

METRICS NEWS

VOLUME 7

2002

NUMBER 2

CONTENTS

Announcement 3

Workshop Report 5

Experience Report 21

New Books on Software Metrics 31

Conferences Addressing Metrics Issues 35

Metrics in the World-Wide Web 37

ISSN 1431-8008

Our next Conferences

- The 13th International Workshop on Software Measurement (IWSM2003) will be held in Montreal (Canada) in September 2003

(see: <http://www.lrgl.uqam.ca/>
for more details)

- DASMA Conference Metrikon 2003, 10th Anniversary of DASMA, will be held on November 10th / 11th, 2003, maybe in Frankfurt, Cologne or Stuttgart

(the GI group FG 2.1.10 will participate at this software measurement highlight in Germany)

(see: <http://www.dasma.org/>
for more details)

Our 12th Workshop on Software Measurement took place in Magdeburg, Germany in October 2002. The following report gives an overview about the presented papers. Furthermore, the papers are published in the following Shaker book:

SOFTWARE MEASUREMENT AND ESTIMATION

Proceedings of the IWSM 2002

Reiner Dumke, Alain Abran, Manfred Bundschuh and
Charles Symons (Eds.)

**12th International Workshop, IWSM 2002
Magdeburg, Germany
October 7 – 9, 2002**

Shaker Publisher 2002

ISBN 3-8322-0765-1

Problems and Pitfalls in Software Metrics Applications

Horst Zuse

TU-Berlin, Germany

horst.zuse@t-online.de

<http://www.zuse.info>

Preface. For more than fifty years computers have played a more and more important role in our life. It was estimated that, by 2000, more than one half of American work force will rely on computers and software to do its daily work. As computer hardware costs continue to decline, the demand for new applications software continues to increase at a rapid rate.

Software measurement has activated an enormous amount of research since the middle of the seventies. Today, on any software engineering conference, the topic software measures is mentioned in some way. However, today, software measurement is not yet a mature science.

INITIAL MODELING OF THE MEASUREMENT CONCEPTS IN THE ISO VOCABULARY OF TERMS IN METROLOGY

ALAIN ABRAN, ASMA SELLAMI

[École de Technologie Supérieure – ETS, Montréal, Canada](#)

aabran@ele.etsmtl.ca

ABSTRACT. THE FIELD OF SOFTWARE METRICS IS USUALLY DISCUSSED FROM THE PERSPECTIVE REFERRED TO AS 'MEASUREMENT THEORY'. HOWEVER, IN OTHER DISCIPLINES, THE DOMAIN OF KNOWLEDGE REFERRED TO AS 'METROLOGY' IS THE FOUNDATION FOR THE DEVELOPMENT AND USE OF MEASUREMENT INSTRUMENTS AND MEASUREMENT PROCESSES. THIS PAPER PRESENTS AN INITIAL MODELLING OF THE SETS OF MEASUREMENT CONCEPTS DOCUMENTED IN THE ISO INTERNATIONAL VOCABULARY OF BASIC AND GENERAL TERMS IN METROLOGY. IN PARTICULAR, THIS MODELLING ILLUSTRATES THE VARIOUS LEVELS OF ABSTRACTION OF THE CONCEPTS AS WELL AS THE RELATIONSHIPS ACROSS RELATED CONCEPTS AND SUB-CONCEPTS. WE REFER TO THIS REPRESENTATION TYPE AS THE TOPOLOGY OF THE CONCEPTS WITHIN THE ISO VOCABULARY. THESE MODELS WILL PROVIDE THE BASIS FOR ANALYSING THE CURRENT STATUS OF THE FIELD OF 'SOFTWARE METRICS' AND TO SUGGEST IMPROVEMENTS ALONG THE CLASSICAL PATH OF THE FIELD OF METROLOGY.

XMI-Relational Model of a Software Metric Database

Harry M. Sneed

Institut für Wirtschaftsinformatik, Universität Regensburg, Germany

Software Data Service, Wien, Austria

Abstract. Based on the metrics specified in the ANSI/IEEE Standards 982 and 1061 for Software Measures and the ISO-9126 Standard for Product Quality Evaluation as well as on the results of the ESPRIT-METKIT Project, an object-oriented data model is proposed for storing and retrieving software metrics. The SoftMess Model unites process, product and resource metrics collected from various sources. These metrics are stored in a relational database by project, product component and

by organisational unit. They are then used for querying and reporting purposes as well as a data basis for other software management tools. The import and export interfaces to the metric database are XML files created by means of static analysis, dynamic analysis, defect analysis, and from the time reports. These are converted to excel tables for presentation and further processing. The model itself is specified in the XMI language which combines UML with XML. The metric data stored is intended to support cost estimation, project planning, reliability prediction, product assessment and overall quality management. A metric database based on this model has been implemented at the Software Data Service in Vienna and is currently being used there to calculate and control maintenance activities performed on the GEOS stock brokerage system.

ISO TRANSPOSITION AND CLARIFICATIONS OF THE COSMIC FFP METHOD OF FUNCTIONAL SIZING

Alain Abran, Peter Fagg, Roberto Meli, Charles Symons
<http://www.lrgl.uqam.ca/cosmic-ffp/>

Abstract. This paper describes and explains a few significant changes which have been made to the COSMIC FFP method of functional sizing of software targeted to be published in the Measurement Manual Version 2.2 (October 2002) and in the draft ISO/IEC 19761 standard version of the method. These changes have been made to help understanding and consistent use of the method.

None of the changes alter the principles of the method. These have never needed to be changed since the method was first defined, and have been confirmed by successful practical use in many organisations on different types of software. All the changes have arisen because it was found either that certain terms and definitions could be misunderstood and needed clarification, or because of the need to ensure consistency with existing ISO/IEC standard terminology and definitions (a design goal of the COSMIC FFP method).

AUTOMATION OF COUNTING OF FUNCTIONAL SIZE USING COSMIC FFP IN UML

Malcolm S. Jenner
School of Computing and Information Technology
University of Wolverhampton, UK
M.S.Jenner@wlv.ac.uk

Abstract. In this paper I build on previous work showing a method for estimating the functional size in COSMIC Functional Size Units of a software system specified using UML. I show how this can be automated when the model is developed using a suitable CASE tool.

Design of a Diagnostic Tool to Improve the Quality of the Functional Measurement

Jean-Marc Desharnais¹, Tim Küssing², Alain Abran³, André Mayers⁴

¹Software Engineering Laboratory in Applied Metrics, Canada
Desharnais.jean-marc@uqam.ca

²Department of Data and Information Technology
University of Applied Science, Nuremberg, Germany
tim.kuessing@gmx.de

³Département de génie électrique
École de Technologie Supérieure – ETS Montreal, Canada
aabran@ele.etsmtl.ca

⁴Département de Mathématique et d'Informatique
Université de Sherbrooke, Canada
andre.mayers@DMI.USherb.ca

Abstract. This document presents the design of a diagnostic tool to assist measurers in applying consistently and systematically a functional measurement method. The design of the diagnostic tool is based on the UML (Unified Mark-up Language) method and a specific application of van Heijst knowledge modeling method. The result is a hybrid diagnostic tool using CBR and rule based techniques.

COMPARING ERP REQUIREMENTS ENGINEERING PROCESSES: A CASE STUDY

Maya Daneva
TELUS Mobility, Canada
maya.daneva@telus.com

Abstract. Developing business requirements for complex Enterprise Resource Planning (ERP) projects within time and budget relies more than ever on the application of standard packaged off-the-shelf requirements engineering (RE) processes.

This paper extends results from our previous experiences in maturity assessment of standard RE processes; it addresses the following questions associated with the adoption of standard RE processes in immature organizations: what happens when making a standard process model a live process, what are the essential practices that contribute to the process success, and what are the costs of the process and the quality of the delivered results. The purpose of this paper is to provide some practical advice for using assessments to increase project teams' understanding of the RE process itself and the factors that impact its successful adoption.

Quality Evaluation of Large-Scale Software Systems

Mathias Lothar, Andreas Schmietendorf, Thomas Böhm, Reiner R. Dumke

University of Magdeburg, Germany,

{lothar, schmiete, boehm, dumke}@ivs.cs.uni-magdeburg.de

<http://ivs.cs.uni-magdeburg.de/sw-eng/us/>

Abstract. In a time of increasing penetration by software in nearly all areas of our lives the software's quality is a very important criteria in order to trust in its reliability and functionality. But also for several business decisions, e.g. the decision to reuse software or to develop it from scratch, or in order to get a general system overview, information about the software quality are desired. Especially for large systems, that already exist for a long period of time, it is difficult to get information of its quality, since they grow and grow to complex structures over time and often also the documentation is not up-to-date anymore.

Because of the size and complexity it is usually impossible to evaluate the software manually; for this reason methods and tools are needed to support this task. This paper will introduce a methodology supporting the tool-based quality evaluation of software systems as well as demonstrate the application of the methodology for a telecommunication software system.

With help of the Logiscope tool the explained theoretical foundations will be applied to a large software system and evaluation examples from the project quality report will be presented. The underlying quality model will be explained in detail as well as the experiences made (e.g. tool handling, surplus value) will be shared.

HIERARCHICAL SOFTWARE QUALITY MODELS – A STEP TOWARDS QUANTIFYING NON-FUNCTIONAL PROPERTIES

Roland Neumann, Lars Grunske, Bernhard Kaiser

Hasso-Plattner-Institute for Software Systems Engineering Potsdam, Germany

{neumann | grunske | kaiser}@hpi.uni-potsdam.de

Abstract. In the last decade, software rapidly replaced hardware in terms of functionality. There are many advantages of this evolution, but new quality assurance techniques for software had to be found. Hierarchically decomposed influence models became necessary to quantify the quality of software. However, there is a missing link between software measures and their meaning for quality. This paper tempts to bridge this gap by presenting decompositions of quality in the embedded environment.

Estimation of Maintenance Tasks

Manfred Bundschuh

AXA Service AG, Cologne, President of DASMA e.V., Germany

bundschuhm@acm.org

Estimation of Maintenance Tasks. *Project estimation often doesn't include the maintenance effort during the lifetime of an application system. But it usually exceeds the other application development costs.* Software maintenance is often defined as the modification of a software product after delivery to correct faults, to improve performance or other attributes, or to adapt the product to a changed

Workshop Report

environment. *Practical experience shows that IT systems do live longer than expected.* It's a common practice that the costs for maintenance are cumulated during the lifetime of a system without controlling the amount and without differentiating between the different kinds of costs. Like in a supermarket the user is afterwards astonished that the many cheap goods in the basket (comparably maintenance requirements) cumulate to a large sum at the cash point. It's only a pity that in software maintenance nothing can be removed from the basket afterwards (to stick to the analogy with the supermarket).

It should always be kept in mind that the IFPUG (International Function Point User Group) definition holds that maintenance tasks do not change the functionality of an application system. If it does so it will be an enhancement instead.

METRICS-BASED ANALYSIS OF ENTERPRISE JAVABEANS COMPONENTS

Andreas Schmietendorf^{1,2}, Reiner Dumke²

¹T-Systems Nova, Entwicklungszentrum Berlin, Germany
andreas.schmietendorf@t-systems.com

²Otto-von-Guericke-University of Magdeburg
schmiete|dumke@ivs.cs.uni-magdeburg.de

Abstract. This article deals with the granularity of EJB-based software components intended for use as “business components”. It begins by providing an overview of Enterprise JavaBeans (EJB) technology, followed by a brief introduction to the basic problems of software measurement applied within the context of component-oriented technologies. Finally, it presents selected results from empirical analyses performed. A total of 40 EJB components were analyzed; the analyses focused on both their atomic characteristic and their characteristic within the framework of an EJB network that conforms to a design pattern. From the knowledge gained, we can derive initial trends in terms of design recommendations for the development of EJB-based business components.

ICEBERG: A DIFFERENT LOOK AT SOFTWARE PROJECT MANAGEMENT

Luigi Buglione¹, Alain Abran²

¹SchlumbergerSema, Rome, Italy

Quebec-University of Montréal, Canada

lbuglione@rome.sema.slb.com

²École de Technologie Supérieure – ETS, Montréal, Canada

aabran@ele.etsmtl.ca

Abstract. Every project – whatever the application field – should be managed taking into account at least four dimensions: Time, Cost, Quality and Risk. To manage these dimensions, a key tool for a Project Manager is to increase project visibility, defined as the amount of information about the project associated with its probability of occurrence. This paper uses the “iceberg” metaphor to introduce the ICEBERG (Improvement after Control and Evaluation-BasEd Rules and Guidelines) approach that can help Project Managers through the use of standard (de jure and de facto) ICT methods and techniques. This approach focuses not only on the management, and measurement, of resources, process and product, but also of the project and the organization itself. A list of candidate measures related to these 5 entities is suggested for a comprehensive software measurement plan in order to reduce project risk.

THE SWEBOK INITIATIVE AND SOFTWARE MEASUREMENT INTENTIONS

Alain Abran, Pierre Bourque, Robert Dupuis
École de Technologie Supérieure – ETS, Montréal, Canada

aabran@ele.etsmtl.ca

Abstract. Articulating a body of knowledge is an essential step toward developing a profession because it represents a broad consensus regarding the contents of the discipline. The IEEE Computer Society, with the support of a consortium of industrial sponsors, has recently published the Guide to the Software Engineering Body of Knowledge (SWEBOK). Throughout this Guide, measurement is pervasive as a fundamental engineering tool. In addition, ISO is at present in the process of adopting this Guide as an ISO Technical Report.

This presentation will provide overviews of the development process that was followed and of the current version of this Guide. In addition, the topic of measurement will be highlighted, both in terms of its presence throughout the ten SWEBOK knowledge areas and of its depth of treatment.

TESTABILITY MEASUREMENT AND SOFTWARE DEPENDENCIES

Stefan Jungmayr
FernUniversität Hagen, Germany
stefan.jungmayr@fernuni-hagen.de

Abstract. Testability is an important quality characteristic of software. A lack of testability contributes to a higher test and maintenance effort. Metrics can be used locate parts of a program which contribute to a lack of testability.

In this paper we present a new approach to define metrics for software dependencies. We use this approach in the context of testability to identify test-critical dependencies, i.e. those dependencies within a system that are critical for test complexity. The results of three case studies show that 1) a small subset of the dependencies within a system has an exceedingly high impact on particular testability metrics, 2) conventional coupling metrics are not good predictors of these test-critical dependencies, 3) dependencies automatically identified to be test-critical are good indicators of design that needs improvement.

New Measurement Intentions in Agent-based Systems Development and Application

Cornelius Wille¹, Reiner R. Dumke¹, Stanimir Stojanov²

¹Otto-von-Guericke-University of Magdeburg, Germany
(wille,dumke)[@ivs.cs.uni-magdeburg.de](mailto:ivs.cs.uni-magdeburg.de)

²University of Plovdiv, Bulgaria
csstani@pu.acad.bg

Abstract. The given paper describes the current situation of the software measurement involvement in software agents, agent-based systems and their development. The presented analysis points out some lacks of control and manage of agents-based systems in the different kinds of agent modelling and design technologies.

Grounded on a general measurement view it was derived new areas of measurement methods, structures, and workflows. Involving some first examples of new measurement principles, we demonstrate the appropriateness of our intentions. Finally, we discuss the application of software agents in the measurement and evaluation process itself.

Software Performance Measures to Assist Decision Makers within the Rational Unified Process

Antonia Bertolino¹, Gaetano Lombardi², Eda Marchetti¹, Raffaella Mirandola³

Istituto di Elaborazione della Informazione, CNR, Pisa, Italy
[{bertolino, e.marchetti}@iei.pi.cnr.it](mailto:{bertolino,e.marchetti}@iei.pi.cnr.it)

Ericsson Lab Italy SpA, Via Anagnina, Roma, Italy
gaetano.lombardi@eri.ericsson.se

³Dip. Informatica, S&P, Università di Roma TorVergata, Roma, Italy
mirandola@info.uniroma2.it

Abstract. Rigorous and automatable approaches to software process measurement are a technology essential for supporting decision makers, such as project managers, in obtaining reliable estimations of relevant properties of the development project and in controlling it. We introduce a methodology, called Propean (Project Performance Analysis), which applies classical techniques of performance analysis to a UML model of the development process. For this purpose, the Real-Time UML profile is considered. The analysis of the UML diagrams produces as an output, in automatable way, the time to completion and the utilization rate of employed resources (personnel in this cases). This paper focuses in the application of Propean to the widespread Rational Unified Process (RUP) adapted to a specific industrial project. We describe the steps necessary for Propean application, the typology of results that can be obtained and how project managers can use them for producing estimations about project completion within fixed schedules and budgets.

A STRATEGY FOR A CREDIBLE & AUDITABLE ESTIMATION PROCESS USING THE ISBSG INTERNATIONAL DATA REPOSITORY

*Alain Abran¹, Reiner Dumke², Jean-Marc Desharnais³, Iphigénie Ndyaje³ and
Christian Kolbe²*

¹École de Technologie Supérieure – ETS, Montréal, Canada

aabran@ele.etsmtl.ca

²Otto-von-Guericke-University of Magdeburg, Germany

dumke@ivs.cs.uni-magdeburg.de

christian.kolbe@student.uni-magdeburg.de

³Quebec-University of Montreal, Canada

desharnais.jean-marc@uqam

ndiayei@hotmail.com

Introduction: the business context. The software estimation process must provide credible input for business decision-making. Most of the time, business managers must rely on incomplete information to make decisions: there is almost always some information lacking or too expensive to gather within the time frame of the decision-making process. Within their field of expertise, decision-makers can make valuable expert judgments on the missing or incomplete information. However, many decision-makers have not mastered the information technology domain and need expert support to fill in the gaps in their knowledge of the subject.

For this reason, the software estimation process must provide decision-makers not only with estimates (the “the numbers”), but also information on the quality and confidence level of that estimate. The key assumptions and the key uncertainties inherent in the estimation process must be conveyed to business managers to help them make informed business decisions on the basis of the estimates provided to them. For example, decision-makers should have a feel for the quality and accuracy of the inputs, as well as for the estimation models used for deriving the estimates.

The estimation process must be credible from a business perspective, and the outcomes of the estimation process must include statements on the credibility of its various components. Furthermore, since major investment decisions are made based on these estimates, the full estimation process should be auditable.

This paper highlights the key elements that make an estimation process both credible and auditable. This includes a discussion on the quality of the input measures (products, processes and resources), the reliability of the productivity models built into an estimation process, the other inputs to the estimation process (key assumptions, constraints and expert judgments) and the type of decisions that should be taken on the basis of the confidence level of the outcomes of the estimation process. This is illustrated with examples from the multi-organizational project repository of the International Software Benchmarking Standards Group – ISBSG.

WHY DOES THE FUNCTION POINT ANALYSIS FIND SO LITTLE ACCEPTANCE?

Robert Hürten

Hürten & Partner Consulting, Blankenheim, Germany
<http://www.huerten-partner.de/>

Present-day situations. Who, like the author of this essay, works with the software metrics, and the Function Point analysis intensively over years, must determine, that these topics are little known by the German IT-specialists. Even those, which were engaged in software metric, show low acceptance. In Germany you will find only some few companies using function point analysis or software metrics. If you look for the causes of this situation, you will find many-layered reasons. The causes lie

- in the general human behaviour,
- in the mode of operation in the software-development,
- in the business management oriented behaviour,
- in the description of the function point analysis in the literature and
- in the lacks in the present-day variations of the Function Point analysis.

Analysis of Software Defects in a Large Evolutionary Telecommunication System

Marek Leszak, Werner Brunck, Gerd Mößler

Lucent Technologies, Optical Networking Group, Nürnberg, Germany
{mleszak, wbrunck, gmoessler}@lucent.com

Abstract. Measurement and evaluation of product and process characteristics is a critical activity throughout the entire software development and evolution lifecycle. It is fundamental to determining whether the software products we develop have the desired functional and non-functional properties; it is fundamental to determining whether we have achieved the desired cost, schedule and quality attributes of our software development and evolution processes.

In this paper several aspects of a previous study of industrial large-scale embedded software development are explored further, by focussing on two related goals. First, we aim to analyze and understand the dynamics of software evolution, across several consecutive project releases. The analysis is based on product & process metrics, especially over time. Second, we perform the first steps towards an empirical model for defect prediction, to achieve better project control, esp. during the hot 'endgame' of a software project. For this we measure various process metrics and relate those to observed pre-delivery defects and other measures of software changes.

The case study presented in this paper has been done in parallel to an on-going development of the MSS product family. MSS is a large embedded hardware/software system for the 'optical internet infrastructure' market. For the first three releases of this product, some process and product metrics data has been collected and analyzed .

This study revealed several interesting and partly surprising results. It also shows the limitations with the granularity of the available data. Further, more detailed studies are necessary to be able to construct adequate predictive models of e.g. defect distribution over time.

Defining Measures for Memory Efficiency of the Software in Mobile Terminals

Hannu Toivonen

Senior Consultant, Nokia Research Center, Finland
hannu.i.toivonen@nokia.com

Abstract. Efficient usage of memory is one of the key cost drivers of the software for mobile terminals. But how to measure, monitor and predict the memory efficiency of the software? This paper introduces a study and results of an undertaking to define a measure for memory efficiency of software. Memory efficiency was defined as the amount of functionality packed per megabyte of memory in the mobile terminal, where the functionality was seen from the viewpoint of the terminal user. Practical measurements were done for two different mobile phones.

Situation and Trends in Software Measurement – A Statistical Analysis of the SML@b Metrics Bibliography

Reiner R. Dumke, Mathias Lothar, Cornelius Wille

Otto-von-Guericke-University of Magdeburg, Germany
(dumke, lothar, wille)@ivs.cs.uni-magdeburg.de

Abstract. The given paper describes the current situation of the software measurement as result of an analysis of the software metrics bibliography available in the Software Measurement Lab at the University of Magdeburg (SML@b).

First we give a general overview about the software measurement situation based on some essential papers. Analysing the current contents of our metrics bibliography, we present the situation of works on the area of software measurement especially from the community point of view.

The paper results are discussed for the motivation in the field of software measurement and evaluation.

THIRD INTERNATIONAL WORKSHOP ON SOFTWARE AND PERFORMANCE

WOSP 2002 – Workshop Report

Andreas Schmietendorf

* *Otto-von-Guericke-Universität Magdeburg, Fakultät Informatik, Institut für Verteilte Systeme, AG Software-Technik, Universitätsplatz 2, D-39016 Magdeburg, Email: schmiete@ivs.cs.uni-magdeburg.de*

1 Overview

After Santa Fe/USA in the year of 1998, Ottawa/Canada in the year of 2000 the 3rd Workshop about the theme of the intersection of Software Engineering and Performance Engineering was realised in this year in Rom/Italy and therefore for the first time in Europe. The aim of the workshop is to provide a discussion and experience platform in the field of software performance engineering for industry experts and researchers from universities. Furthermore, it is valid to identify pending challenges from investigation and industry. The topics on the workshop rich from concrete methods of performance analysis, the relationships between performance engineering and software test, the workload characterisation of new application domains (e.g. web services), UML-based performance models up to necessary processes within the industrial environment to establish software performance engineering.

The term of software performance engineering (SPE) was first coined by Connie Smith in the year 1981 on a conference of the Computer Measurement Group (short CMG). The SPE foundations introduced by Connie Smith over 20 years ago have not yet been incorporated into the current software engineering processes. That means the commercial software development works without a quantifiable and methodical procedure of performance engineering. The difficulty of the transfer of the scientific know-how still exists into the industrial evolution of software.

The presentations on the Workshop in this year reflect this problem too. Only 15% of the in total 40 contributions was provided from the pure industrial environment. Most contributions came at 57,5% from the purely on-campus environment, 17,5% of all contributions were realised in co-operation between industry and university and 10% in the environment of research institutes financed partially publicly.

2 Topic emphases

The workshop was structured by the following topics of session presentations:

- Session 1 – Qualitative and quantitative validation of software systems. (joint with ISSTA¹ session)
- Session 2 – Dynamic analysis. (joint with a ISSTA session)
- Session 3 – Performance analysis in the software lifecycle
- Session 4 – Performance modelling and analysis
- Session 5 – Middleware performance analysis
- Session 6 – Performance evaluation of software architectures
- Session 7 – Software, Performance, or Engineering?
- Session 8 – Extending performance approaches to new application domains

One Highlight of the workshop was the presentation of the invited speaker Daniel Menasce [Menasce 2002]. He spoke within session 7 about the topic „Software, Performance, or Engineering?“ He pursued the following question within the presentation:

Is the term “Performance” in “Software Performance Engineering” redundant? – If Software Engineering is an engineering discipline, then it should produce efficient systems by definition.

In accordance with Menasce, it should be the destination to establish these tasks as a integral component of software engineering. However, diverse unresolved problems which justify an independent consideration of this setting of a task from point of view of Menasce at present face this goal. The following items were called as primary causes for it by [Menasce 2002]:

1. *Lack of scientific principles* – insufficient scientific basics and missing methods respectively models which can be inserted effective in the industrial context.
2. *Education and curricula problems* – the topic of Software Performance Engineering is currently rare considered into the curricula of the universities. For example: A joint IEEE CS/ACM task force works on the “Model Curricula for Computing”, but the performance is overlooked!
3. *IT Workforce Issues* - Many individuals without formal training are employed in IT and learn on the job! Only 50% had a bachelors or other degree with a major or minor in computer science or computer science-related discipline!
4. *Single-user mindset* - Concurrency for physical and software resources (e.g. used networks, database locks, critical sections, and software threads) is typically overlooked by people who write code.

¹ ISSTA International Symposium of Software Testing and Analysis

Workshop Report

5. *Small database mindset* - Most programmers write database access code without taking into account the size of the database. The used tests are done in small development databases typically used to test functionality.

3 Challenges

In addition to the problems identified by Menasce (see previous point) those lead as a matter of course to the corresponding challenges in the field of software performance engineering, further challenges were carried out from all workshop participants. The following 3 workshop sessions were instead carried out.

- First Session - Performance taxonomy
- Second Session – Performance issues in next generation middleware
- Third Session – Relationships between maintainability and performance

From point of view of the author of this workshop report, the following topics with it were discussed primarily:

- Integration of the tasks of software performance engineering within the software development process.
- The task of a so called business-driven integration of existing systems or applications under development requires the consideration of the performance behaviour as „Business Success Factor“.
- Another important topic was the performance engineering activities in context with components. In addition to analytical consideration of components, it can be considered the evolution to performance adaptive components (that means components with an agent oriented behaviour).
- With increasing implementation of business crucial integration solutions on the basis of web service technologies, requirement exists, performance statements in particular in this field to win.

Also in the case of providing performance relevant information about products (e.g. MOM²-products) inserted industrially, it is important to cover the field of integration solutions. First approaches like an industry standard benchmark (SPECjAppServer200x) [SPEC 2002] designed to measure the performance behaviour of J2EE compliant application servers are currently available in a first version.

4 Further Information

Further information about the workshop can be found under the following URL within the internet:

<http://univaq.it/~wosp02/>

² Message oriented Middleware

The 4th International Workshop on Software and Performance (WOSP 2004) will be held from the 14th to the 16th January 2004 in San Francisco Bay Area, California/USA. All proposals for presentations, tutorials and panels will be due late Summer 2003.

In addition to the WOSP meeting it gives also in the German-speaking area a corresponding interest group. This working group within the GI FG 2.1.10 (Software-Measurement) realise each year the so called performance engineering workshop (short PE) and works together with the WOSP working groups. The contributions of the last Workshop (PE 2002) can be found under [Schmietendorf 2002], further information about the planed PE2003 workshop can be found under the following web-side:

<http://ivs.cs.uni-magdeburg.de/sw-eng/us/giak/>

5 Sources

- [Menascé 2002]: Menascé, D. A.: Software, Performance, or Engineering? www.cs.gmu.edu/faculty/menasce.html
- [Schmietendorf 2002] Schmietendorf, A.; Dumke, R.; Hopfer, R.; Scholz, A. (Hrsg.): Tagungsband – 3. Workshop Performance Engineering in der Softwareentwicklung (PE2002)
- [SPEC 2002] SPEC/OSG Java Subcommittee: SPEC jAppServer Development Page, <http://www.specbench.org/osg/jAppServer/>
- [WOSP 2002] Proc. of the Third International Workshop on Software and Performance, ACM Press, Rom/Italy, July 2002

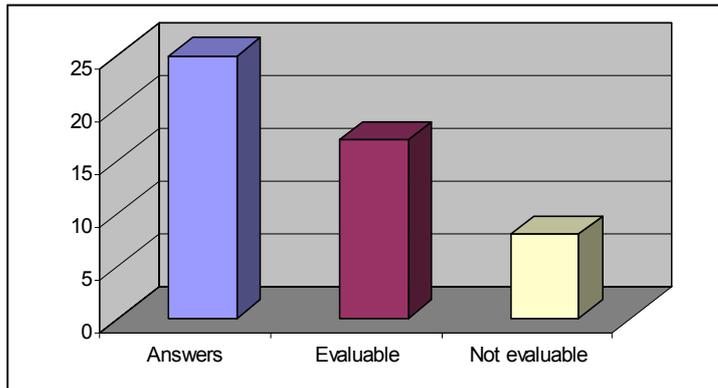
**Software Measurement and -Metrics
in external Enterprises
presented by
Manfred Bundschuh
AXA Service AG, Cologne**

The Inquiry was sent to:

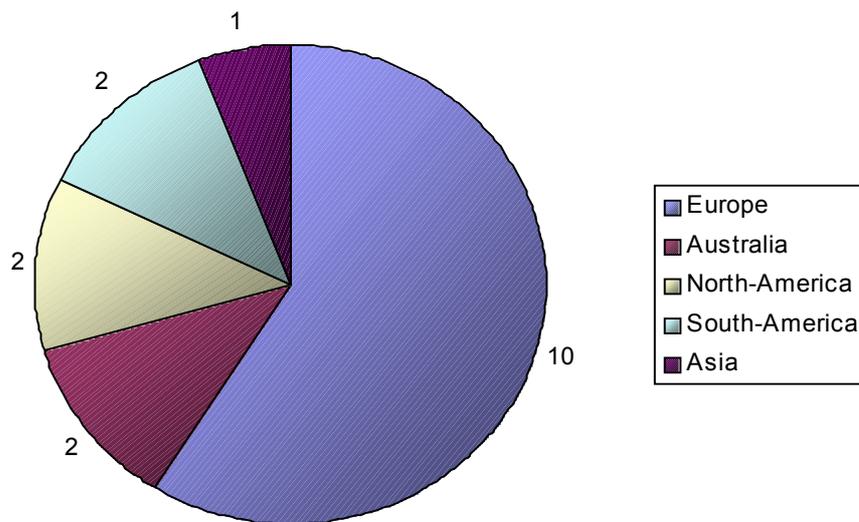
- ◆ **AXA RFI (Request for Informations)**
- ◆ **DASMA e.V.**
- ◆ **GI Working group by Professor Dumke**
- ◆ **MAIN (all European Metrics Organisations)**
- ◆ **Function Point Listserv (IFPUG)**

Answers

	Altogether	from AXA
Answers	25	10
Evaluable	17	5
Not Evaluable	8	5

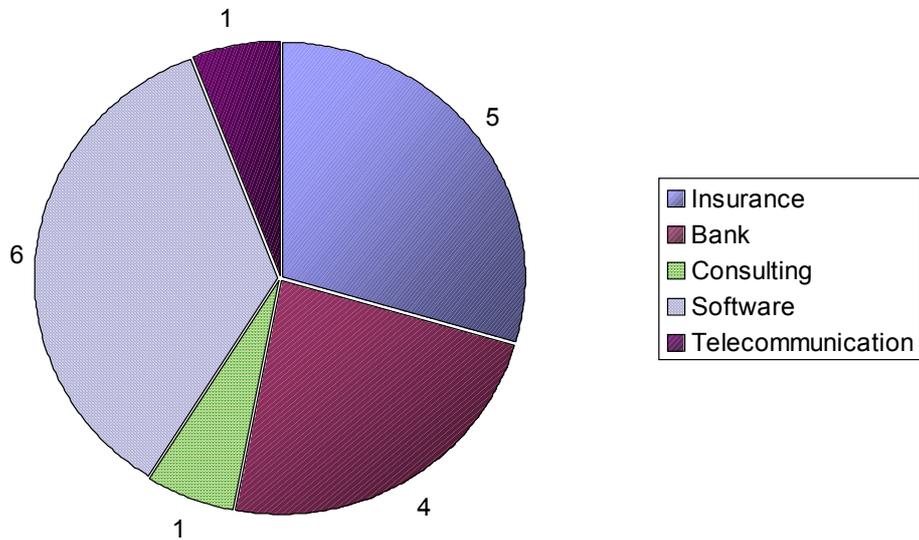


Geographical Distribution of the Enterprises



Locations of Enterprises with evaluable Answers.

Distribution by Industries

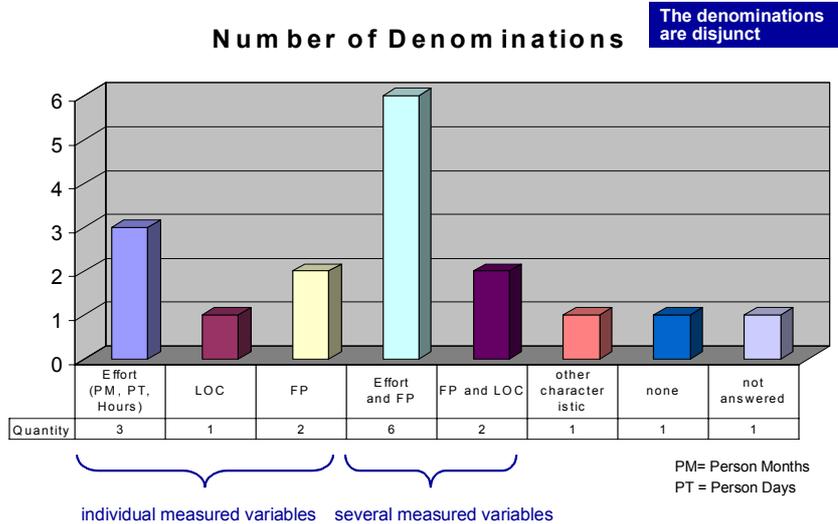


Companies with evaluable answers

- ♦ AXA - Australia
- ♦ AXA - Consulting (for AXA Belgium und AXA Corporate)
- ♦ AXA - Chinaregion
- ♦ AXA - Sun Life
- ♦ AXA Investment Managers
- ♦ Bank of Canada
- ♦ BBI-Consulting AG (Switzerland)
- ♦ Bonndata GmbH
- ♦ CFPS (Software-Enterprise from Brazil)
- ♦ CFPS Embratel - Brazil
- ♦ Dexia Bank Belgium
- ♦ Fidenta
- ♦ National Australia Bank Group
- ♦ ITERGO Informationstechnology GmbH (IT of Ergo Insurance)
- ♦ SwiSMA (Switzerland)
- ♦ SWX Swiss Exchange
- ♦ Technical Computing Division - Software Engineering Department

Question 1

Which size measure do you use for your application systems and/or IT projects ?



Question 1 - Continued

Which size measure do you use for your application systems and/or IT projects ?

Further measured variables, which are used:

- Use Case Points (+ FP und LOC)
- Objects (small; middle; large) (+ FP und LOC)
- WBS + Cost
- WBS (+ PM)
- For Projects: Observation (risk referred) (+ LOC)
- For Operations:
- Availability and performance of Codesystems(+ LOC)
- Tool: ExperiencePro (+ FP)
- Budget / Actual comparison (+ PM)
- Number of defects
- Project (+ PM)
- Specialist area (+ PM)
- Cost Centres (+ PM)
- AE (+ PM)
- Technologies (+ PM)
- Skills (+ PM)

PM = Person Months

Question 2

- ➔ **How is the measurement process organized in your Organisation ?**
- How often and at which milestones ?
 - Who is involved (and how many persons) ?
 - Who is responsible:
 - External staff,
 - Internal staff from a Competence Center,
 - Internal Developers,
 - Internal Users ?
 - Which support is delivered from Management (e.g. SLA's, Bonus)?

Note: Not every participant answered all detail questions

➔ The following methods were mentioned:

- ◆ SDI + Standards
- ◆ Cocomo
- ◆ Zählung Altsysteme
- ◆ Einige Abteilungen zählen
- ◆ only in individual cases pro Project
- ◆ detailed description of process
- ◆ Backfiring
- ◆ Prozentsatzmethode

Question 2 - Continued

- How is the measurement process organized in your Organisation ?
- ➔ **How often and at which milestones ?**
- Who is involved (and how many persons) ?
 - Who is responsible:
 - External staff,
 - Internal staff from a Competence Center,
 - Internal Developers,
 - Internal Users ?
 - Which support is delivered from Management (e.g. SLA's, Bonus)?

Note: Not every participant answered all detail questions.

➔ How often / at which milestones

- ◆ Permanent
- ◆ Daily for operations
- ◆ weekly project report
- ◆ Monthly FP count
- ◆ Quarterly
- ◆ For each project phase
- ◆ At end of project

Question 2 - Continued

- How is the measurement process organized in your Organisation ?
- How often and at which milestones ?
- ➔ **Who is involved (and how many persons) ?**
- Who is responsible:
 - External staff,
 - Internal staff from a Competence Center,
 - Internal Developers,
 - Internal Users ?
- Which support is delivered from Management (e.g. SLA's, Bonus)?

Note: Not every participant answered all detail questions

➔ Who is involved ?

- ◆ 1 st person (FP Count, support for estimation)
- ◆ 2 nd person (part time,, FP Count)
- ◆ Developer
- ◆ Enduser
- ◆ Metrics specialist
- ◆ System analyst
- ◆ External FP Counter
- ◆ 1 FP Counter per team
- ◆ 1 metrics specialist per department
- ◆ Project leader, leader of partial project
- ◆ concerned stakeholders

Question 2 - Continued

- How is the measurement process organized in your Organisation ?
- How often and at which milestones ?
- Who is involved (and how many persons) ?
- ➔ **Who is responsible:**
 - External staff,
 - Internal staff from a Competence Center,
 - Internal Developers,
 - Internal Users ?
- Which support is delivered from Management (e.g. SLA's, Bonus)?

Note: Not every participant answered all detail questions.

➔ Who is responsible ?

- ◆ Internal Staff / Competence Center
- ◆ Developer
- ◆ Project leader
- ◆ Company owner / Department leader
- ◆ Internal Staff
- ◆ Program Sponsor

Question 2 - Continued

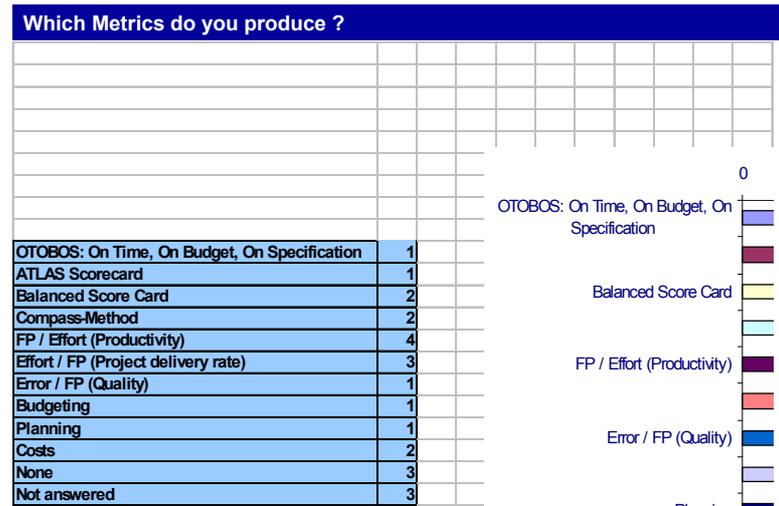
- How is the measurement process organized in your Organisation ?
 - How often and at which milestones ?
 - Who is involved (and how many persons) ?
- Who is responsible:
- External staff,
 - Internal staff from a Competence Center,
 - Internal Developers,
 - Internal Users ?
- ➔ Which support is delivered from Management (e.g. SLA's, Bonus)?

Note: Not every participant answered all detail question.

➔ Management - Support ?

- Program - Sponsor
- Bonus System

Question 3

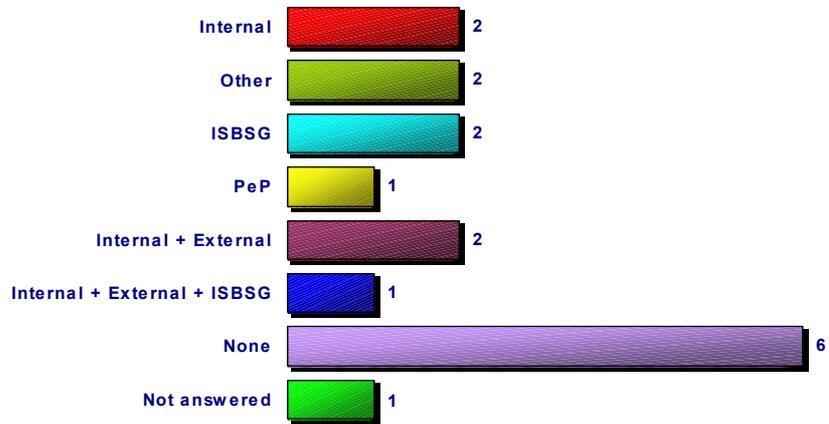


Reference:

- Expense is measured in hours, weeks or month.
- Multiple answers were allowed.

Question 4

Do you perform internal / external benchmarking with the measures and which comparisons do you make ?



The denominations are disjunct

Question 4 - Continued

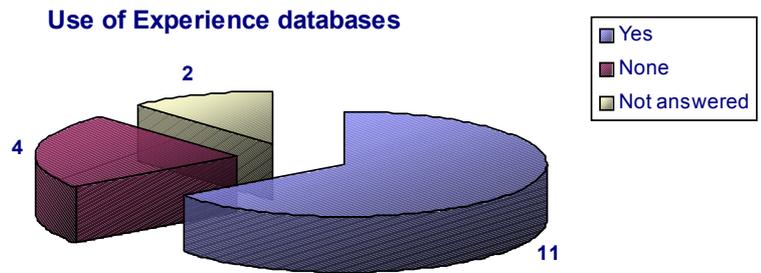
Do you perform internal / external benchmarking with the measures and which comparisons do you make ?

List of individual answers

- ♦ Experience Pro (+ internal)
- ♦ Development and Support Productivity Compass 2000 (+ internal + external)
- ♦ Changes to the previous year and comparisons with "Best Performer" by COMPASS (+ externally)
- ♦ Comparison with empirical values (+ internal)

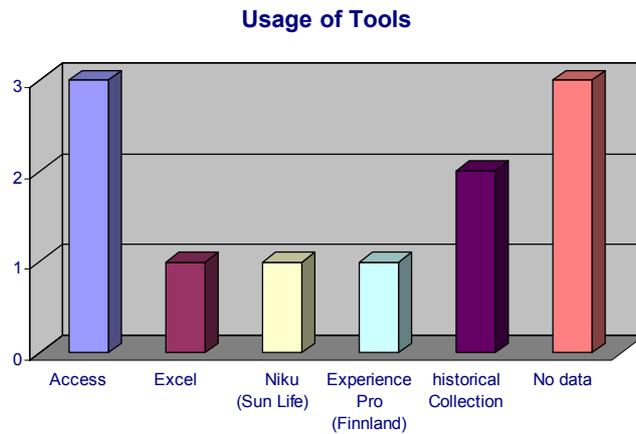
Question 5

Do you have a measurement database (e.g. Access, Excel, ...) with measurement data and metrics ? If yes, which Measures do you collect ?



Question 5 - Continuation

Do you have a measurement database (e.g. Access, Excel, ...) with measurement data and metrics ? If yes, which Measures do you collect ?



Question 5 - Continued

Do you have a measurement database (e.g. Access, Excel, ...) with measurement data and metrics ? If yes, which Measures do you collect ?

List of the stored data in the data base:

Data of Projects:

- ♦ Project start date
- ♦ Project end date
- ♦ Duration in Months
- ♦ Time Accounting
- ♦ FP

Informations about Projects:

- ♦ Project Manager - Name
- ♦ Development/Support Team - Name
- ♦ Project-Type
- ♦ Operating Environment
- ♦ Programming Language
- ♦ Application - Complexity
- ♦ Team - Experience
- ♦ Tool - Set

Dumke, R.; Abran, A.; Bundschuh, M.; Symons, C. (Eds.):

1.1.1.1 Software Measurement and Estimation

Shaker Publ., Aachen, 2002 (315 pages)
ISBN 3-8322-0765-1

The book includes the proceedings of the 12th International Workshop on Software Measurement (IWSM2002) held in Magdeburg in October, 2002, 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 in Bulgaria, Canada, Finland, Germany, Italy and the United Kingdom.

In the proceedings the problems in metrics applications are discussed, the COSMIC-FFP functional size method is investigated further, and new kinds of measurement for object-oriented and agent-based systems are described. Specific aspects of the software development process (risk analysis, code inspection and dealing with remaining defects, among others) and improvement of the development process itself are also addressed. We conclude with our own exploration of ways to improve the process and a discussion of possible new approaches in software engineering and measurement education.

The book will be of interest to software engineering researchers, as well as to practitioners in the areas of project management and quality improvement programs, for both software maintenance and software development.

Dumke, R.; Abran (Eds.):

1.1.1.2 Current Trends in Software Measurement

Shaker Publ., Aachen, 2002 (325 pages)
ISBN 3-8265-9681-1

In this proceedings of the 11th International Workshop on Software Measurement (IWSM2001), new kinds of

measurement for object-oriented and agent-based systems are described, and the COSMIC-FFP functional size method is investigated further. Specific aspects of the software development process (requirements engineering, risk analysis, code inspection and dealing with remaining defects, among others) and improvement of the development process itself are also addressed. We conclude with our own exploration of ways to improve the process and a discussion of possible new approaches.

The book will also be of interest to software engineering researchers, as well as to practitioners in the areas of project management and quality improvement programs, for both software maintenance and software development.

Dumke, R.; Rombach, D. (Eds.):

1.1.1.1.3 Software-Messung und -Bewertung

Deutscher Universitätsverlag (DUV), Wiesbaden, 2002, (254 pages)
ISBN 3-8244-7592-8

This book includes the proceedings of the annual Workshop of the GI FG 2.1.10 held in Kaiserslautern in September 2001. Main categories of the papers collection are

- National initiatives for a virtual competence center of software engineering knowledge services and web-based experimentation,
- Overviews and new results of the methodologies of functional size measurement and cost estimation,
- New measurement approaches for distributed CORBA-based software, agent-based and object-oriented systems,
- New intentions of software measurement for special aspects such as quality model-based risk analysis, performance engineering and cost estimation,
- Further measurement application for software maintenance and project controlling.

The book will be of interest to software engineering researchers, as well as to practitioners in the areas of project management and quality improvement programs, for both software maintenance and software development in general.

Schmietendorf, A.; Dumke, R.; Hopfer, R.; Scholz, A. (Eds.):

1.1.1.1.3.1 *Tagungsband vom 3. Workshop Performance Engineering*

in der Softwareentwicklung (PE 2002), Mai 2002, Dresden, Germany

This proceedings includes the papers from the annual Workshop on Performance Engineering (PE2002) held in Dresden in May 2002. The main topics are

- Software agent-based performance tuning and controlling,
- Performance analysis of enterprise application integration (EAI) solutions,
- Software systems speed management,
- Performance engineering of agent-based systems and system development.

This proceedings can be ordered by a message to Reiner Dumke, University of Magdeburg, Faculty of Informatics.

Dumke, R.; Bundschuh, M. (Eds.):

1.1.1.1.3.2 *Software-Metriken in der Praxis*

Shaker Publ., Aachen, 2002, (154 pages)

ISBN 3-8322-0470-9

This book includes the papers or slides from the presentations of the annual DASMA conference as **Metrikon 2001** held in Dortmund in October 2001. Some of the conference topics are

- Software metrics and project controlling as general approach and as lesson learned from long time experience,
- Psychological aspects of software metrics applications,
- Metrics-based quality analysis of large object-oriented software systems,
- Automation in sourcecode analysis and metrics databasis,
- GQM applications and experiences in industrial environments.

For more information please contact the DASMA office or see the contents description at the GI FG2.1.10 home page (<http://ivs.cs.uni-magdeburg.de/us/giak/>).

Process concepts to guarantee the software performance engineering in enterprise IT organizations

Andreas Schmietendorf

Otto-von-Guericke-Universität Magdeburg, Fakultät Informatik, Institut für verteilte Systeme,
Arbeitsgruppe Softwaretechnik, PF 4120, D-39016 Magdeburg, E-Mail: schmieta@ivs.cs.uni-magdeburg.de



Contents

The PhD-thesis, available only in German language, contains the following topics:

1. Introduction
 - Motivation for the theme
 - Definition of SPE
 - Aims of SPE
 - Aims of the PhD-thesis
 - Structure of the work
2. Current situation of SPE
 - Basics of the performance assessment
 - Influences on the performance
 - Methods for performance assessment
 - Delimitation to other disciplines
 - Derivation of investigation emphases
3. Software Engineering Processes
 - Overview to the tasks of SE
 - Software-management
 - Software quality assurance
 - Software measurement
 - Coverage of SPE in SE processes
4. Detail analysis
 - Effort estimation methods and SPE
 - Performance requirements
 - UML and performance aspects
 - Models, methods and tools for SPE
 - Use of application benchmarks
 - Information requirements for SPE

- Organisation and roles for SPE-tasks
- 5. New operational approaches
 - SPE-oriented process-model
 - Performance risk model
 - Storage and information exchange
 - UML-driven methods for SPE
 - SPE process evaluation
- 6. Validation of the new approaches
 - Application of the PRM
 - Application of PEMM
 - Case studies
- 7. Conclusion and outlook

Publication

German title: "Prozess-Konzepte zur Gewährleistung des Software Performance Engineerings in grossen IT-Organisationen"

Publisher: Shaker-Verlag

Reihe: Magdeburger Schriften zum Empirischen Software Engineering

ISBN: 3-8265-9590-4

Juristo, N.; Moreno, A.M.:

BASICS OF SOFTWARE ENGINEERING EXPERIMENTATION
Kluwer Academic Publishers, 2001 (395 pages)
ISBN 0-7923-7990-X

Basics of Software Engineering Experimentation is a practical guide to experimentation in a field which has long been underpinned by suppositions, assumptions, speculations and beliefs. It demonstrates to software engineers how Experimental Design and Analysis can be used to validate their beliefs and ideas.

The book does not assume its readers have an in-depth knowledge of mathematics, specifying the conceptual essence of the techniques to use in the design and analysis of experiments and keeping the mathematical calculations clear and simple.

Basics of Software Engineering Experimentation is practically oriented and is specially written for software engineers, all the examples being based on real and fictitious software engineering experiments.

"If you are a researcher, you should master the approaches to empirical software engineering described by Juristo and Moreno ...

If you area practitioner, the advice in this book will enable you to read an assess the studies you find in your journals and at your conferences ...

If you are an educator, this book will help you to guide your students in understanding that software engineering is far more than simply having a good technology idea and trying it out on a project." (Pfleeger, S.L.)

SEA 2002:

IASTED International Conference on Software Engineering and Applications

November 4 - 6, 2002, MIT, Cambridge, USA

see: <http://www.iasted.org/conferences/2002/cambridge/sea.htm>

EuroSTAR 2002:

10th European International Conference on Software Testing Analysis & Review November 11 - 15, 2002, Edinburgh, Scotland

see: <http://www.testingconferences.com/eurostar/home/>

PROFES 2002:

4th International Conference on Product Focused Software Process Improvement

December 9 - 11, 2002, Rovaniemi, Finland

see: <http://www.vtt.fi/ele/profes2002/>

Annual Senior Executive eBusiness Summit:

2nd Annual Senior Executive Business Summit

February 4 - 5, 2003, Radisson SAS, Amsterdam, Netherlands

CSMR 2003:

7th European Conference on Software Maintenance and Reengineering

March 26 - 28, 2003, Benevento, Italy

see: <http://rcost.unisannio.it/csmr2003>

WWW 2003:

International World Wide Web Conference

May 20 - 24, 2003, Budapest, Hungary

see: <http://www2002.org/> with the Alternate Track *Web Engineering*
<http://webengineering.org/events/>

ASM 2003:

Applications of Software Measurement

June 2 - 6, 2003, San Jose, CA

see: <http://www.sqe.com/asm/events.asp>

Metrics 2003:

9th International Symposium on Software Metrics

September 3 - 5, 2003, Sidney, Australia

see: <http://metric.cse.unsw.edu.au/Metrics2003/documents/>

METRICS2003CFP.PDF

Conferences Addressing Metrics Issues

36

see also: **OOIS**, **ECOOP** and **ESEC** European Conference

Other Information Sources and Related Topics

- <http://rbse.jsc.nasa.gov/virt-lib/soft-eng.html>
Software Engineering Virtual Library in Houston
- <http://www.mccabe.com/>
McCabe & Associates. Commercial site offering products and services for software developers (i. e. Y2K, Testing or Quality Assurance)
- <http://www.sei.cmu.edu/>
Software Engineering Institute of the U. S. Department of Defence at Carnegie Mellon University. Main objective of the Institute is to identify and promote successful software development practices.
Exhaustive list of publications available for download.
- <http://dxsting.cern.ch/sting/sting.html>
Software Technology Interest Group at CERN: their WEB-service is currently limited (due to "various reconfigurations") to a list of links to other information sources.
- <http://www.spr.com/index.htm>
Software Productivity Research, Capers Jones. A commercial site offering products and services mainly for software estimation and planning.
- <http://fdd.gsfc.nasa.gov/seltext.html>
The Software Engineering Laboratory at NASA/Goddard Space Flight Center. Some documents on software product and process improvements and findings from studies are available for download.
- <http://www.qucis.queensu.ca/Software-Engineering/>
This site hosts the World-Wide Web archives for the USENET usegroup comp.software-eng. Some links to other information sources are also provided.
- <http://www.esi.es/>
The European Software Institute, Spain
- http://saturne.info.uqam.ca/Labo_Recherche/lrg1.html
Software Engineering Management Research Laboratory at the University of Quebec, Montreal. Site offers research reports for download. One key focus area is the analysis and extension of the Function Point method.

38 Metrics in the World-Wide Web

- <http://www.SoftwareMetrics.com/>
Homepage of Longstreet Consulting. Offers products and services and some general information on Function Point Analysis.
- <http://www.utexas.edu/coe/sqi/>
Software Quality Institute of the University of Texas at Austin. Offers comprehensive general information sources on software quality issues.
- <http://www.trese.cs.utwente.nl/~vdberg/thesis.htm>
Klaas van den Berg: Software Measurement and Functional Programming (PhD thesis)
- <http://divcom.otago.ac.nz:800/com/infosci/smr1/home.htm>
The Software Metrics Research Laboratory at the University of Otago (New Zealand).
- <http://ivs.cs.uni-magdeburg.de/sw-eng/us/>
Homepage of the Software Measurement Laboratory at the University of Magdeburg.
- <http://www.cs.tu-berlin.de/~zuse/>
Homepage of Dr. Horst Zuse
- <http://dec.bournemouth.ac.uk/ESERG/bibliography.html>
Annotated bibliography on Object-Oriented Metrics
- <http://www.iso.ch/9000e/forum.html>
The ISO 9000 Forum aims to facilitate communication between newcomers to Quality Management and those who have already made the journey have experience to draw on and advice to share.
- <http://www.qa-inc.com/>
Quality America, Inc's Home Page offers tools and services for quality improvement. Some articles for download are available.
- <http://www.quality.org/qc/>
Exhaustive set of online quality resources, not limited to software quality issues
- <http://freedom.larc.nasa.gov/spqr/spqr.html>
Software Productivity, Quality, and Reliability N-Team
- <http://www.qsm.com/>
Homepage of the Quantitative Software Management (QSM) in the Netherlands
- <http://www.iese.fhg.de/>

Homepage of the Fraunhofer Institute for Experimental Software Engineering (IESE) in Kaiserslautern, Germany

- <http://www.highq.be/quality/besma.htm>
Homepage of the Belgian Software Metrics Association (BeSMA) in Keebergen, Belgium
- http://www.cetus-links.org/oo_metrics.html
Homepage of Manfred Schneider on Objects and Components
- <http://dec.bournemouth.ac.uk/ESERG/bibliography.html>
An annotated bibliography of object-oriented metrics of the Empirical Software Engineering Research Group (ESERG) of the Bournemouth University, UK

News Groups

- <news:comp.software-eng>
- <news:comp.software.testing>
- <news:comp.software.measurement>

Software Measurement Associations

- <http://www.aemes.fi.upm.es>
AEMES Association Espanola de Metricas del Software
- <http://www.asqf.de>
ASQF Arbeitskreis Software-Qualität Franken e.V., Nuremberg, Germany
- <http://www.cosmicon.com>
COSMIC Common Software Measurement International Consortium
- <http://www.dasma.de>
DASMA Deutsche Anwendergruppe für Software Metrik und Aufwands-schätzung e.V.
- <http://www.esi.es>
ESI European Software Engineering Institute in Bilbao, Spain
- <http://www.mai-net.org/>
Network (MAIN) Metrics Associations International
- <http://www.sttf.fi>

Metrics in the World-Wide Web

FiSMA Finnish Software Metrics Association

- <http://www.iese.fhg.de>
IESE Fraunhofer Einrichtung für Experimentelles Software Engineering
- <http://www.isbsg.org.au>
ISBSG International Software Benchmarking Standards Group, Australia
- <http://www.nesma.nl>
NESMA Netherlands Software Metrics Association
- <http://www.sei.cmu.edu/>
SEI Software Engineering Institute Pittsburgh
- <http://www.spr.com/>
SPR Software Productivity Research by Capers Jones
- <http://fdd.gsfc.nasa.gov/seltext.html>
SEL Software Engineering Laboratory - NASA-Homepage
- <http://www.vrz.net/stev>
STEV Vereinigung für Software-Qualitätsmanagement Österreichs
- <http://www.sqs.de>
SQS Gesellschaft für Software-Qualitätssicherung, Germany
- <http://www.ti.kviv.be>
TI/KVIV Belgisch Genootschap voor Software Metrics
- <http://www.uksma.co.uk>
UKSMA United Kingdom Software Metrics Association

Software Metrics Tools (Overviews and Vendors)

Tool Listings

- <http://www.cs.umd.edu/users/cml/resources/cmetrics/C/C++ Metrics Tools by Christopher Lott>
- <http://mdmetric.com/meast11.htm>
Maryland Metrics Tools
- <http://cutter.com/itgroup/reports/function.html>
Function Point Tools by Carol Dekkers

Tool Vendors

- <http://www.mccabe.com>
McCabe & Associates
- <http://www.scitools.com>
Scientific Toolworks, Inc.
- <http://zing.ncsl.nist.gov/webmet/>
Web Metrics
- <http://www.globalintegrity.com/csheets/metself.html>
Global Integrity
- <http://www.spr.com/>
Software Productivity Research (SPR)
- <http://jmetric.it.swin.edu.au/products/jmetric/>
JMetric
- <http://www.imagix.com/products/metrics.html>
Imagix Power Software
- <http://www.verilogusa.com/home.htm>
VERILOG (LOGISCOPE)
- <http://www.qsm.com/>
QSM