Cost and Time

DASMA Diplomarbeiten-Preisträger

In diesem Jahr wurde der DASMA-Preis an Harald Foidl (Universität Innsbruck) mit dem Titel "The Usage of Quality Models in Risk-based Testing" vergeben.

SLIDESHARE:



Accounting for non functional and project requirements - cosmic and ifpug development - talmon ben-cnaan - charles symons 33 views

SPRINGER BOOK:

Book

Lecture Notes in Business Information Processing

Volume 230

Software Measurement

25th International Workshop on Software Measurement and 10th International Conference on Software Process and Product Measurement, IWSM-Mensura 2015, Kraków, Poland, October 5–7, 2015, Proceedings

Editors: Andrzej Kobyliński, Beata Czarnacka-Chrobot, Jarosław Świerczek

ISBN: 978-3-319-24284-2 (Print) 978-3-319-24285-9 (Online)

Book Contents

A Suite of Rules for Developing and Evaluating Software Quality Models . . . 1
Anas Bassam AL-Badareen, Jean-Marc Desharnais, and Alain Abran

The Effects of Duration-Based Moving Windows with Estimation by Analogy 14
Sousuke Amasaki and Chris Lokan

Software or Service? That's the Question! 30
Luigi Buglione, Alain Abran, Christiane Gresse von Wangenheim, Fergal Mc Caffery, and Jean Carlo Rossa Hauck

A Process to Improve the Accuracy of MkII FP to COSMIC Size Conversions: Insights into the COSMIC Method Design Assumptions 46
Aveek Dasgupta, Cigdem Gencel, and Charles Symons

Applying Manufacturing Performance Figures to Measure Software Development Excellence 62
Andreas Deuter and Hans-Jürgen Koch

Quantitative Functional Change Impact Analysis in Activity Diagrams: A COSMIC-Based Approach 78
Mariem Haoues, Asma Sellami, Hanêne Ben-Abdallah, and Nourchène Elleuch Ben Ayed

Application of Function Points and Data Mining Techniques for Software Estimation - A Combined Approach 96
Przemysław Pospieszny, Beata Czarnacka-Chrobot, and Andrzej Kobyliński

Functional Size Measurement for Processor Load Estimation in AUTOSAR . . . 114
Hassan Soubra, Alain Abran, and Mehdi Sehit

Selecting the Right Visualization of Indicators and Measures – Dashboard Selection Model 130
Miroslaw Staron, Kent Niesel, and Wilhelm Meding

Measurement-as-a-Service – A New Way of Organizing Measurement Programs in Large Software Development Companies 144
Miroslaw Staron and Wilhelm Meding

Designing an Unobtrusive Analytics Framework for Monitoring Java Applications 160
Sampo Suonsyrjä and Tommi Mikkonen

A Functional Software Measurement Approach to Bridge the Gap Between Problem and Solution Domains. 176
Erdir Ungan and Onur Demirörs

Improving the COSMIC Approximate Sizing Using the Fuzzy Logic EPCU Model 192
Francisco Valdés Souto and Alain Abran

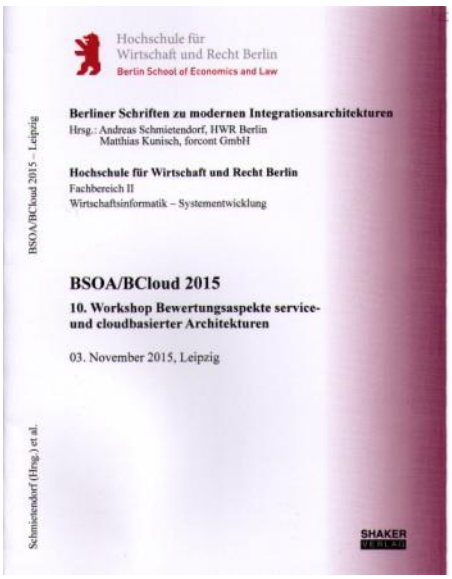
Author Index 209

Bewertungsaspekte service- und cloudbasierter Architekturen (BSOA/BCloud2015)

detaillierter Workshopbericht

Andreas Schmietendorf[+], *Frank Simon*[#]

[+]Hochschule für Wirtschaft und Recht Berlin
Email: andreas.schmietendorf@hwr-berlin.de
[#]BLUECARAT AG
Email: frank.simon@bluecarat.de



1. Hintergründe zur Veranstaltung

Programmierschnittstellen, häufig als Application Programmable Interface (API) bezeichnet, galten lange Zeit als eine ausschließlich dem Software Engineering, also klar der Technik zugeordnete Themenstellung. Mit der weltweiten Akzeptanz sozialer Medien, dem allgegenwärtigen Einsatz mobiler Technologien und dem Internet der Dinge steigt der erwartete Interoperabilitäts-Grad in einem nie da gewesenen Ausmaß. Die dahinter stehenden Schnittstellen werden dabei immer offenkundiger, in dem sie entweder direkt in der GUI angeboten werden (z.B. Login via Facebook) oder wenigstens separat hinzu installiert werden können (z.B. Social Media Connectoren). In diesem Zusammenhang fällt immer häufiger der Begriff der API economy, wobei dann ein geschäftszentrierter Servicebegriff im Mittelpunkt des Interesses steht. Zu beobachten ist ein neues Verständnis, mit dem webbasierte Programmierschnittstellen (Web-oriented Architecture, WOA) zum geschäftlichen Erfolg eines Unternehmens beitragen können. Allgemein bekannte Beispiele finden sich bei Amazon, Google, Salesforce oder auch Twitter. Der 10. Workshop ging u.a. der Frage auf den Grund, was wirklich neu an einer WOA ist und welche Lessons-Learned wir aus SOA- und Cloudeinführung dringend bei einer WOA Einführung berücksichtigen sollten. Im Vorfeld des Workshops wurde zum Einreichen von Beiträgen im Diskurs folgender Themenschwerpunkte aufgerufen [Schmietendorf/Kunisch 2015]:
Herausforderungen für API-Anbieter:

- Gestaltungsprinzipien für erfolgreich einzusetzende Web-APIs,
- Semantische Implikationen branchenspezifischer Web-APIs,
- Über APIs Daten für Big Data-Lösungen bereitstellen,
- …

Herausforderungen für API-Händler:

- Mehrwertpotentiale aus Sicht verschiedener Interessenten,
- Nutzungs- und Vertragsbedingungen für Web-APIs,
- Anforderungen an ein erfolgreiches API-Management,
- …

Herausforderungen für API-Konsumenten:

- Identitätenflut: Von der geschützten SOA zur freien WOA,
- Sicherheits-, Qualitäts- und Compliance-Anforderungen,
- Verwendung von Web-APIs im Hintergrund mobiler Apps.
- …

2. Beiträge des Workshops

Die im Folgenden kurz zusammengefassten Beiträge konzentrierten sich auf wirtschaftliche, organisatorische, qualitative und technische Fragen angebotener und konsumierter Serviceschnittstellen. Der Tagungsband zum Workshop enthält die entsprechenden Artikel (vgl. [Schmietendorf/Kunisch 2015]).

Olaf Resch: API-economy – eine Situationsbestimmung

Im Mittelpunkt des Beitrages stehen die ökonomischen Aspekte der API economy anhand eines sehr konkreten Beispiels. Besonders hervorgehoben und zugleich als Risikoaspekt eingeführt werden die vielfältigen Geschäftsbeziehungen erläutert, die im Diskurs eines solchen „Kompetenznetzwerks“ entstehenden können, aber nicht in jedem Fall zu einem funktionierenden API-Ecosystem führen. Dabei sieht der Autor die Dynamik und Anonymität als zentrale Faktoren der API-economy, für die es noch keine Musterlösung gibt. Unterfüttert werden die Aussagen mittels empirischer Analysen eines konkreten Anwendungsfalls als Consumer eines Ecosystems.

Frederik Kramer, Klaus Turowski: Der wirtschaftliche Nutzen weborientierter Architekturen – Eine wirtschaftliche Untersuchung am Beispiel eines E-Commernce-KMU

Die Entscheidung zur Auswahl und Integration von Schnittstellen wird durch vielfältige ökonomische und technische Faktoren beeinflusst. Mit Hilfe einer industriellen Fallstudie wird auf konkrete Problembereiche zu implementierender Schnittstellen eingegangen. Auf der Grundlage dieser Erfahrungen wird ein Modell zur Ermittlung des wirtschaftlichen Nutzens einzusetzender Schnittstellen vorgeschlagen.

Tobias Kraft, Frank Simon: Das Schnittstellen Paradoxon: Weniger Eigenentwicklung führt nicht zu weniger Testaufwand

Der Einsatz vielfältiger Schnittstellen innerhalb eines implementierten Systems führt zu besondern Anforderungen an die Qualitätssicherung. Im Zusammenhang mit dem Testen bereitet insbesondere die zeitgerechte und für Testzwecke beliebig konfigurierbare Bereitstellung der benötigten Testinfrastruktur Probleme. Eine effektive Problemlösung ist hierbei die sogenannte Servicevirtualisierung, mittels der im Test benötigte Schnittstellen intelligent, zentral und effizient simuliert werden können.

Sabine Wieland: 10 Gebote für Flexibilität und Sicherheit

Die Autorin identifiziert die Wiederverwendung von Software bzw. Services als Fluch und Segen zugleich. Probleme bereiten insbesondere die Einhaltung qualitativer Anforderungen, wofür auf beeindruckende Sicherheitsmängel Bezug genommen wird. In Anlehnung an die 10 heiligen Gebote des Christentums werden sehr plakativ Gebote (Werte und Prinzipien) des Software-Engineerings aus diesen Geboten abgeleitet, mit teilweise überraschenden Parallelen.

Anja Fiegler, Sebastian Herden, Andre Zwanziger, Sebastian Kiepsch, Reiner Dumke: Die Betrachtung und Messbarkeit der Skalierung von Cloud-Systemen, SOA und Microservices

Zumeist beziehen sich Betrachtungen zur Skalierbarkeit entweder auf die Ebenen der Infrastruktur und Plattform oder aber auf die Ebene der Anwendung. Die Bereitstellung von SOA- und Micro-Services bedürfen allerdings einer integrierten Sicht der kompletten Interaktionskette. Zur Bewertung der Wirtschaftlichkeit und Messbarkeit der Skalierung wird ein entsprechend ganzheitlicher Ansatz hergeleitet.

André Nitze: Entwicklung mobiler Applikationen in der API Economy

Der Beitrag greift die Wechselwirkungen zwischen weborientierten Architekturen und mobilen Applikationen auf. Bei der Entwicklung plattformübergreifender APPs identifiziert der Autor einen Wandel hin zu „API zentrischen“ Vorgehensweisen. Dementsprechend werden die Funktionalitäten verfügbarer APIs in den Mittelpunkt der Entwicklung gestellt, was sich in der Klassifikation mobiler APIs niederschlägt.

Jan Hentschel, Robert Neumann, Andreas Schmietendorf: Data Nebula – what is wrong with data lakes?

Die bei Data Lakes durchgeführte Aufnahme, Speicherung und Analyse roher und unverschlüsselter Daten aus unterschiedlichen Quellsystemen impliziert ein hohes Sicherheitsrisiko. Das Konzept des Data Nebula verändert diesen Ansatz, so dass nur verschlüsselte Daten abgelegt werden. Dem entsprechend werden nur Signaturen abgelegt, die mit Hilfe von Algorithmen des maschinellen Lernens ausgewertet werden.

Sandro Hartenstein: Toolunterstütztes Messen der Vertrauenswürdigkeit von Webapplikationen

Der Beitrag geht auf die Anforderungen, die Entwicklung und die Nutzung einer Metrikendatenbank ein, welche die Messung der Vertrauenswürdigkeit von Webapplikationen unterstützt. Neben der Analyse existierender Mess- und Toolansätze geht der Autor insbesondere auf einen selbst eigen entwickelten Ansatz zur toolgestützten Vermessung von Websystemen ein.

3. Ergebnisse der Diskussionsrunde

Im Rahmen einer durchaus kontrovers geführten Diskussionsrunde wurde auf die Implikationen von bereits etablierten Konzepten wie SOA, Cloud und Co auf das neue Konzept einer WOA eingegangen. Viele Diskussionen haben in allen Bereichen immer wieder unterschiedliche Verständnisse der Grundbegriffe offenkundig werden lassen. Nach einer entsprechenden Abgrenzung wurden dann vor allen Dingen die folgenden organisatorischen Implikationen auf eine weborientierte Architekturgestaltung arbeitet:

- Webbasierte APIs implizieren viele Herausforderungen, die bereits bei serviceorientierten Architekturen anstanden. Nicht selten wird daher von „SOA im Kleinen“ gesprochen, wobei sich das „klein“ allerdings eher auf die Leichtgewichtigkeit denn des Business Impacts bezieht.

- Bei klein- und mittelständischen Unternehmen (KMUs) kann ein Skalieren von selbst angebotenen APIs nicht automatisch angenommen werden. Hier kann sich die Begrenztheit der eigenen Ressourcen sehr schnell limitierend auswirken.

- Der Bedarf an zu konsumierenden APIs kann insbesondere bei KMUs beobachtet werden, da mit deren Hilfe eine Konzentration auf die Kernkompetenzen sehr effektiv erfolgen kann.

- Großkonzerne zeichnen sich durch eine gewisse Änderungsfeindlichkeit bzw. Innovationsträgheit aus, was nicht zuletzt aus einem Schutz der eigenen Ressourcen und der etablierten Machtverhältnisse resultiert.

- Während serviceorientierte Architekturen eher eine infrastrukturelle Maßnahme (kein Enabler für zusätzliches Geschäft) darstellen, bieten fachliche APIs geschäftliches Potential: Sie liefern neue Absatzmärkte für bereits existierende Produkte als auch die Möglichkeit, weitere Zwischenprodukte der eigenen Wertschöpfungskette anzubieten

- Besonders interessant ist die Verwendung von APIs bei evolutionären bzw. pragmatisch agilen Vorgehensweisen zur Softwareentwicklung. Hier zählt der große Vorteil, dass die Bereitstellung eines zeitnahen Feedbacks direkt unterstützt wird (fast fail).

Entsprechend Conways Law folgen die Systemgestaltung und daraus resultierende Systemarchitekturen den Prinzipien der eigenen Organisation.

> “Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure.” (Quelle des Zitats: [Conway 1968])

Im Umkehrschluss kann damit unterstellt werden, dass eher technisch getriebene Initiativen zum Scheitern verurteilt sind, sobald sie Auswirkungen auf die fachliche Architektur haben. Dementsprechend gilt, es die Interessen der Business Analysten konkreter Unternehmen stärker in die Überlegungen der BSOA/BCloud-Community einzubeziehen, da in diesem

Kontext die echten, fachlich begründeten Bedürfnisse für API-Angebote bzw. zu konsumierende APIs entstehen. Diese neue Interdisziplinarität scheint wesentlich für die erfolgreiche Einführrung von einer WOA und die effektive Ausweitung der BSOA/BClod-Community. Dort könnte dann auch die spannende aufgekommene Frage analysiert werden, wie der Grad des organisatorischen Einflusses auf die Gestaltung weborientierter Architekturen empirisch nachzuweisen ist.

4. Weitere Informationen

Auch für das Jahr 2016 ist die Durchführung des BSOA/BCloud-Workshops in Berlin vorgesehen. Weiterführende Informationen werden unter der folgenden URL im Internet bereitgestellt:

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

5. Quellenverzeichnis

[Conway 1968] Conway, M. E.: How Do Committees Invent?, Thompson Publications, Inc., April 1968, download: http://www.melconway.com

[Schmietendorf/Nitze 2015] Schmietendorf, A.; Nitze, A.: API economy – Qualität als Schlüssel zum Erfolg, in SQ Magazin 36, Arbeitskreis Software-Qualität und -Fortbildung (ASQF), S. 16-17, September 2015

[Schmietendorf/Kunisch 2015] Schmietendorf, A.; Kunisch, M. (Hrsg.): Tagungsband BSOA/BCloud 2015, in Berliner Schriften zu modernen Integrationsarchitekturen, Shaker-Verlag, Aachen, November 2015

Dank

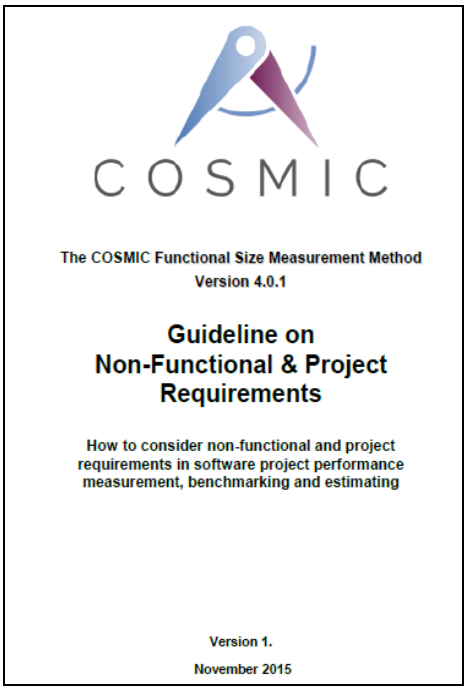
Seit Gründung der BSOA-Initiative im Jahre 2006 erfährt diese vielfältige Unterstützung aus dem industriellen und akademischen Umfeld. Ein besonderer Dank geht an die forcont GmbH als Gastgeber und Hauptsponsor der diesjährigen Veranstaltung. In diesem Zusammenhang sei auch Frau Kerstin Krohn, ebenfalls von der forcont GmbH, für ihre umfangreiche organisatorische Unterstützung gedankt. Ebenso danken wir der Ultra Tendency UG (Magdeburg) für das Sponsoring gedankt. Organisatorische Unterstützung bei den vielfältig eingesetzten Websystemen zur Bewerbung der Veranstaltung erfuhr der Workshop von Herrn Dr. Dmytro Rud von der Roche Diagnostics AG/Schweiz und von Herrn Kevin Grützner von der HWR Berlin.

Organisation

Der Workshop wurde in Kooperation zwischen der Hochschule für Wirtschaft und Recht Berlin, der Otto-von-Guericke-Universität Magdeburg (Softwaremesslabor) unter der Schirmherrschaft der ceCMG (Central Europe Computer Measurement Group) veranstaltet. Darüber hinaus erfährt die BSOA/BCloud-Initiative Unterstützung durch die GI (Gesellschaft für Informatik - Fachgruppe Softwaremessung- und Bewertung), die DASMA (Deutschsprachige Interessengruppe für Softwaremetrik und Aufwandsschätzung) und durch die ASQF (Arbeitskreis Software-Qualität und Fortbildung).

COSMIC Document from November 2015

See our new documents in our home page ***cosmic-sizing.org***:
(see **http://cosmic-sizing.org/cosmic-publications/overview/)**



COSMIC

The COSMIC Functional Size Measurement Method
Version 4.0.1

Guideline on Non-Functional & Project Requirements

How to consider non-functional and project requirements in software project performance measurement, benchmarking and estimating

Version 1.

November 2015

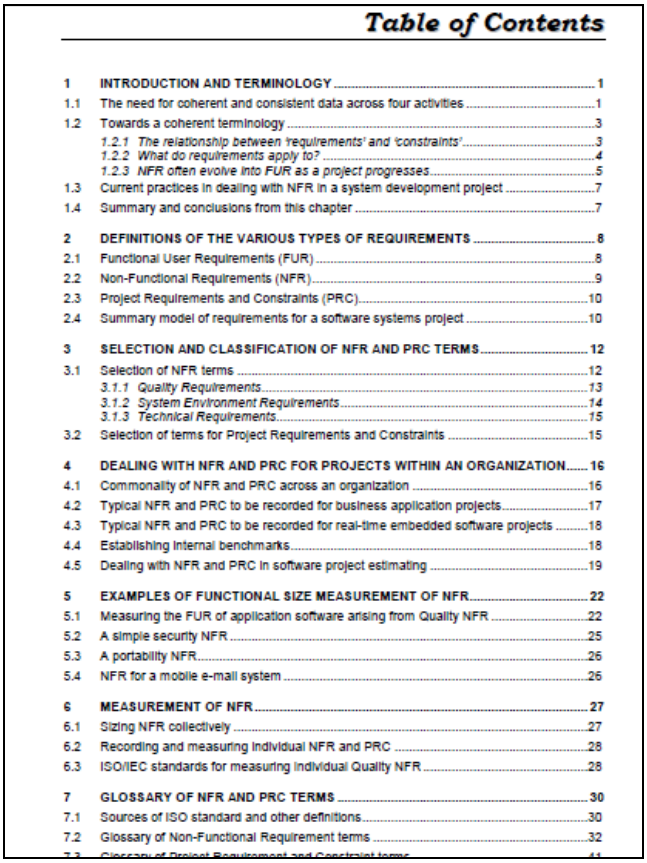


Table of Contents

1 INTRODUCTION AND TERMINOLOGY 1
1.1 The need for coherent and consistent data across four activities 1
1.2 Towards a coherent terminology 3
1.2.1 The relationship between 'requirements' and 'constraints' 3
1.2.2 What do requirements apply to? 4
1.2.3 NFR often evolve into FUR as a project progresses 5
1.3 Current practices in dealing with NFR in a system development project 7
1.4 Summary and conclusions from this chapter 7

2 DEFINITIONS OF THE VARIOUS TYPES OF REQUIREMENTS 8
2.1 Functional User Requirements (FUR) 8
2.2 Non-Functional Requirements (NFR) 9
2.3 Project Requirements and Constraints (PRC) 10
2.4 Summary model of requirements for a software systems project 10

3 SELECTION AND CLASSIFICATION OF NFR AND PRC TERMS 12
3.1 Selection of NFR terms 12
3.1.1 Quality Requirements 13
3.1.2 System Environment Requirements 14
3.1.3 Technical Requirements 15
3.2 Selection of terms for Project Requirements and Constraints 15

4 DEALING WITH NFR AND PRC FOR PROJECTS WITHIN AN ORGANIZATION 16
4.1 Commonality of NFR and PRC across an organization 16
4.2 Typical NFR and PRC to be recorded for business application projects 17
4.3 Typical NFR and PRC to be recorded for real-time embedded software projects 18
4.4 Establishing internal benchmarks 18
4.5 Dealing with NFR and PRC in software project estimating 19

5 EXAMPLES OF FUNCTIONAL SIZE MEASUREMENT OF NFR 22
5.1 Measuring the FUR of application software arising from Quality NFR 22
5.2 A simple security NFR 25
5.3 A portability NFR 26
5.4 NFR for a mobile e-mail system 26

6 MEASUREMENT OF NFR 27
6.1 Sizing NFR collectively 27
6.2 Recording and measuring individual NFR and PRC 28
6.3 ISO/IEC standards for measuring individual Quality NFR 28

7 GLOSSARY OF NFR AND PRC TERMS 30
7.1 Sources of ISO standard and other definitions 30
7.2 Glossary of Non-Functional Requirement terms 32

. . .

5. Examples of Functional Size Measurement of NFR

Quality NFR Group	The initial Non-Functional Requirement (NFR)	Example functionality that may evolve from the NFR, whose size may be measured by COSMIC	The remaining part of the NFR that that cannot contribute to Functional Size
Data Quality	Accountability	Functionality that links a person responsible for maintaining data or software to changes	The specific accountability requirements statement
	Accuracy	Validation checks by R's of stored data. GUI elements that accept only valid data	The quantified accuracy targets
	Auditability	Functionality needed by auditors, e.g. X or W of hash totals; special enquiries	Adequate documentation
	Precision	Not measurable	The quantified precision targets
	Validation (of data)	See 'Accuracy'	E.g. legal requirements for independent audits. Need to store data within national boundary or 'safe haven'

Quality NFR Group	The initial Non-Functional Requirement (NFR)	Example functionality that may evolve from the NFR, whose size may be measured by COSMIC	The remaining part of the NFR that that cannot contribute to Functional Size
System Performance	Response time	Functionality to import external data so that it is available for immediate use.	The quantified response time target; Fast hardware; Low-level programming languages
	Transaction rate	Functionality to measure the transaction rate (usually measured by the OS, not by the application to which the NFR applies)	The quantified transaction rate target
Compatibility	Co-existence	Achieved via the OS?	The specific co-existence requirements
	Compatibility	Functionality of interfaces, achieved either via X/E exchanges or via shared persistent data, or via conversion functionality	The specific compatibility requirements
	Interoperability	See 'Compatibility'	The specific inter-operability requirements
Ease of use	Accessibility	See 'Usability'	The requirements for accessibility by specific classes of people; Provision of braille keyboards
	Aesthetics (of the UI)	Not measurable	The specific aesthetic requirements
	Customer satisfaction (software)	Not measurable	The customer satisfaction target, as measured by a specific technique.
	Learnability	Functionality to assist users to learn the system	The specific learnability targets
	Multi-lingual support	Functionality for the users to select the language and to enter E's and receive X's in multiple languages	The requirement to support named languages and their character sets
	Operability	Functionality to assist operators (maybe provided by the OS)	The specific operability requirements
	Usability	Graphical User Interface functions; Functionality to assist users, e.g. 'Help' functions	The specific usability requirements (e.g. 'must be usable by public with no training')

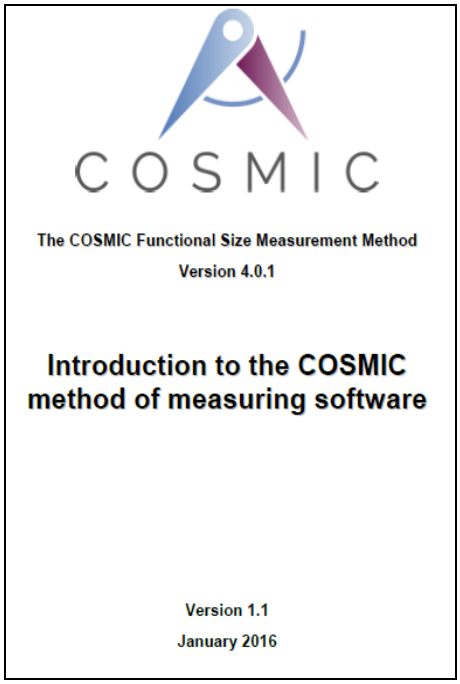
Quality NFR Group	The Initial Non-Functional Requirement (NFR)	Example functionality that may evolve from the NFR, whose size may be measured by COSMIC	The remaining part of the NFR that that cannot contribute to Functional Size
System reliability	Availability	Functionality to exploit multiple parallel processors to ensure high availability	The specific, quantified availability target; High reliability or fault-tolerant hardware
	Back-up	Back-up functionality, if not provided by the OS	The specific requirements for back-up of named files
	Dependability	See 'Availability', 'Fault tolerance', Recoverability'	The specific Dependability target
	Diversity	The diverse software solutions resulting from a Diversity NFR	The specific Diversity target
	Failure management	Functionality to assist failure management	Manual processes; Specific quantified failure management targets
	Fault tolerance	Functionality to exploit multiple parallel processors to ensure continuity when one or more processors fail	The specific target for continuity in spite of faults. Fault-tolerant hardware
	Recoverability	Functionality to recover data and resume processing following faults or interruptions	The specific recoverability target
	Reliability	See 'availability', 'security', maintainability', etc.	The specific Reliability target
	Safety	Functionality (as in safety-critical systems) to ensure human safety	The specific safety targets
Access control	Authenticity	Functionality that enables the identity of a person or a resource to be proven	The specific authenticity target. Finger-print recognition hardware
	Confidentiality	Functionality that protects data against unauthorized access	The specific confidentiality target
	Non-repudiation	Functionality that helps prove that an event or action did take place	The specific non-repudiation target
	Privacy	Functionality to control access to personal data	The specific privacy requirements
	Security	Functionality to control access to systems and/or data	The specific security requirements
	Usage mode (live vs training)	Functionality to provide both modes and to allow the user to access both modes	The specific requirements for different modes

Quality NFR Group	The initial Non-Functional Requirement (NFR)	Example functionality that may evolve from the NFR, whose size may be measured by COSMIC	The remaining part of the NFR that that cannot contribute to Functional Size
Maintainability	Adaptability	Functionality to enable the system to be adapted to different hardware, operational, etc., environments	The requirements for the specific environments for which the software must be adaptable
	Maintainability	Functionality to enable the system to be maintained without re-programming, e.g. via parameterization	The specific maintainability requirements
	Reusability	The functionality that must be made re-usable	The specific re-usability requirements
	Re-use type	Functionality of re-used artefacts	The specific re-use types required
Ease of deployment	Installability	Functionality to facilitate ease of installation	The specific installability requirements
	Portability	'Middleware' functionality to enable portability across multiple DBMS or OS software	The requirements for the specific environments across which the software must be portable
System or software architecture or design	Architecture/ Design	Software can be measured in any layer of an architecture	Design requirements that do not affect FUR
	Interfaces	Functionality achieved either via X/E exchanges or via shared persistent data	The specific interface requirements
	Open source	It is irrelevant to FSM whether the software is open source or not but the functionality of such software may be measured	The specific open source requirements
	Operational processing mode	Any functionality can be measured regardless of its Operational Processing Mode	The specific operational processing mode requirement

▪ ▪ ▪

COSMIC Document from January 2016

(see **http://cosmic-sizing.org/cosmic-publications/overview/)**



COSMIC

The COSMIC Functional Size Measurement Method

Version 4.0.1

Introduction to the COSMIC method of measuring software

Version 1.1

January 2016

Table of contents

1 WHY MEASURE SOFTWARE SIZE? 8
1.1 Why would anyone want to 'measure' software? 8
1.2 Software size measurements have many other uses 8
1.3 Who typically benefits from these measurements? 9
2 HOW TO MEASURE SOFTWARE SIZE? 10
2.1 Like any other unit of measurement, you need standards 10
2.2 What are the most important ways of measuring software size? 10
2.3 Counting source lines of code 10
2.4 Measuring software requirements 11
2.5 Other ways of measuring software size 11
2.6 A closer look at software requirements – FUR and NFR 11
2.7 Skills needed for COSMIC Measurers 12
3 A BRIEF HISTORY OF FUNCTIONAL SIZE MEASUREMENT 13
3.1 How did it all start? 13
3.2 The International Organization for Standardization (ISO) steps in 13
3.3 COSMIC gets going 13
3.4 ISO's final word: 'Let the market decide' 14
4 A VERY BRIEF OVERVIEW OF THE COSMIC METHOD 15
4.1 Applicability of the method 15
4.2 The three phases of the COSMIC functional size measurement process 15
4.3 Phase 1: Measurement Strategy 15
4.4 Phase 2: Mapping 16
4.5 Phase 3: Measurement 16
5 COSMIC METHOD - THE MEASUREMENT STRATEGY PHASE 17
5.1 Why do we need a 'strategy'? 17
5.2 The five key strategy parameters to be determined 17
5.3 Software 'layers' 17
5.4 Examples of how the 'purpose' of a measurement affects the other measurement strategy parameters 18
5.5 What else should you think about before starting to measure? 19
6 COSMIC METHOD - THE MAPPING PHASE 21
6.1 The Generic Software Model 21
6.2 A key relationship: events / functional users / functional processes 22
6.3 The structure of FUR and of functional processes 23
6.4 Accounting for data manipulation 24
6.5 The four types of data movements 24
6.6 Persistent storage 24

. . .

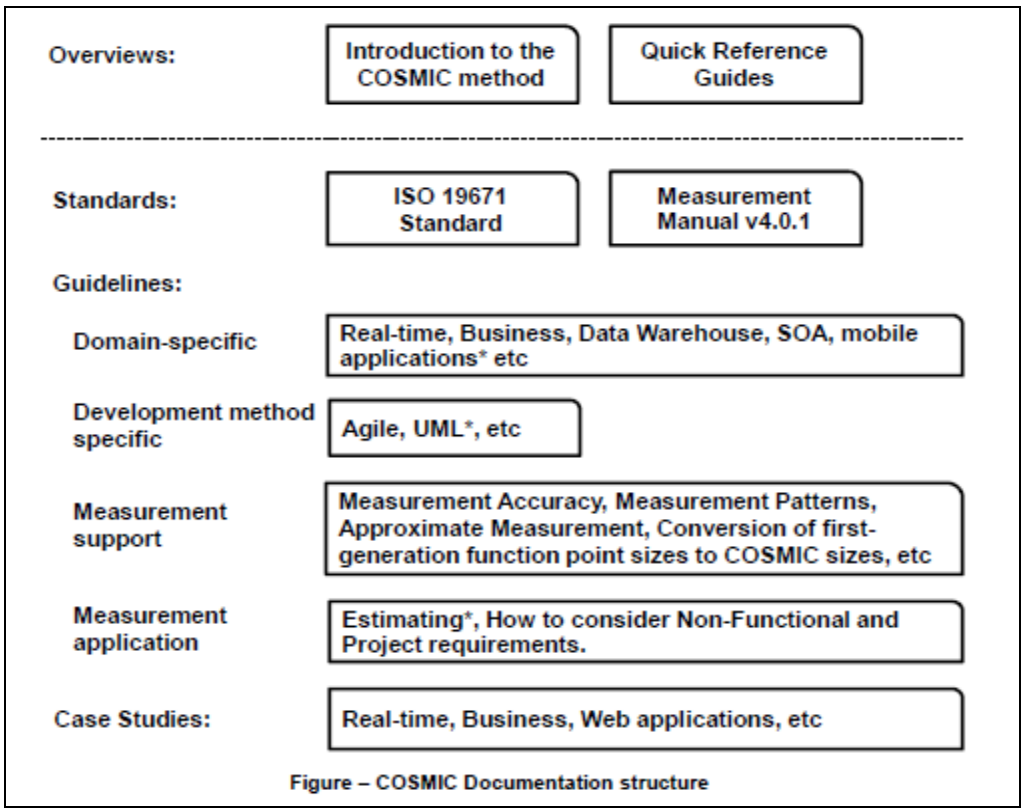


Figure – COSMIC Documentation structure

. . .

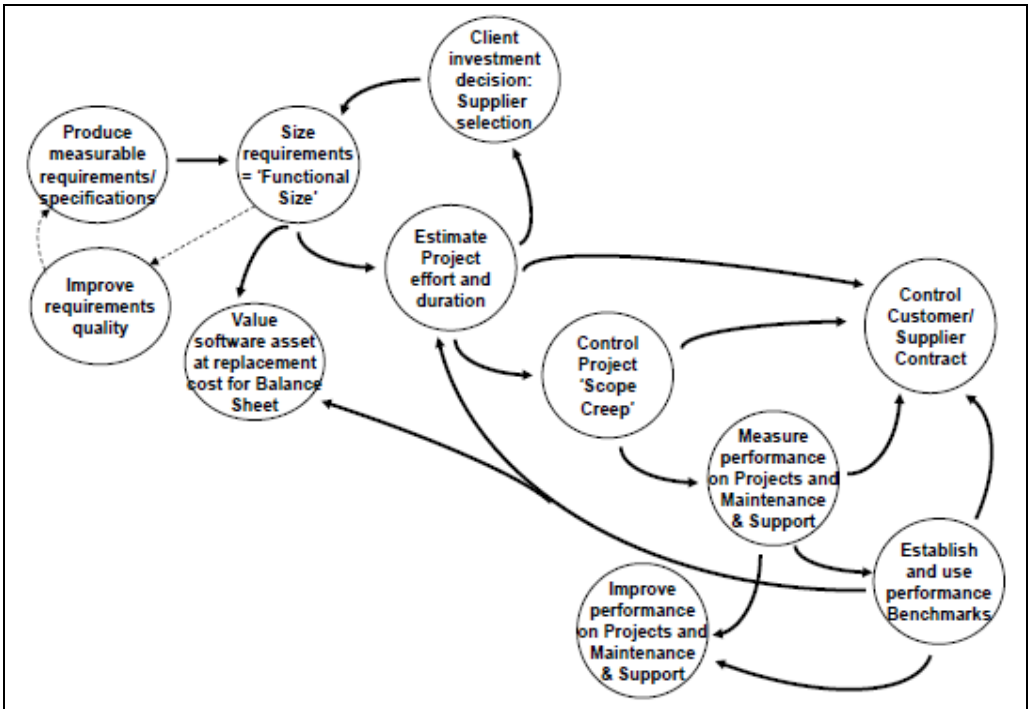


Figure 8.1 – Mind map of possible uses of functional size measurements

. . .

Effort Estimation with Story Points and COSMIC Function Points - An Industry Case Study

Christophe Commeyne, Alain Abran, Rachida Djouab

Abstract

In Agile software projects developed using the Scrum process, Story Points are often used to estimate project effort. Story Points allow comparison of estimated effort with actual effort, but do not provide a size of the software developed and, therefore, do not allow determination of the productivity achieved on a project nor comparison of performance across projects and across organizations. In contrast to Story Points, international standards on functional size measurement, such as ISO 19761 (COSMIC Function Points), provide explicit measurement of software size for both estimation and benchmarking purposes. This study reports on the performance of estimation models built using Story Points and COSMIC Function Points and shows that, for the organization in which the data was collected, the use of COSMIC Function Points leads to estimation models with much smaller variances. The study also demonstrates that the use of COSMIC Function Points allows objective comparison of productivity across tasks within a Scrum environment.

Keywords: Story Points, Planning Poker, Function Points, COSMIC, ISO 19761, Effort Estimation, Scrum

1. Introduction

From their beginnings in the early 2000s, the Agile approach and Scrum process have been adopted in a number of organizations [1-3]. The Planning Poker technique is often used in Scrum projects as an effort estimation technique: it is based on practitioners' opinions, and its measurement unit, called a Story Point, allows comparison of estimated effort with actual effort deployed for a completed project.

However, Story Points do not explicitly provide a size of the software developed, and therefore cannot be used to objectively determine the productivity achieved on a project, nor to compare performance across projects and across organizations. Furthermore, since the Planning Poker technique does not provide direct information on the size of the software to be delivered nor of what was actually delivered, neither the teams nor their management have any way of knowing if their estimates are way off or if their productivity across Scrum iterations is erratic.

In addition to Story Points, there are a number of international standards that provide well-documented methods for measuring the functional size of a piece of software, independently of the technologies and development processes. These ISO standards provide a sound basis for productivity and performance comparisons across projects, technologies and development processes. Of the five functional size measurement (FSM) methods adopted by ISO, four correspond to the first generation of FSM methods, while the most recent one, COSMIC – ISO 19761 [4-5], is the only one recognized as a second-generation method that has addressed the structural weaknesses of the first-generation FSM methods.

The data analyzed in this paper come from an industry software organization where the available post-iteration information indicated that the Story Point estimates were frequently off-target and productivity across iterations could not be determined or monitored. This paper reports on a case study where Scrum iterations were also measured using COSMIC Function Points as an avenue to overcome the estimation and productivity benchmarking issues. The study reports on the performance of estimation models built using both Story Points and COSMIC Function Points and investigates whether or not the use of COSMIC Function Points leads to estimation models with smaller variances.

The paper is structured as follows: Section 2 presents an overview of the Planning Poker technique and Story Points, and of the COSMIC measurement method. Section 3 gives the background for the case study (a set of 7 Scrum iterations and 24 tasks) and analyzes the performance of Planning Poker estimates. Section 4 presents the data collection with the COSMIC measurement method of this same set of iterations and tasks, as well as the information obtained on the unit effort and productivity ratios for these tasks. Section 4 presents the derived COSMIC-based estimation model and compares it to the Story Points-based estimates. Section 5 presents the summary.

2. Overview of Planning Poker and COSMIC - ISO 19761

This section presents an overview of the Planning Poker technique and of the COSMIC – ISO 19761 Function Points measurement method.

2.1 Planning Poker technique – Overview

The Planning Poker technique was initially described by Grenning [6] and popularized by Cohn [7-8]: effort is estimated in terms of units of work referred to as 'story points' quantifying the relative complexity. Scenarios are compared and person-days estimated according to team ability, experience and knowledge of the field. The key concepts of the Planning Poker technique are based on:

- a list of features to deliver (the product backlog)
- a set of cards based on the Fibonacci sequence: 0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100, These values correspond to the expected number of days for developing a particular feature X (in a 'story point'). Cohn indicates that such estimates are intuitively transformed into intervals of hours based on the perceived 'velocity' of the team [7-8].

Planning Poker is used by a team during a planning meeting to give estimates for the development of a feature by evaluating the user scenarios of the product backlog. Each story is sized in terms of Story Points representing an estimate of the effort in days [7-8]. Designed as a consensus-based technique (i.e. a Delphi-like approach), Planning Poker calls for the participation of all developers on the team (programmers, testers, database engineers, analysts, user interaction designers, and so on.) The product owner participates but does not estimate. The technique includes the following steps, which may vary when implemented in various organizations:

1. The product owner explains each story and its acceptance criteria in detail.
2. The team discusses the work involved and asks questions to the product owner.
3. Each team member secretly makes an estimate, in terms of Story Points.
4. Each team member selects a poker card that reflects their estimate of the effort required for that story.
5. Everyone shows their cards simultaneously.
6. The team members with the lowest and highest estimates explain the reasoning behind their estimates, and any misunderstanding of scope is resolved.
7. After discussion, the team members choose new cards to reflect their estimates based on the current discussion.
8. Agreement on the final number is taken as the team estimate.
9. The process is repeated for all the stories in the sprint backlog.

On the one hand, a number of benefits are directly derived from the consensus-building process of Delphi-like approaches:

1. Brings together practitioners from all disciplines involved in a software project to do the estimating.
2. Allows everyone to express themselves freely and promotes exchanges between the product manager and the development team.
3. Encourages open discussion rather than reliance on the opinion of the most influential or vocal member of the team. This allows the team to benefit from the experience of all team members.

On the other hand, some weaknesses are mostly related to the subjective nature of the estimating technique:

1. The estimation could vary when an identical backlog is provided to another Scrum team. This may cause conflict when multiple teams are working together on the same backlog.
2. Estimates will vary with skills and experience. Those more skilled and experienced would likely provide lower values than someone new to the field, which can skew the estimates.

It should also be noted that:

1. The estimate is made by direct consensus in terms of days (or hours), without explicitly or objectively sizing the product being estimated.
2. The team's 'velocity' is typically not based on historical data collected using international standards for measuring the size of software to be delivered.
3. Previous estimates are not reused in Scrum iterations; the team starts from scratch each time.
4. There is no process for analyzing the quality of the estimates and subsequently improving their accuracy.

Furthermore, there is a lack of well-documented empirical or experimental studies comparing Story Point estimates with actual project effort or comparing Story Points with other estimation techniques.

2.2 COSMIC Measurement Method - Overview

The COSMIC Function Points method is an international standard (ISO 19761 [4-5]) for measuring the functional user requirements of the software. The result obtained is a numerical value representing the functional size of the software itself, which can be used for benchmarking and estimation purposes. As required by ISO, COSMIC functional size is designed to be independent of any implementation decisions embedded in the operational artifacts of the software. This means that the functional user requirements can be extracted not only from software already developed but also from the software requirements before the software itself is implemented.

In COSMIC, a functional process is a set of data movements representing the functional user requirements for the software being measured. According to COSMIC, software functionality is embedded within the functional flows of data groups. Data flows can be characterized by four distinct types of movement of a data group:

- Entry (E) and Exit (X) data movements of a data group between the functional user of the software and a COSMIC functional process allow data exchange with a functional user across a software boundary.

- Read (R) and Write (W) data movements of a data group between a COSMIC functional process and persistent storage allow data exchange with the persistent storage hardware.

Each data movement, of any of the four types above, is assigned a single size expressed in COSMIC Function Points (CFPs), so that one data movement of one data group = 1 CFP.

The COSMIC measurement procedure consists of three phases:

1. The measurement strategy, which specifies the purpose and scope of the measurement.

2. The mapping phase, which aligns the software to be developed with the COSMIC generic model of software functionality and assigns a COSMIC size unit to each software functional requirement.

3. The measurement phase, which aggregates the individual measurements into a consolidated size measurement.

The measurement result corresponds to the functional size and is expressed in COSMIC Function Points (CFPs).

With this ISO standard, it is possible to obtain a measure of the functional size of a software application that is objective and reproducible by distinct measurers.

Once a baseline of completed projects has been measured with COSMIC, the team productivity (also referred to as velocity, in Scrum) can be derived based on historical data and estimation models can be built using various statistical techniques such as linear or non-linear regressions [9]. The COSMIC Group has also developed a Guideline document on how to apply COSMIC Function Points in the context of Agile projects [10].

3. Case study context and Story Points data

3.1 Industry context

The data were collected from an organization specializing in the development of IP solutions for security and surveillance [11]. It provides a platform that can be tailored to meet the needs of law enforcement agencies, schools, airports, casinos, retailers, and many others, and is used all over the world.

The development team follows the Scrum methodology with iterations ranging from three to six weeks. At the beginning of each iteration, all team members meet (four to eight hours, never more) and review with the team manager (who has the role of product owner) the list of change requests for the current iteration. These discussions include technical details about the changes to be made, candidate existing extension points (without having access to the source code) and expected impacts of these changes on the rest of the application. The objective of the meeting is to ensure that everyone has an understanding of the tasks in the current iteration.

In this organization, the team usually prepares its Story Point estimates of effort directly in terms of hours, without relying on any references or historical data.

Throughout an iteration, at the 15-minute daily morning meeting, each team member briefly describes the work of the previous day and what he or she expects to accomplish during the day. Each day, team members enter the number of hours spent on tasks. The number of hours allocated to a task are adjusted upwards if need be. The progress chart is presented daily by the Scrum Master, who, along with the product owner has the responsibility to shift features that could not be developed within the current iteration to another iteration. On the last day of the iteration, the product owner is shown the changes implemented during the iteration process. Depending on acceptance criteria and what is observed, the product owner determines whether the application can be sent to the testing team for validation and further verification.

When the task is completed, the Planning Poker estimates are compared with the actual effort and recorded on the team's intranet; however, within this organization, the estimates and actuals are not reused for subsequent iterations.

3.2 Data collection

For this case study, 24 tasks executed by the same team on the same application within a set of nine iterations were available for analysis, of which eight included functional requirements. Of the nine steps listed in section 2, only the following two were applied in the organization:

1. A first task was selected and its Story Points size was estimated in hours. Each team member offered a value selected from the Fibonnaci sequence: 0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100.

2. The team prepared its estimates of direct effort in terms of hours.

Table 1 presents, for each task of each iteration, the effort estimated with Story Points. The real effort, and the relative error of the Story Points estimates was calculated as follows [9]:

$$\text{Relative error (\%)} = \frac{\text{(Real effort – Estimated effort)}}{\text{Real effort}}$$

For the dataset of 24 tasks, the total estimated effort corresponds to 1,329 hours, with an average of 55 hours per task (Table 1, column 1).

The total real effort by task for this dataset corresponded to 1051 hours with an average of 44 hours per task (Table 1, column 2). However, with actual effort time of individual projects varying from five to 173 hours the standard deviation was 72 hours, making it of little value for estimation purposes.

The differences between the estimated and real effort (Table 1, column 3) were considerable, ranging from -3 to -76 hours underestimation for seven tasks, and from 1 to 83 hours overestimation for 17 tasks, making for very large estimation inaccuracies at the task level. It was also noted that 18 out of 24 tasks required an effort below 50 hours.

Table 1 Story Points: Estimated and Actual Hours (N= 24 tasks)

No.	Iteration	Task	(1) Estimated hours	(2) Actual hours	(3) Under/Over estimation (hours) (3)=(2) – (1)	(4) Relative Error (4)=(2)–(1)/(2)
1	1	1.1	97	173	-76	43,9%
2	1	1.2	179	96	83	-85,8%
3	2	2.1	28	68	-40	58,8%
4	2	2.2	36	33	3	-9,1%
5	3	3.1	23	22	1	-3,0%
6	3	3.2	110	58	52	-91,3%
7	4	4.1	76	39	37	-94,2%
8	4	4.2	84	39	45	-116,1%
9	4	4.3	61	26	35	-137,3%
10	4	4.4	90	61	29	-47,6%
11	6	6.1	34	63	-29	45,6%
12	6	6.2	10	5	5	-102,0%
13	6	6.3	104	36	68	-189,7%
14	6	6.4	16	7	9	-141,8%
15	7	7.1	31	36	-5	12,7%
16	7	7.2	76	41	35	-84,7%
17	8	8.1	49	38	11	-28,7%
18	8	8.2	9	36	-27	76,0%
19	8	8.3	29	21	8	-37,7%
20	8	8.4	13	29	-16	55,2%
21	8	8.5	34	32	2	-7,2%
22	8	8.6	33	36	-3	8,7%
23	8	8.7	30	18	12	-66,7%
24	9	9.1	77	40	37	-92,9%
			Total= 1329 hrs	Total= 1051 hrs	Total error (+ & -) = 278 hrs	
			Average: 55 hrs	Average: 44 hrs	Absolute error: 669 hrs or 66%	

3.3 Analysis of the estimation performance of Story Points

The relative error (RE) for each task is presented in the rightmost column of Table 1 and graphically in Figure 1:

- Four projects were estimated within 10% of real hours,
- One project was estimated within 10% and 25% of real hours,
- 19 projects were estimated with an RE from 25% to 190% of real hours.

The mean magnitude of relative error (MMRE) was defined as follows [9]:

$$\text{Mean Magnitude Of Relative Error} = \frac{1}{n} \times \sum_{i=1}^{n} \left| \frac{(\text{Real effort} - \text{Estimated effort})}{\text{Real effort}} \right|$$

The mean magnitude of the relative error (MMRE) of the Planning Poker estimation process in this organization was 58%, which was large by any standard of estimation accuracy.

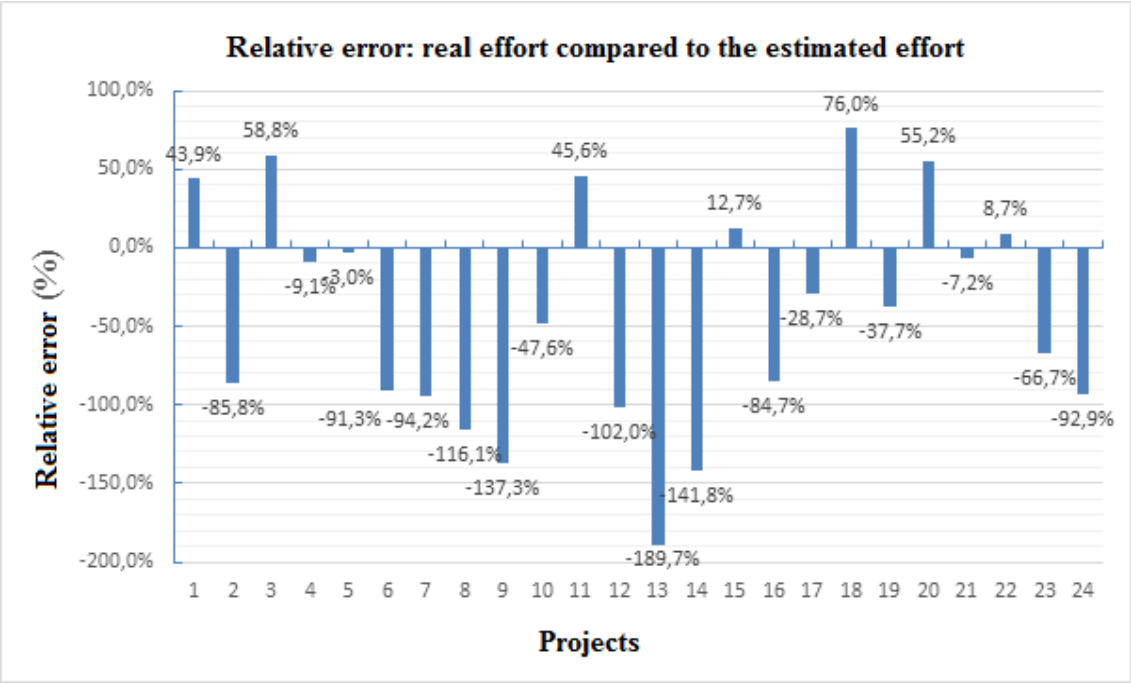


Figure 1: Story Points - Relative error of Estimated Effort (N = 24 tasks)

It can be observed from Figure 1 that of the 24 tasks, the team overestimated 17 (12 of these from 40% to 189%) and only underestimated seven. However, in absolute values, as shown in Table 1, some of the most underestimated tasks (Nos 1,3,11,18, and 20) were larger than a number of the overestimated tasks, indicating, here as well, a severe problem in underestimation.

3.4 Estimation model built with Story Points as the independent variable

The relationship between the estimates (here, the independent variable) derived from the Story Points and actual effort (the dependent variable) can be represented by linear regression models as in Figure 2, where the Story Points estimated hours are on the x axis, and the actual hours on the y axis. In Figure 2, the straight line represents the following equation modelling this relationship:

Actual Effort = 0.47 x Story Points Estimated Effort + 17.6 hrs

with an $R^2 = 0.33$

and MMRE = 58%

With a coefficient of determination R^2 of only 0.33, the relationship is very weak (the maximum being 1.0). This means that only 33% of the variation in increase along the y axis (the dependent variable) can be explained by an increase on the x axis (the independent variable). In other words, in this organization, the Story Points effort estimates do not provide reasonable estimates of the effort required to implement a task: It leads to a large MMRE of 58%, with both severe under-estimation and over-estimation.

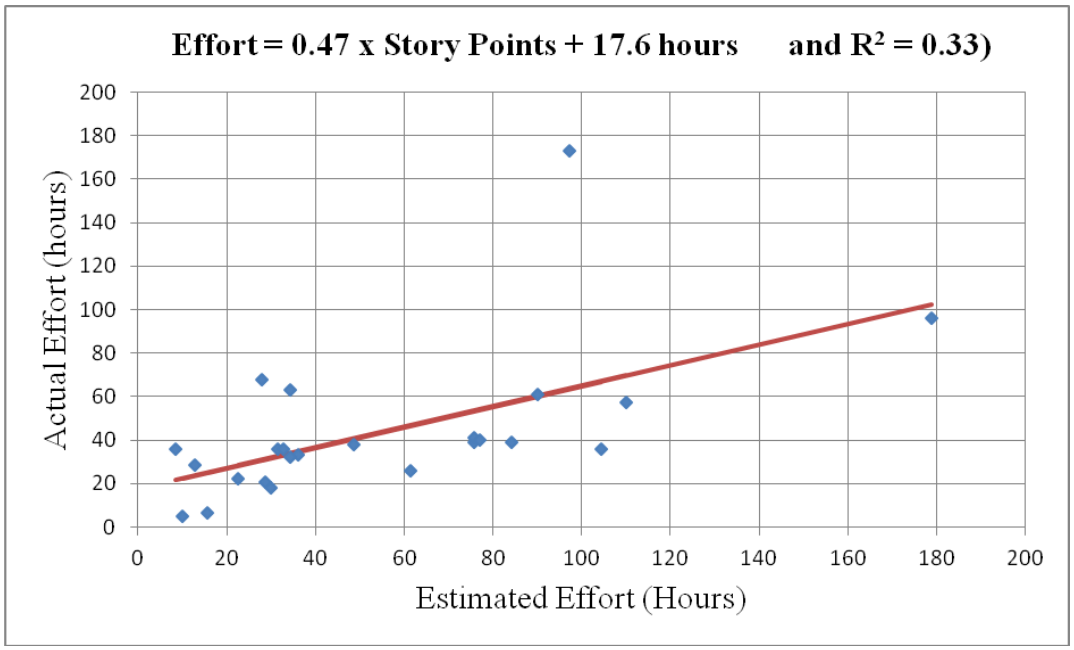


Figure 2: Story Points Estimated Effort versus Actual (N=24)

4. COSMIC data collection and analysis

4.1 COSMIC data collection

The functional size of the same set of 24 tasks was measured with the COSMIC method. The measurement was carried out either by examining the written documentation of the tasks, or by analyzing changes within the source code when the written documentation was insufficient. The measurement of the functional size of all tasks was performed by the same measurer. For each task, the components of the platform were identified, as well as the business users, the functional processes, and the objects of interest transiting across the software boundary.

Table 2 presents the detailed size by data movement types (i.e., Entry, Exit, Read, Write) and, in the rightmost column, the total functional size of these 24 tasks in COSMIC CFP measurement units. In summary:

- The functional size of the tasks varied from two CFP to 72 CFP
- The average functional size was 20 CFP

Table 2 COSMIC – ISO 19761 size of the 24 tasks

Task	Functional Process	Entry (CFP)	Exit (CFP)	Read (CFP)	Write (CFP)	Total Size (CFP)
1	15	27	9	28	8	72
2	9	19	10	7	1	37
3	13	13	10	5	3	31
4	4	4	4	3	0	11
5	4	4	2	3	1	10
6	8	8	7	3	3	21
7	6	6	5	7	1	19
8	6	12	4	11	3	30
9	4	4	4	2	2	12
10	7	9	5	7	3	24
11	12	12	12	35	0	59
12	1	1	0	0	1	2
13	4	4	3	5	1	13
14	1	1	1	1	0	3
15	4	4	1	5	3	13
16	3	3	2	9	1	15
17	5	5	2	4	3	14
18	3	3	3	9	0	15
19	2	2	2	1	0	5
20	7	7	3	4	4	18
21	5	5	4	6	1	16
22	6	9	3	3	3	18
23	3	3	3	3	0	9
24	6	7	5	13	1	26
Total	138	172	104	174	3	493

4.2 COSMIC-based estimation model

This section presents the construction of the estimation model based on COSMIC functional size with COSMIC CFP as the independent variable and actual effort as the dependent variable. For this dataset of 24 tasks, represented graphically in Figure 2, where the COSMIC size in CFP is presented on the x axis, and the actual effort on the y axis, the linear regression model was:

Effort = 1.84 × COSMIC Functional Size + 6.11 hrs
with an $R^2 = 0.782$
and an MMRE = 28%

With the coefficient of determination $R^2 = 0.782$ it can be seen that the COSMIC size now explains 78% of the variation in effort, in contrast to 33% when a regression-based Story Points model was used for estimation purposes. Furthermore, the mean magnitude of the relative error (MMRE) of the COSMIC-based estimation model was 28%, significantly better than an MMRE of 58% for the Story Points estimates.

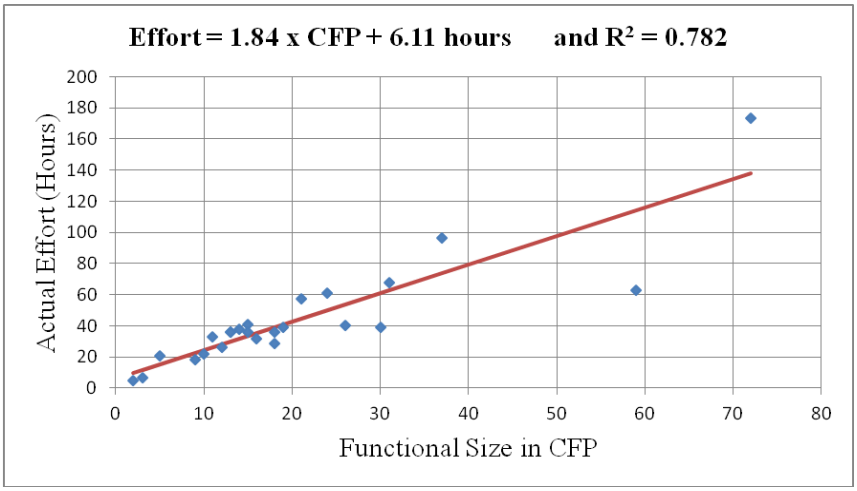


Figure 3: Estimation model with COSMIC CFP (N=24)

4.3 Project performance analysis with COSMIC

4.3.1 Unit effort comparison

In contrast to Story Points that does not size the output of a software task and cannot be used to calculate and compare project performance either in unit effort (i.e., in hours/CFP) or through a productivity ratio (i.e., in CFP/hour), the COSMIC-based measurement results allow one to calculate the performance of each task and to derive relevant management insights. For example, Figure 4 presents the performance in unit effort (i.e., hours/CFP on the y axis) for each of the 24 tasks (i.e., Tasks Id. on the x axis):

- the unit effort of 19 of the 24 tasks is within the 2 to 3 hrs/CFP range,

- four tasks (8, 11, 20 and 24) have a lower unit cost (i.e., higher productivity),

- only a single task (Id.19) has a much higher unit effort at over 4 hrs/CFP (i.e., poorer performance).

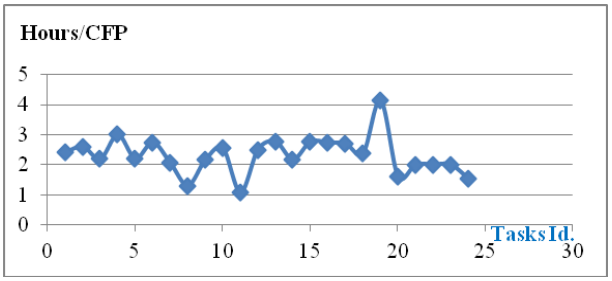


Figure 4: Unit effort in Hours/CFP of the 24 Tasks (ordered by Task Id.)

4.3.2 Analysis of productivity extremes and modified estimation model

To gain further insight into the estimation process Abran recommends [9, chapter 11] investigating the productivity extremes in order to determine what may cause such lower or higher effort. If the cause can be identified, then another independent variable can be added to the estimation model providing there are enough data points for statistical analysis. Alternatively the corresponding projects may be removed from the sample and used individually by analogy with another project with similar characteristics. If however, a reasonable cause cannot be identified, then such a data point should not be removed (i.e., being a productivity extreme alone is not a sufficient reason to remove a data point from a sample).

Within this data set the following productivity outliers can be identified from Figure 4:

- Task 19 has by far the highest unit effort at 4.15 hours/CFP, and

- Tasks 8 and 11 have by far the best performance (i.e., lowest unit effort at close to 1 hr/CFP).

A detailed analysis of the information available on task 19 could not identify any obvious cause for the poor performance of task 19, while it was observed that tasks 8 and 11 shared a common factor which positively impacted their productivity: Their requirements included significant functional reuse which led to very high code duplication which could explain such high productivity. In summary, it was deduced for this data set that the *functional reuse and related code duplication* was one reason contributing to the noise in the regression shown in Figure 2.

Therefore, if a future task includes functional reuse, as in tasks 8 and 11, then a unit effort of approximately 1 hr/CFP can be used in this organization to estimate such a specific task with this functional characteristic. However, for estimating tasks without functional reuse, a regression model should be built excluding, in this case, these two tasks, i.e. with the remaining 22 tasks. Figure 5 presents the new linear regression model with the 22 remaining tasks. It was observed that the coefficient of determination R^2 improved from .782 to .977. The new linear regression model equation becomes:

$$\text{Effort} = 2.35 \times \text{Functional size} - 0.08$$

$$\text{with } R^2 = 0.977$$

$$\text{MMRE} = 16.5\%$$

The new model explains more than 97% of the relationship between an increase in size and a corresponding increase in effort while the MMRE improved to 16.5%.

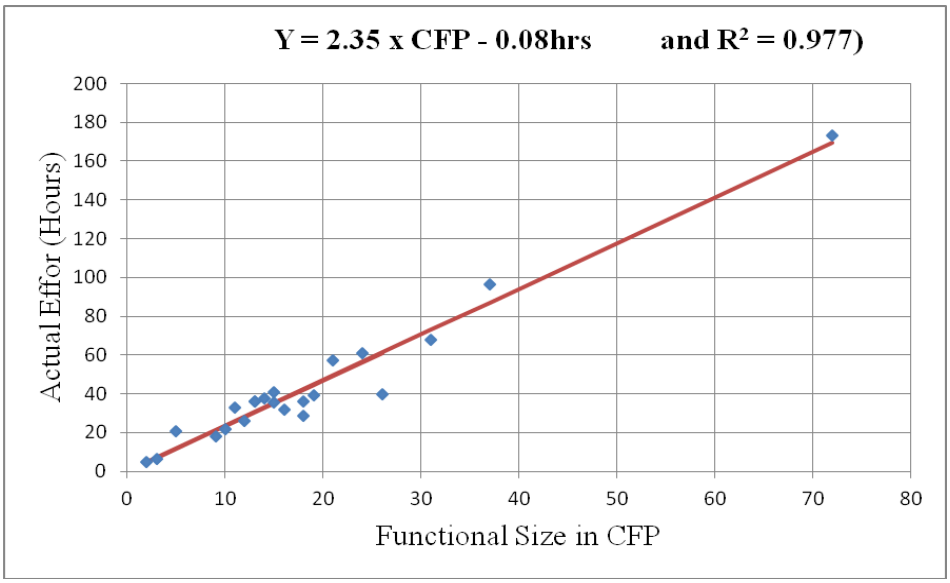


Figure 5: COSMIC-based Estimation Model (N = 22)

4.4 Using the COSMIC-based model for estimating an additional task

The COSMIC-based estimation model built from the set of 22 tasks was used with an additional project from this organization, which project was not included in the initial set of tasks. The new task corresponded to an improvement to the existing software application. The COSMIC measurement was made at the design phase of the iteration by the same measurer. The model was used as a method for a priori estimating, which means measuring a task before its implementation and providing an estimate of the development effort based on functional size measurement then using this size as the input to the COSMIC-based estimation model derived from the previous projects.

The size of the new task was 76 CFP close to task no. 1 with a size of 72 CFP. When 76 CFP was used as the independent variable in the estimation model of section 4.3.2, the model provided an estimated effort of 178 hours with an expected MMRE variation of ± 16.5%, i.e., an effort estimated to be between 149 and 207 hours. In practice, for estimation purpose for this specific software project, the MMRE lower limit was selected for the estimate for this specific additional task.

Once the task was completed, the time recording system indicated an (actual) development effort of 131 hours: this represents a unit effort of 1.72 hours per CFP. Considering the actual effort of 131 hours, the MMRE lower limit selected for estimation represented an overestimation of only 14%.

4.5 Updating the COSMIC-based model with the additional task

The organization can of course update its COSMIC-based estimation model every time data from a completed task becomes available. This is illustrated here with the addition of the project from the previous section as the 23rd data point. This updated estimation model is presented graphically in Figure 6, which now has 23 data points. The new linear regression model becomes:

$$\text{Task Effort} = 2.01 \times \text{COSMIC Functional Size} + 4.93 \text{ hours}$$

$$\text{with an } R^2 = 0.977$$

$$\text{and an MMRE} = 16.6\%$$

The coefficient of determination R^2 has been impacted downwards slightly from 0.977 to 0.962. Its MMRE has increased slightly from 16.5% to 16.6%.

Estimation of the next task can then be based on the updated model and therefore, iteration after iteration, the model can be adjusted to take account of additional information as it becomes available.

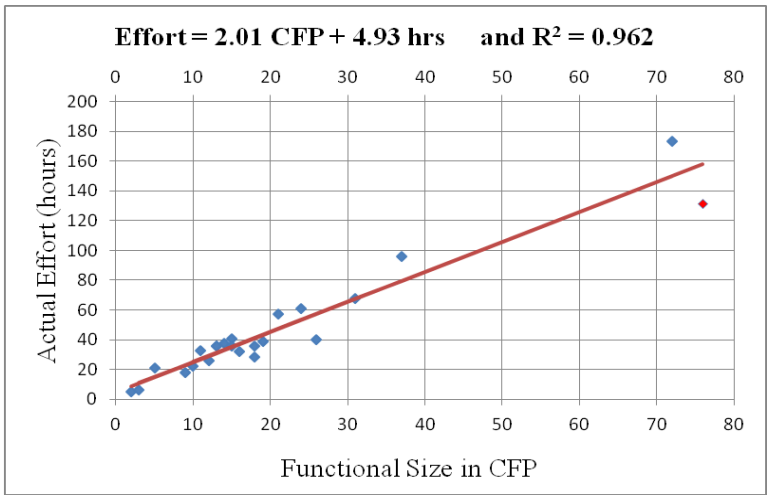


Figure 6: Estimation model updated with information from an additional task completed (N = 23)

5. Summary and Future Work

This study has examined the performance of the Planning Poker / Story Points as an estimation technique within an organization where software is being delivered using the Scrum methodology.

When compared to actual effort for 24 tasks completed at the industry site in the case study, estimates with Story Points in this organization were shown to have led to large under- and over-estimates, with an MMRE of 58% - see Figure 2. It was also observed that since Story Points do not size the output of a software task, it could not be used to calculate and compare tasks performance.

When the functional size of the corresponding completed projects was measured using the COSMIC Function Points method, productivity could be calculated and compared using either unit effort (hours/CFP) or a productivity ratio (CFP/hour). This study has allowed us to emonstrate that in this organization the development team implemented the tasks at a sustained pace within a range of 2 to 3 hrs/CFP for 20 or the 24 tasks, iteration after iteration, throughout the period for which the data was collected – Figure 4.

An estimation model built with better estimation performance could then be compared to the Planning Poker estimates. The COSMIC based estimation model built with the initial 24 tasks in this organization had a much smaller estimation variance (i.e., an MMRE of 28%) – Figure 3.

The analysis of productivity extremes within this data set allowed identifying a functional reuse context within this organization that had led to major productivity gains for 2 tasks with such high functional reuse. For the purpose of estimating new tasks without high functional reuse, a new estimation model was built excluding these two tasks with high reuse: the reduced estimation model based on the subset of 22 tasks led to an improved estimation model with a much smaller MMRE of 16.5% - Figure 5. This paper illustrated as well how this industry site used this estimation model equation and expected MMRE variance to estimate an additional task, and compared its actual effort with the estimation range derived from the model. It also illustrated how to improve the estimation model by adding into it this additional completed task – Figure 6.

Finally, it can be noticed that measuring tasks with the COSMIC ISO standard and using their functional size in CFP unit to build an estimation model is an objective process without subjective judgments. Therefore, using a COSMIC-based estimation model means that, from one task to another, the effort for a software task of a given COSMIC size can be estimated within an expected smaller relative error range, regardless of which team members participating in the estimate, their experience and their knowledge of the software in question.

In summary, although the Planning Poker / Story Points are widely recognized and used in the Agile community, the COSMIC measurement method provides objective evidence of the team performance as well as better estimates.

References

1. Moniruzzaman, ABM; Akhter Hossain, S., "Comparative Study on Agile software development methodologies", Global Journal of Computer Science and Technology, Volume 13, Issue 7, 2013.
2. Schwaber, Ken, "Agile Project Management with Scrum", Redmond, WA: Microsoft Press, 2004.
3. Schwaber, K.; Beedle, M., "Agile Software Development with Scrum", Prentice Hall, Upper Saddle River, N.J., 2002.
4. ISO, "ISO 1971: 2011 - Software engineering - COSMIC: A Functional Size Measurement Method", International Organization for Standardization, Geneva, Switzerland, 2011.
5. COSMIC Group, "The COSMIC Functional Size Measurement Method – Version 4.0.1: Measurement Manual – The COSMIC Implementation Guide for ISO/IEC 19761: 2011", http://cosmic-sizing.org/publications/measurement-manual-401 [Accessed February 2, 2016].
6. Grenning, J., "*Planning Poker*", Renaissance Software Consulting, 2002.
7. Cohn, M., "Agile Estimating and Planning", Prentice Hall, 2005.
8. Cohn M., "User Stories Applied for Agile Software Development, Addison-Wesley, 2004.
9. Abran, Alain, "Software Project Estimation – The Fundamantal for Providing High Quality Information to Decision Makers", John Wiley & Sons, Hoberken, New Jersey, 2015, pp. 261.
10. COSMIC Group, "The COSMIC Functional Size Measurement Method - Version 3.0.1 - Guideline for Sizing Agile Projects", 2011. http://cosmic-sizing.org/publications/guideline-for-the-use-of-cosmic-fsm-to-manage-agile-projects/ accessed on February 21, 2016.
11. Commeyne, Christophe, "Établissement d'un modèle d'estimation *a posteriori* de projets de maintenance de logiciels", Master's thesis in Software Engineering, École de Technologie Supérieure (ÉTS) – University of Québec, Montreal, Canada, 2014.

Seufert, M.; Ebert, C, Fehlmann, T.; Pechlivanidis, S.; Dumke, R. R.:

MetriKon 2015 - Praxis der Softwaremessung
Tagungsband des DASMA Software Metrik Kongresses
5. - 6. November 2015, IBM, Köln

Shaker Verlag, Aachen, 2015 (272 Seiten)

The book includes the proceedings of the MetriKon 2015 held in Cologne in November 2015, 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.

Schmietendorf, A.; Simon, F.:

BSOA/BCloud 2015
10. Workshop Bewertungsaspekte serviceorientierter Architekturen
3. November 2015, Leipzig

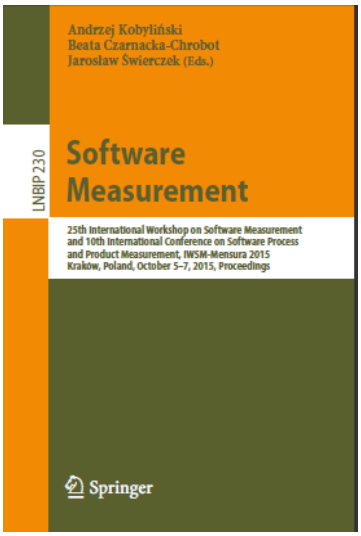
Shaker Verlag, Aachen, 2015 (112 Seiten), ISBN 978-3-8440-2108-0

The book includes the proceedings of the BSOA/BCloud 2015 held in Leipzig in November 2015, which constitute a collection of theoretical studies in the field of measurement and evaluation of service oriented and cloud architectures.

Andrzej Kobyliński · Beata Czarnacka-Chrobot
Jarosław Świerczek (Eds.)

Software Measurement

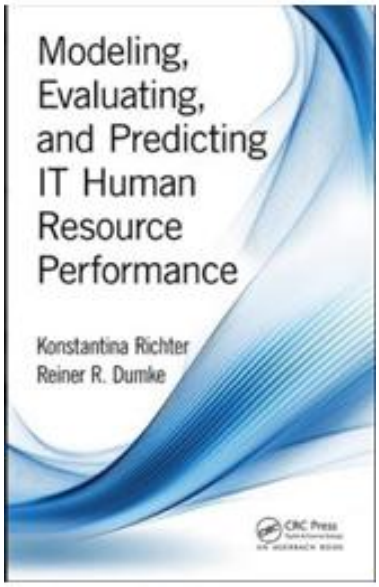
25th International Workshop on Software Measurement and 10th International Conference on Software Process and Product Measurement, IWSM-Mensura 2015
Kraków, Poland, October 5–7, 2015
Proceedings



Konstantina Richter, Reiner Dumke:

Modeling, Evaluating and Predicting IT Human Resource Performance

CRC Press, Boca Raton, Florida, 2015 (275 pages)



Qualitative und quantitative Bewertungsaspekte bei der agilen Softwareentwicklung plattformübergreifender mobiler Applikationen

André Nitze, Andreas Schmietendorf

2014, 168 Seiten
ISBN 978-3-8325-3774-6
Preis: 24.00 €

Bestellung über den Buchhandel oder direkt beim Verlag, entweder online oder per Fax beim Logos Verlag Berlin GmbH· Comeniushof – Gubener Str. 47 · D-10243 Berlin

Schmietendorf, A. (Hrsg.):

Eine praxisorientierte Bewertung von Architekturen und Techniken für Big Data

(110 Seiten) Shaker-Verlag Aachen, März 2015 ISBN 978-3-8440-2939-0

Christof Ebert:

Risikomanagement kompakt

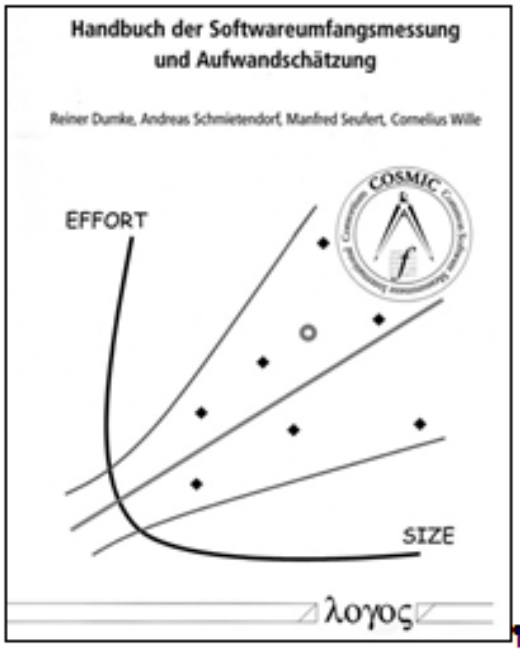
- Risiken und Unsicherheiten bewerten und beherrschen

Springer-Verlag, 2014, ISBN 978-3-642-41047-5

Dumke, R., Schmietendorf, A., Seufert, M., Wille, C.:

Handbuch der Softwareumfangsmessung und Aufwandschätzung

Logos Verlag, Berlin, 2014 (570 Seiten), ISBN 978-3-8325-3784-5



Software Measurement & Data Analysis Addressed Conferences

January 2016:

SWQD 2016: **Software Quality Days**
January 18-21, 2016, Vienna, Austria
see: https://2016.software-quality-days.com/

February 2016:

ICSEFM 2016: **18th International Conference on Software Engineering and Formal Methods**
February 4 - 5, 2016, Melbourne, Australia
see: https://www.waset.org/conference/2016/02/melbourne/ICSEFM

SOFTENG 2016: **International Conference on Advances and Trends in Software Engineering**
February 21 - 25, 2016, Barcelona, Spain
see: http://www.iaria.org/conferences2016/SOFTENG16.html

ISEC 2016: **9th India Software Engineering Conference**
February 18 - 20, 2016, Goa, India
see: http://www./isec2016.org/

March 2016:

ICPE 2016: **7th ACM/SPEC International Conference on Performance Engineering**
March 12-18, 2016, Delft, Netherlands, USA
see: http://icpe2016.spec.org/

REFSQ 2016: **22th International Working Conference on Requirements Engineering: Foundation for Software Quality**
March 14-17, 2016, Göteborg, Sweden
see: http://refsq.org/2016/welcome/

BigDataService 2016: **IEEE BigDataService 2016**
March 29 - April 1, 2016, Oxford, UK
See: http://www.big-dataservice.net/

April 2016:

FASE 2016:	**18th International Conference on Fundamental Approaches to Software Engineering** April 2-8, 2016, Eindhoven, Netherlands see: http://www.etaps.org/index.php/2016/fase
eMetrics 2016:	**eMetrics Summit** April 3 - 6, 2016, San Francisco, USA See: http://www.emetrics.org/sanfrancisco/2016/
QoSA 2016:	**12th International ACM Sigsoft Conference on the Quality of Software Architectures** April 5 - 8, 2016, Venice, Italy see: http://qosa.ipd.kit.edu/qosa_2016/
WICSA and CompArch 2016:	**13th Working IEEE/IFIP Conference on Software Architecture** April 5 - 8, 2016, Venice, Italy see: http://www.softwarearchitecture.org/cfp_smart.html
ASWEC 2017:	**23nd Australian Software Engineering Conferences** April, 2017, Sydney, Australia see: http://www.aswec2017.org/ (not in 2016)
ICST 2016:	**9th International Conference on Software Testing, Verification & Validation** April , 2016, Chicago, USA See http://www.pnsqc.org/icst-2016-ieee-international-conference-on-software-testing-verification-and-validation/
CIbSE 2016:	**19th Iberoamerican Conference on Software Engineering** April 27-29, 2016, Quito, Ecuador see: http://cibse.espe.edu.ec/
CSEE&T 2016:	**26th Conference on Software Engineering Education and Training** April 5 - 6, 2016, Dallas, Texas see: http://paris.utdallas.edu/cseet16/
ICAMDS2016:	**International Conference on Applied Mathematics and Data Science** April 26-27, Hangzhou, China See: http://www.icamds.com/2016/home
iqnite 2016:	**Software Quality Conference** April 26 28, 2016, Düsseldorf, Germany see: https://www.iqnite-conferences.com/de/
ENASE 2016:	**11th International Conference on Evaluation of Novel Approaches to Software Engineering** April 27 - 28, 2016, Rome, Italy see: http://www.enase.org/

May 2016:

ISMA 2016: **12th ISMA Conference of the IFPUG**
May 3-5, 2016, Rome, Italy
see: http://www.ifpug.org/event/isma-12-in-rome-italy/?lang=de

STAREAST 2016: **Software Testing Analysis & Review Conference**
May 1-6, 2016, Orlando, FL, USA
see: http://stareast.techwell.com/

EMEA 2016: **PMI Global Congress 2016 - EMEA**
May 9-11, 2016, Barcelona, Spain
see: http://congresses.pmi.org/emea2016

ASQ 2016: **International Conference on Software Quality (ASQ)**
May 16 - 18, 2016, Milwaukee, USA
see: http://www.pnsqc.org/international-conference-on-software-quality-asq-2016/

SAM 2015: **Workshop on Software Architecture and Metrics**
May, 2015, Florence, Italy
see: http://www.sei.cmu.edu/community/sam2015/ (not 2016)

OSS 2016: **11th International Conference on Open Source Systems**
May 30 - June 2, 2016, Gothenburg, Sweden
see: http://oss2016.cs.tut.fi/

ICSE 2016: **38th International Conference on Software Engineering**
May 14- 22, 2016, Austin, Texas, USA
See: http://2016.icse.cs.txstate.edu/

MSR 2016: **13th Working Conference on Mining Software Repositories**
May 14 - 15, 2014, Austin, Texas, USA
see: http://2016.msrconf.org/#/home

ICPC 2016: **24th International Conference on Program Comprehension**
May 16 - 17, 2016, Austin, Texas, USA
see: http://www.program-comprehension.org/icpc16/

ODSC 2016: **Open Data Science Conference**
May 20 - 22, 2016, Boston, USA
See: http://odsc.com/boston/

IMMM 2016: **Sixth International Conference on Advances in Information Mining and Management**
May 22 - 26, 2016, Valencia, Spain
See: http://www.iaria.org/conferences2016/IMMM16.html

XP 2016: **17th International Conference on Agile Software Development**
May 24-27, 2016, Edinburgh, Scotland
See http://conf.xp2016.org/events/xp-2016-edinburgh/

June 2016:

EASE 2016: **20th International Conference on Empirical Assessment in Software Engineering**
June 1 - 3, 2016, Limerick, Ireland
see http://ease2016.lero.ie/

SERA 2016: **14th ACIS Conference on Software Engineering Research, Management and Applications**
June 8 - 10, 2016, Towson, Maryland, USA
see: http://www.acisinternational.org/sera2016/

EJC 2016: **26th European Japanese Conference on Information Modeling and Knowledge Bases**
June 6 - 10, 2016, Tampere, Finland
see: http://www.tut.fi/en/ejc/ejc-2016/

ICWE 2016: **International Conference on Web Engineering**
June 6 - 9, 2016, Lugano, Switzerland
see: http://icwe2016.inf.usi.ch/

SPICE 2016: **16th International SPICE Conference**
June 9 - 10, 2016, Dublin, Ireland
see: http://www.spiceconference.com/

AGILE 2016: **19th AGILE Conference on Geographic Information Science**
June 14 - 17, 2016, Helsinki, Finland
see: http://www.agile-online.org/index.php/conference/conference-2016

July 2016:

UKPEW 2014: **24th Annual United Kingdom Workshop on Performance Engineering**
July 4 - 5, 2014, Edinburgh, UK
see: http://ukpew.lboro.ac.uk/ ***(not in 2016)***

SQ 2016: **Seventh International Symposium on Software Quality**
July 4 - 7, 2016, Beijing, China
see: http://sq.covenantuniversity.edu.ng/

VDA Automotive SYS Conference 2016: **Quality Management for Automotive Software-based Systems and Functionality**
July 6 - 8, 2016, Berlin, Germany
see: http://vda-qmc.de/en/software-processes/vda-automotive-sys/

ICSOFT 2016: **11th International Conference on Software and Data Technologies**
July 24 - 26, 2016, Lisbon, Portugal
see: http://www.icsoft.org/

SERP 2016: **14th International Conference on Software Engineering Research and Practice**
July 25 - 28, 2016, Las Vegas, Nevada, USA
see: http://worldcomp.org/events/2016

DMIN'16: **12th International Conference on Data Mining**
July 25 - 28, Las Vegas, USA
See: http://www.dmin-2016.com/

August 2016:

ICGSE 2016: **11th International Conference on Global Software Engineering**
August 2 - 5, 2016, Orange Country, California, USA
see: http://www.ics.uci.edu/~icgse2016/2_0cfp.html

ICSEA 2016: **10th International Conference on Software Engineering Advances**
August 21 - 25, 2016, Brussels, Belgium
see: http://www.iaria.org/conferences2016/ICSEA16.html

QEST 2016: **13th International Conference on Quantitative Evaluation of Systems**
August 23 - 25, 2016, Quebec City, Canada
see: http://www.qest.org/

ICDSE 2016: **International Conference on Data Science and Engineering**
August 23 - 25, Kerala, India
See: http://icdse.cusat.ac.in/

Euromicro DSD/ SEAA 2016: **Software Engineering & Advanced Application Conference**
August 31 - September 2, 2016, Limassol, Cypros
see: http://dsd-seaa2016.cs.ucy.ac.cy/

September 2016:

ESEM 2016: **10th International Symposium on Empirical Software Engineering & Measurement**
September 8 - 9, 2016, Ciudad Real, Spain
see: http://alarcos.esi.uclm.es/eseiw2016/esem/

RE 2016: **24th IEEE International Requirement Engineering Conference**
September 12 - 16, 2016, Beijing, China
see: http://re16.org/

EuroAsiaSPI^2 2016: **23th European Systems & Software Process Improvement and Innovation Conference,**
September 14 - 16, 2016, Graz, Austria
see: http://www.eurospi.net/

ASQT 2016: **Arbeitskonferenz Softwarequalität, Test und Innovation**
September 21 - 23, 2016, Klagenfurt, Austria
see: http://www.asqt.org/

Big Data 2016: **Big Data Analysis and Data Mining**
September 26 - 27, 2016, London, UK
See: http://datamining.conferenceseries.com/

October 2016:

IWSM-MENSURA 2016: **Common International Conference on Software Measurement**
October 5 - 7, 2016, Berlin, Germany
see: http://www.iwsm-mensura.org/

ISSRE 2016: **27th International IEEE Symposium on Software Reliability Engineering**
October 23 - 27, 2016, Ottawa, Canada
see: http://issre.net/

November 2016:

BSOA/BCloud 2016: **11. Workshop Bewertungsaspekte service-orientierte und Cloud-Architekturen**
November , 2016, Berlin, Germany
see: http://www-ivs.cs.uni-magdeburg.de/~gi-bsoa/

ICDM 2016: **IEEE International Conference on Data Mining**
November 28 - 30, 2016, Barcelona, Spain
See: http://icdm2016.eurecat.org/

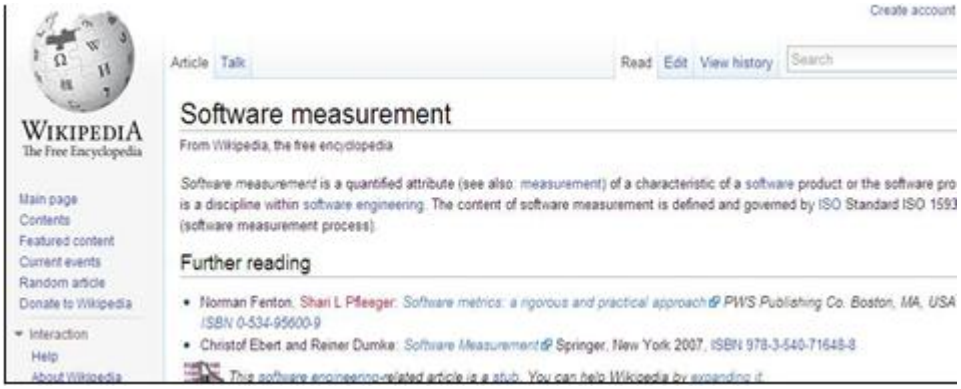
December 2016:

PROFES 2015: **16th International Conference on Product Focused Software Process Improvement**
December 2 - 4, 2015, Bolzano, Italy
see: http://profes2015.inf.unibz.it/ ***(not in 2016)***

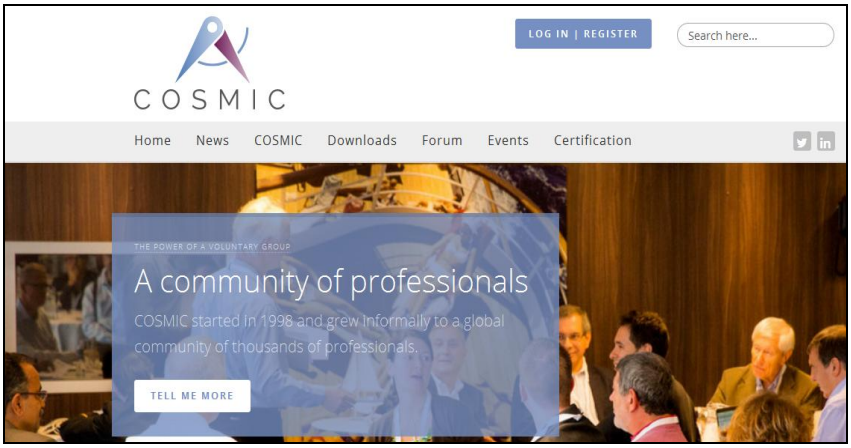
see also: Conferences Link of **Luigi Buglione (**http://www.semq.eu/leng/eveprospi.htm**)**

See the GI-Web site **http://fg-metriken.gi.de/** for the digital contents of the Software Measurement News:

Help to qualify the software measurement knowledge and intentions in the world wide web:



cosmic-sizing.org:



See our overview about software metrics and measurement in the Bibliografie at ***http://fg-metriken.gi.de/bibliografie.html*** including any hundreds of books and papers:

Gesellschaft für Informatik

Fachgruppe Software-Messung und -Bewertung

Startseite | Vorstand | Aktuelles | Bibliografie | Arbeitskreise | Software Measurement Ne

Sie befinden sich hier: Startseite/Bibliografie

Software Measurement Bibliography

Basisliteratur finden Sie hier

1 Software Measurement Foundations

- Measurement Overview
- Measurement Principles & Foundations
- Measurement Standards
- Basic (Set of) Measures
- Measurement Validation
- Measurement & Statistics

2 Software Process & Product Measurement

2.1 Software Product Measurement

- Requirements Measurement
- Review/Inspection/Audits Measurements
- Specification Measurement
- Architecture Measurement
- Product Line Measurement
- Reliability Measurement
- Security Measurement
- Performance Measurement

2.2 Software Process & Resources Measurement

- Process Management & Measurements
- Risk Management & Measurements
- Process Improvement Measurements
- Quality Assurance Measurements
- Software Test Measurements
- Maintenance Measurements
- Personal Measurements
- SPC (Statistical Process Control)

2.3 Software Process Maturity Models and Measurement

- General Maturity Quantifications
- CMMI/SPICE/ITIL Measures
- Maintenance Maturity Measures (S3M)
- Agile Process Measures (AMMI)

2.4 Software Size Measurement & Cost/Effort Estimation

- Size Measurement
- Cost/Effort Estimation
- COCOMO (Constructive Cost Model)
- (Empirical-based) Function Points
- Use Case Points
- Object/Data/Feature Points

2.5 Software Metrics Tools and Infrastructures

- Program (Source Code) Measurement Tools
- Metric Analysis Tools
- Process Management & Measurement Tools
- SQA (Software Quality Assurance) Tools
- Metrics Data Bases & Experience Factories
- Software Benchmarks & ISBSG (International Software Benchmark Standard Group)
- Cockpits/Dashboards & Measurement Infrastructures
- Measurement Services, eMeasurement & WEB Tomography

3 Measurement of Software Paradigms and Technologies

3.1 Procedural-Based Software Engineering (PBSE)

- Source Code Measurements (Halstead, McCabe etc.)
- Control/Data Flow Measures
- Module/Procedure Measures

3.2 Object-Oriented Software Engineering (OOSE)

- Object Oriented Design Measures
- Object Oriented Programming Measures
- Object Oriented Test Measures
- Aspect-Oriented Measures

3.3 Applicative & Declarative Approaches (Calculus)

- Functional Programming Measurement
- Logical Programming Measurement
- Formal Specification Measurement

3.4 Component-Based Software Engineering (CBSE)

- Components (COTS) Measurements
- EAI Measurements
- Reusability Measurement
- Feature-Oriented Measurement

3.5 Service-Oriented Software Engineering (SOSE)

- Web (Service) Measurement
- SOA (Service-Oriented Architecture) Measurement
- Cloud Measurement
- Big Data Measurement

3.6 Agent-Oriented Software Engineering (AOSE)

- Software Agents Measurement
- MAS (Multi Agent System) Measurement
- MAS Development Measurement
- Self Management & Measurement

4 Measurement Frameworks

4.1 General Measurement Approaches

- General Frameworks
- GQM (Goal Question Metric)
- PDCA (Plan, Do Check, Act)
- E4 (Establish, Extract, Evaluate, Execute)
- CAME (Choice, Adjust, Migration, Efficiency)
- DMIAC (Define, Measure, Analyze, Improve, Control)

4.2 Modern Measurement Frameworks

- Adaptive Measurement Frameworks
- Proactive Measurement Frameworks
- Ontology-Based Measurement Frameworks
- Categorical Theory Based Approaches
- Ubiquitous Measurement Frameworks

4.3 Measurement Process Evaluations

- Measurement Evidence & Evaluation
- Measurement Process Classification

See our further software measurement and related communities:

www.dasma.org:

www.isbsg.org:

www.cecmg.de:

www.mai-net.org:

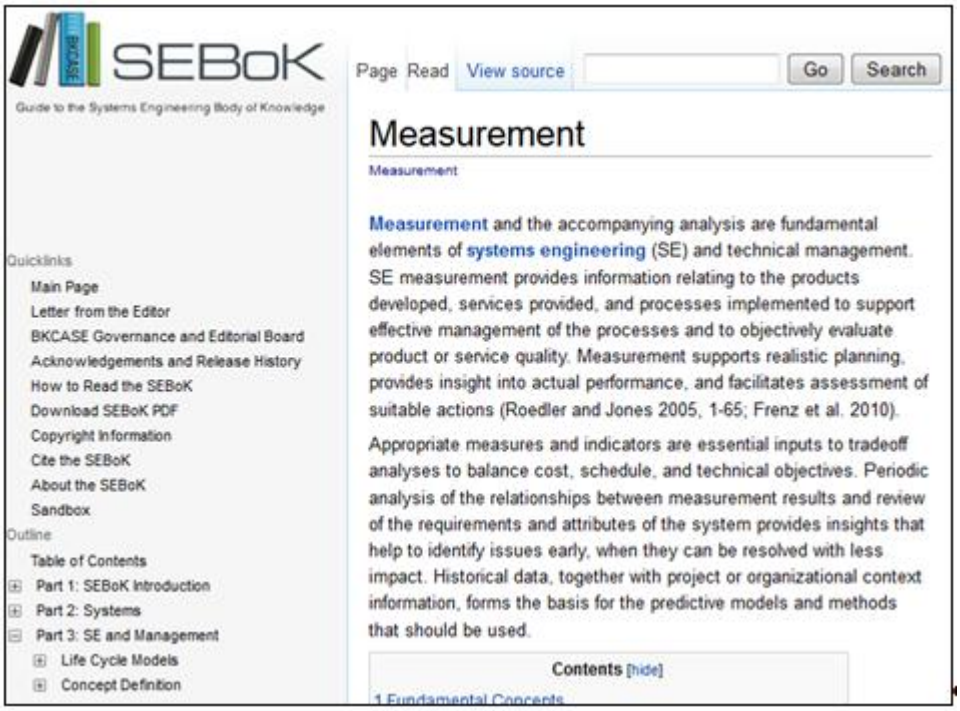
www.swebok.org:

isern.iese.de:

www.smlab.de:

www.psmsc.com/:

sebokwiki.org/wiki/Measurement:

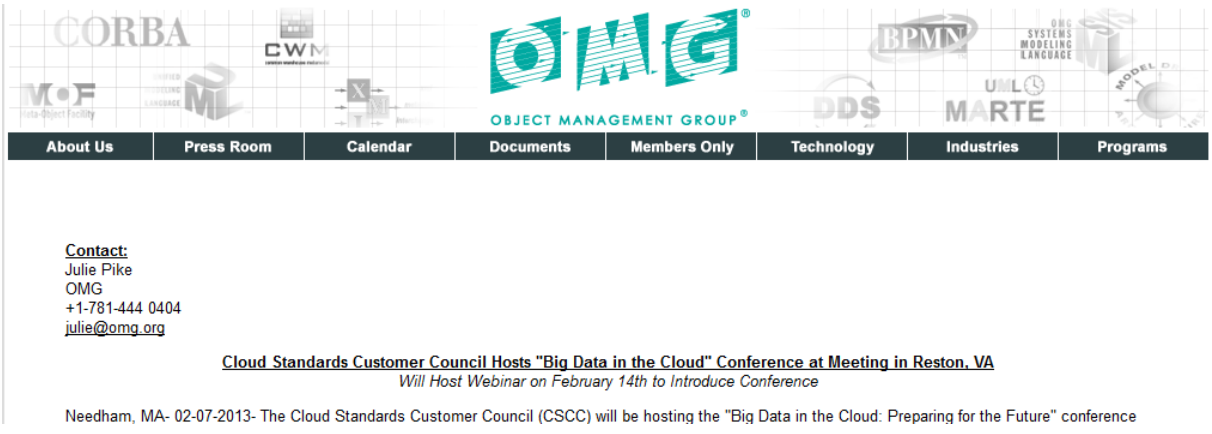


www.fisma.fi/in-english/:

http://nesma.org/:

www.sei.cmu.edu/measurement/:

http://www.omg.org/news/releases/pr2013/02-07-13.htm:



SOFTWARE MEASUREMENT NEWS

VOLUME 21 2016 NUMBER 1

CONTENTS

Announcements **3**

Conference Report **9**

COSMIC Information **19**

Position Paper

Christophe Commeyne, Alain Abran, Rachida Djouab:
Effort Estimation with Story Points and COSMIC Function Points - An Industry Case Study **25**

New Books on Software Measurement **37**

Conferences Addressing Measurement Issues **41**

Metrics in the World-Wide Web **47**

ISSN 1867-9196