INTERNATIONAL WORKSHOP ON SOFTWARE MEASUREMENT – IWSM 2003
Sept 23-25, 2003

Advanced Program

Co-sponsored by:
École de Technologie Supérieure - Université du Québec
Otto von Guericke Universitat (Magdeburg, Germany)
In cooperation with:
COSMIC – Common Software Measurement International Consortium
German Interest Group on Software Metrics
SPIN-Montréal

Location:

École de technologie supérieure
1100 Notre-Dame West Street
Montréal (Québec) CANADA
www.lrgl.uqam.ca/iwsm2003

GENERAL THEME & SCOPE: SOFTWARE MEASUREMENT

Software measurement is one of the key technologies to control or to manage the software development process. Measurement is also the foundation of both sciences and engineering and will contribute significantly to software engineering being recognized as a true engineering discipline.

Conference co-chairs:

Alain Abran &
aabran@ele.etsmtl.ca
Phone: +1-514-396-8632
Fax: +1-514-396-8684

Reiner Dumke
dumke@ivs.cs.uni-magdeburg.de
Phone: +49-391-6718664
Fax: +49-391-6712810

Fees: $200 for the three days or $75 per day
(Including Canadian taxes)

FEES: for authors = none

September 23 (Tuesday)		
8:00	ABRAN, A. DUMKE, R.	Welcome and Introduction
8:30	Paulus & Lopez	Integrated Validation Process for Software Measure
9:30	Sellami & Abran	The contribution of metrology concepts to the understanding and clarification of a proposed framework for software measurement validation
10:00	**Break**	
10:30	Meridji & Ormandjieva	Measuring consistency of the analysis model: an XML approach
11:30	De, Lévesque & Meunier	Software functional complexity measurement with task complexity approach
12:00	**Lunch at ETS Restaurant**	
13:00	Buglione & Gasparro	A quality model for web-based environments: GUFPI-ISMA viewpoint
13:30	Kunz, Abran & Dumke	The prototypical web-based implementation of the QEST model
14:00	- Break	
14:30	Seffah, Padda & Strika	QIM: A tool and knowledge map for usability measurement
15:00	Wille, Abran, Desharnais & Dumke	The quality concepts and sub concepts in SWEBOK: An ontology challenge
15:30	Reitz, Dumke & Schmietendorf	Metrics based comparison of project lines in the industrial software development
16:00	Hoffman & Dumke	When the stomach ails from a headache - Quality assurance in software development of Bosch Diesel Systems
16:30	Palza, Miranda & Abran	Design of a generic performance measurement repository at Ericsson Research Canada
17:00	Schmietendorf & Dumke	Empirical Analysis of available Web Services
17:30	**Closing**	
19:00	**Recommended Activity**	

September 24 (Wednesday)		
8:00	ABRAN, A. DUMKE, R.	Welcome and Introduction
8:30	Nagano & Nishiyama	Measuring functions in OO real-time software
9:00	Dé & Lévesque	Maintenance effort and cost estimation using software functional sizes
9:30	Bevo, Lévesque & Meunier	Toward an ontological formalization for a software functional size measurement method's application process: the COSMIC-FFP case
10:00	- Break	
10:30	Lother & Dumke	Applicability of COSMIC-FFP for BOSCH specifications
11:30	Braungarten, Abran & Dumke	The second generation of the ISBSG Effort Estimation Prototype
12:00	**Lunch Time**	
13:00	Vaugelezang	Applicability of COSMIC Full Function Points in a MIS environment
13:30	Dekkers	Functional size measurement methods are also applicable in enhancement projects
14:00	- Break	
14:30	Korovic	From IT-centric to Business-centric productivity measurement
15:00	Kececi & Shelat	Lessons Learned from Establishing a Software Measurement Program
15:30	Desharnais & Abran	Rapid techniques for measuring Function Points
16:00	Dekkers	Expertise Centre Metrics: The Sogeti way to support project management.
16:30	Buglione & Abran	Assessment of measurement indicators in SPI frameworks
17:00	**Closure**	
19:00	**Recommended Activity**	

September 25 (Thursday) : **ISO 19761 – COSMIC-FFP Tutorials**		
9:00 – 12:00	Instructor: Peter Fagg	***Software Functional User Requirements Validation using Cosmic-FFP*** ***(See tutorial description next page)*** ***Participants will use the Beta version of a software tool to measure the functional size of case studies***
12:00	**Lunch Time**	
13:30 – 14:30	Instructor: Jean-Marc Desharnais	***Exercises in measurement repeatability*** ***Participants will use the Beta version of an expert system software tool (COSMIC-Xpert) to measure functional size with ISO 19761***
14:30	- Break	
15:00-17:00	Instructors: COSMIC team members	***Measurement exercises with ISO case studies from ISO 14143-4***
17:00	**Closure**	

Tutorial by
Peter Fagg (Sept. 25)

Software Functional User Requirements Validation using Cosmic-FFP (ISO 19761)

Aim
The aim of the tutorial is to demonstrate the practical application of the COSMIC-FFP to derive a measurement of function size, and how this can contribute to the overall software development process.

Audience
1. FSM practitioners wishing to learn more about the application of the COSMIC-FFP method by means of a detail analysis of ISO case studies
2. Anyone wishing to learn more about the COSMIC-FFP methods and how it can contribute to the overall development process.

Tutorial Content
Functional Size Measurement Methods (FSMM) measure the size of the functionality delivered by software as expressed in the software Function User Requirement (FUR). Given a complete and accurate FUR, analysts experienced in the FSMM will generally report the same functional size.

In practice however, software FUR are less than accurate and often incomplete. The analyst has to "fill in the gaps", and in doing so a degree of inconsistency is introduced. The set of FUR derived by one analyst may not be the same as another, and this is reflected in the reported size. Quite often the differences are attributed to deficiencies in the FSMM used, but with experienced analysts this is rarely the case.

The COSMIC-FFP FSMM defines a process for measuring size, and one of the steps in that process is called "Mapping". In one scenario, this entails taking the software FUR, extracting the information required to perform the measurement, and reformatting in the form of the Measurement Model. It is this model that is measured.

Gaps or inaccuracies in the FUR must be resolved before measurement. This is essential for accurate and consistent size measurement. As the Measurement Model is based on the functionality expressed in the FUR, can also contribute to the overall software development process; "If you can't measure it, you probably can't build it".

In this tutorial, two of the case studies from ISO-14143-4 are used to construct the COSMIC-FFP Measurement model, and in doing so detect problem that affect measurement. Those problems will also probably affect the development of the software based on the requirement.

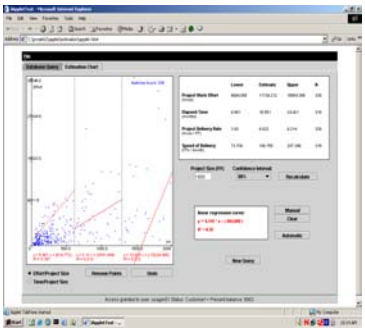
For each case study the COSMIC-FFP method will then be used to measure the functional size. This will demonstrate the application of the COSMIC-FFP method in a practical situation.

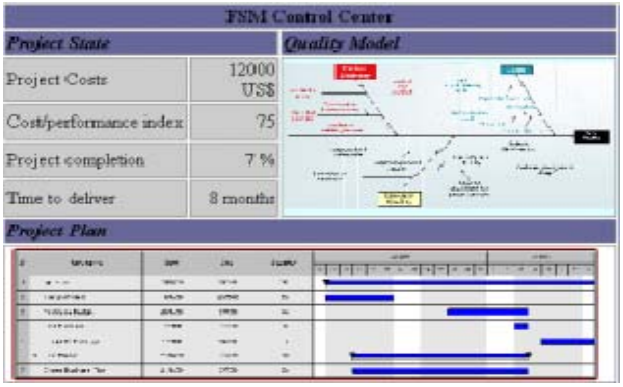
CALL FOR PARTICIPATION

Software eMeasurement

For the development of a web-based eMeasurement service we would like to get your support.

Please attend our survey/questionaire.





Start of survey: Sep. 1st 2003

For your participation please visit the following home page:

http://FSMPortal.cs.uni-magdeburg.de

eMeasurement
powered by SML@b

Mathias Lother
lother@ivs.cs.uni-magdeburg.de
Otto-von-Guericke University
Universitätsplatz 2
39106 Magdeburg / Germany

Functional Size eMeasurement Portal

CALL FOR PARTICIPATION

Since software more and more penetrates our everyday life the need for high quality software products as well as for high quality software development processes increases.

For that reason the SML@b intends to build up an eMeasurement portal supporting the Functional Size Measurement (FSM) as well as the generation, visualization and reporting of the corresponding business data.

Presently, the following eMeasurement Portal functionalities are intended:

1. Reports realized as Cockpit Charts

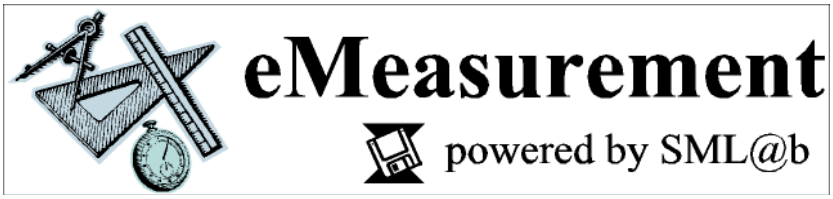
2. Up- and download to a company data base and/or to the ISBSG data base

3. Conferencing/collaboration tools

Please support our development intention with your participation in our survey and let us know, what portal components and conditions are important for you. So you can influence our development decisions.

Please contact us for further details, questions or comments.

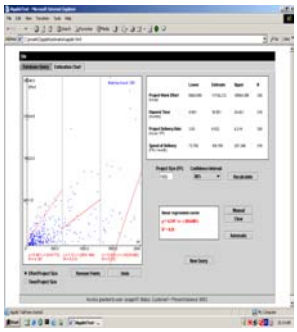
http://FSMPortal.cs.uni-magdeburg.de



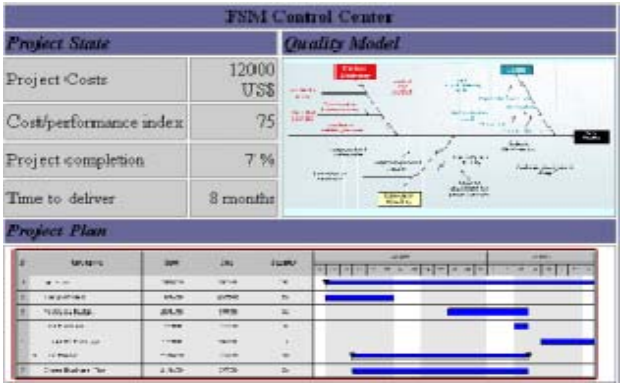
CALL FOR PARTICIPATION

Software eMeasurement

Für den Aufbau eines webbasierten eMeasurement-Dienstes benötigen wir Ihre Unterstützung.



Bitte nehmen Sie an unserer Umfrage teil.



Fragebogenfreigabe 01.09.2003

Zur Teilnahme besuchen Sie uns auf folgender Internetseite:

http://FSMPortal.cs.uni-magdeburg.de

eMeasurement
powered by SML@b

Mathias Lother
lother@ivs.cs.uni-magdeburg.de
Otto-von-Guericke-Universität
Universitätsplatz 2
39106 Magdeburg

Functional Size eMeasurement Portal

CALL FOR PARTICIPATION

Softwareprodukte stellen, nicht zuletzt durch die zunehmende Durchdringung des Alltags, immer höhere Anforderungen an die Qualität der Software und die zu Grunde liegenden Prozesse.

Das SML@b plant deshalb den Aufbau eines eMeasurement-Portals, welches Sie bei der funktionalen Größenmessung (FSM) von Software einerseits und bei der Herleitung, Visualisierung und bei der Reportierung von Business-Daten andererseits unterstützen soll.

Zur Zeit sind folgende Funktionalitäten des eMeasurement-Portals geplant:

1. Reports in Form von Cockpit-Charts

2. Up- und Download zu einer Firmendatenbank bzw. zur ISBSG-Datenbank

3. Conferencing/Collaboration tools

Bitte unterstützen Sie uns mit Ihrer Teilnahme an unserer Umfrage an der Auswahl der „für Sie wichtigen“ Komponenten und Rahmenbedingungen.

Für weitere Fragen und Anregungen wenden Sie sich an unseren Ansprechpartner

http://FSMPortal.cs.uni-magdeburg.de

SM lab

eMeasurement

powered by SML@b

Maturity evaluation of Web Services

ANDREAS SCHMIETENDORF[#*], REINER DUMKE[*]

T-Systems International, Entwicklungszentrum Berlin, Wittestraße 30G, D-13476 Berlin, E-Mail: andreas.schmietendorf@t-systems.com

* *Otto-von-Guericke-Universität Magdeburg, Fakultät Informatik, Institut für verteilte Systeme, Postfach 41 20, D-39016 Magdeburg, E-Mail: {schmiete ,dumke}@ivs.cs.uni-magdeburg.de*

Summary

This paper gives a first insight in an ongoing investigation. Our generic aim is to describe the current situation of available Web Services by the use of metrics. The current maturity of offered Web Services should be evaluated in particular. The first section provides an overview about the Web Service technology. The following section describes the aims of the investigation and the measurement aspects in detail. The third section shows selected results of the empirical analysis and gives a first interpretation of the measurements. Finally the reader will find conclusions and planned further investigations. This research is realized by the Software Measurement Laboratory of the University of Magdeburg and the System- and Technology-Development Group of the Development Centre Berlin.

Overview about the Web Service technology

Currently it is not possible to find one standard definition about Web Services. In general terms provides a Web Service business functionality that can be accessed through an Internet connection. The Web Service technology combine the advantages of the component based development, the Internet technology and more and more the possibilities of the agent technology. Available Web Services vary from simple functionality like weather service or a Internet address locator to complex functionality like a complete order of telecommunication services. The basic architecture of Web Services knows the service-provider (publish a Web Service), the service-broker (managed catalogues of Web Services) and the service-requestor (user of Web Services). With the use of Web Services, the following advantages are expected:

- Easier to use and cheaper than dedicated EAI-frameworks.
- High level of standardisation.
- Synchronous and asynchronous models of communications.
- Support of the component paradigm.
- Provides an easy way for communication through firewalls.
- Broad acceptance within the industry.
- Bridge between heterogeneous technologies.

From the viewpoint of [Abrams 2002] Web Services will be the catalyst for the most innovation in the software industry within the next five years. Currently the Web Service standards are incomplete and we find different approaches within the industries for the implementation of specific aspects. For this reason it is important to know the market leaders and the maturity of available Web Services solutions.
Primary used standard Web Service technologies are SOAP, WSDL and UDDI:

- SOAP (Simple Object Access Protocol), a XML-based protocol for the message oriented information exchange.
- WSDL (Web Service Description Language), this XML-based file describes what the Web Service does and where it is installed and how to use it.
- UDDI (Universal Description, Discovery, and Integration), offer of a list service for the location of a specific Web Services.

In addition to these core technologies for the Web Service implementation it gives further approaches for transaction- and business process-control, for security-aspects and also for SLA-management. Further information about the Web Service technology and usable products for the implementation can be found within the following literature. [Chappel 2002], [Heafel 2002], [Juric 2001], [Schmietendorf 2003]

Aims of the investigation

The presented investigation should be the basis for further automatized analyses. Considering these target positions, our aim was to start with relatively simple measurement aspects. As first step it was necessary to find out available Web Services. For this task we used the Web Service portal provides by XMethods (www.xmethods.com). At the 20th April 2003 were there 332 Web Service registered. For the first analysis we considered approx. 30% of the offered Web Services. As mentioned in the summary the goal of our investigation is to describe the current maturity (e.g. availability, maintainability, scalability) of available Web Service. By the application of the GQM-method (Goal - Question - Metric) we derived from the generic goal “maturity” several questions and finally the following measurement aspects:

- *Overview about available Web Services* – which kind of functionality are currently covered by Web Services?
- *Kindness and maturity of the descriptions* – is the offered functionality described in detail, is a contact person available, is an implementation example available?
- *Frequency of used technologies* – which technology (e.g. .net, Delphi, Cold Fusion) was used for the implementation of the Web Service?
- *Number of provided methods* - it means the number of functions offered by the Web Service interface.
- *Number of input- and output-parameters* - number of the parameters within the function call and the corresponding returned values.
- *Correlation between the wsdl-file and other interface aspects* – the size of the wsdl-file will be measured in Lines of Code (LoC)
- *Web Services availability and performance* - considered is the correct carrying out of a function of the Web Service and the performance behaviour.
- *Frequency of changes of a Web Service* – how often is a new version of a specific Web Service available.

Selected results

As mentioned in the summary, the investigation presented here was not finished during the preparation of this paper. Therefore it is currently not possible to make secured statistically statements. The results shown here represent only first trends.

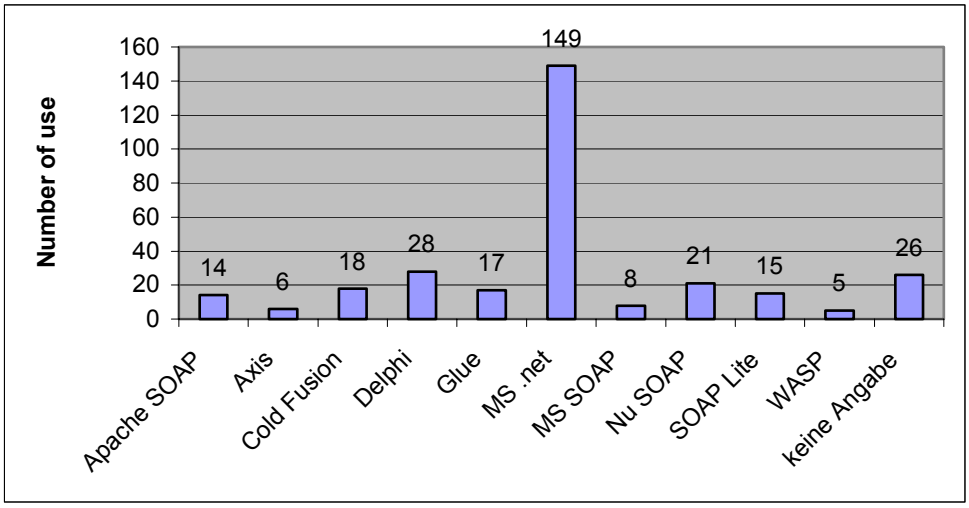


Figure 1: Used technology for the Web Service implementation

Figure 1 shows the frequency of used technologies for implementation of Web Services. The analysis considers all 332 Web Services available under www.xmethods.com (state on 20th April 2003). Considered were only such technologies which at least used 5 times for the implementation of Web Service based solutions. The high part of Microsoft .net based solutions is remarkable. 45% of all Web Services implementation were realised by the use of the Microsoft .net Technology. With nevertheless 8% of all implementation provides no kind of information about the used technology.

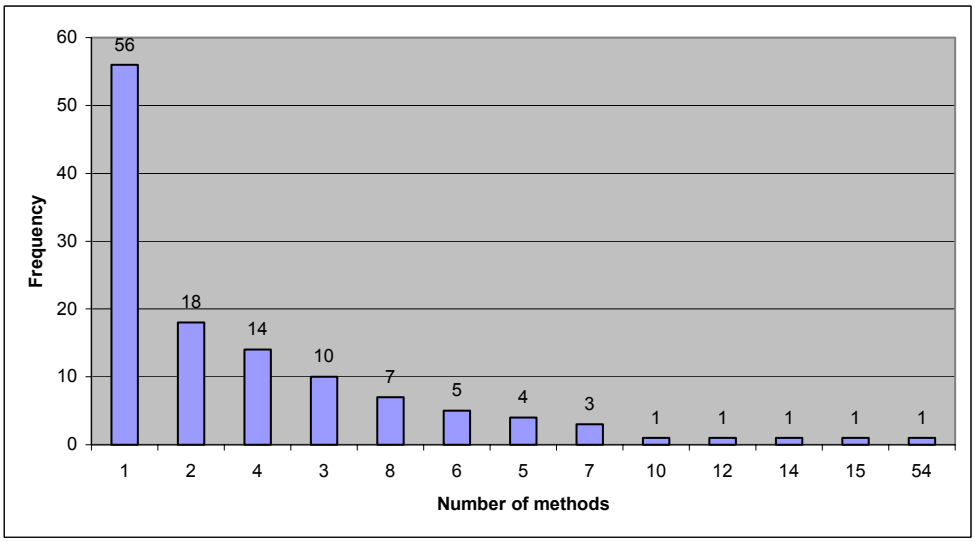


Figure 2: Number of methods per Web Service

Figure 1 shows a distribution diagram about the number of provided methods within the analysed Web Services. 56 of the in total 122 analyzed Web Service offered only one method. The calculated arithmetic mean x_A is **3.36** methods. In the case of 96% of all analyzed Web Services, 1 to 8 methods were provided. This is comparable to the 7±2 law of Miller.

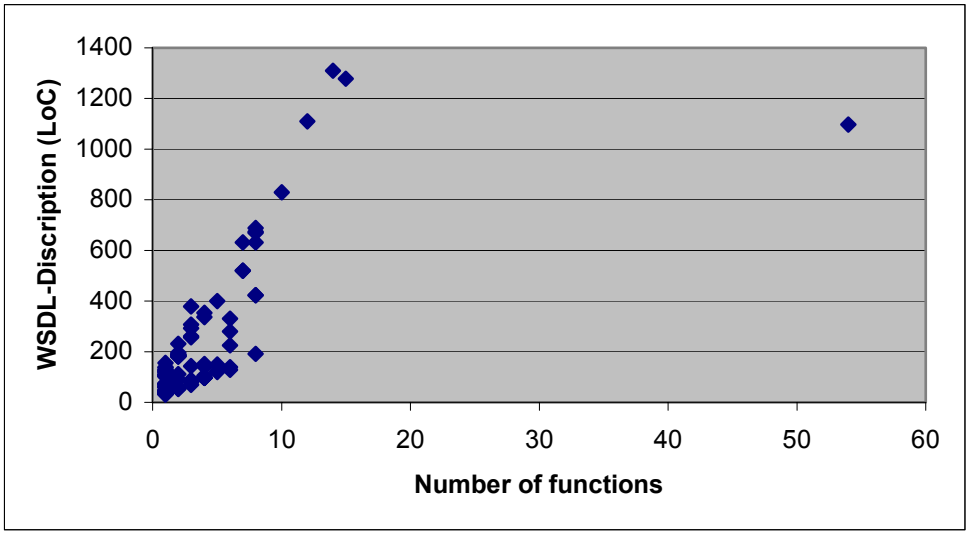


Figure 3: Relations between wsdl-size and number of provided functions

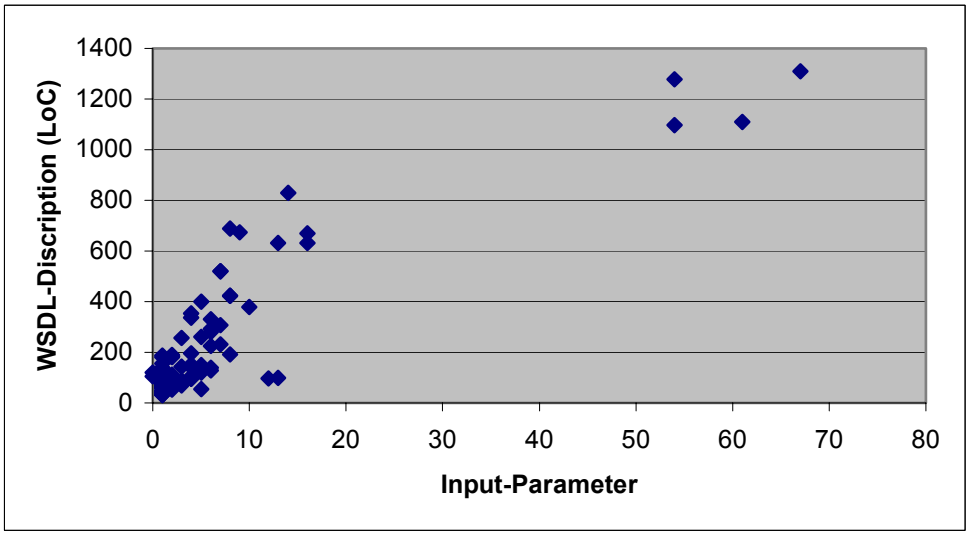


Figure 4: Relations between wsdl-size and number of input-parameters

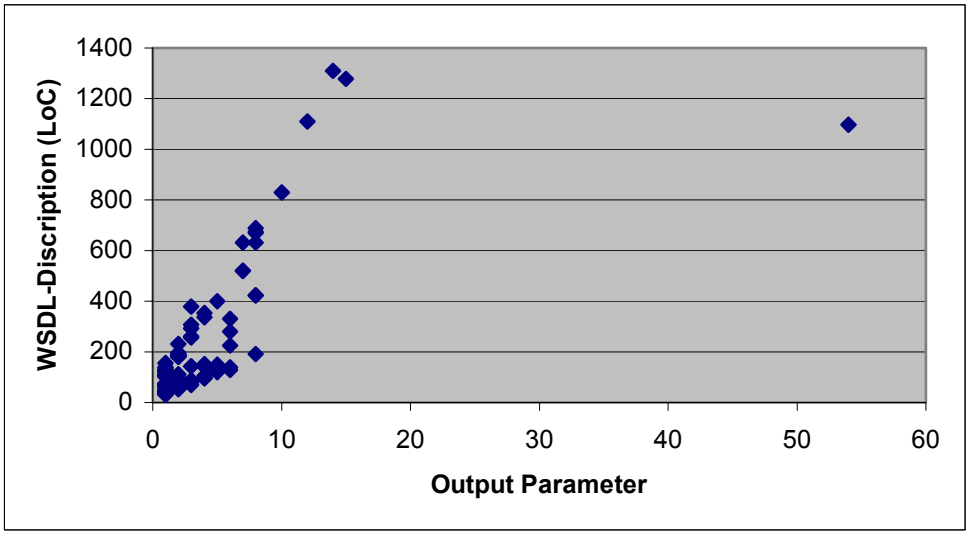


Figure 5: Relations between wsdl-size and number of output-parameters

Figure 3, Figure 4 and Figure 5 shows a scatter plots of measured data points. At the first look, a linear relationship can be assumed. At the following, this relationship should be further quantified by the calculation of the linear regression model.

- Statistical calculations for figure 3 (x – number of functions, y – LoC of the wsdl-file):

  Linear regression model:

  $$y = a + bx = 83.37 + 32.44x$$

From this equation we can say, that in the case of one provided method the wsdl size is 83 LoC plus additional 32 LoC per each method.

Correlation-coefficient r:

$$r = 0.716$$

The correlation-coefficient r shows a light linear correlation between the size of the wsdl-file and the number of provided methods.

- Statistical calculations for figure 4 (x – input-parameters, y – LoC of the wsdl-file):

  Linear regression model:

  $$y = a + bx = 89.52 + 20.34x$$

From this equation we can say, that in the case of one provided input-parameter the wsdl size is 89 LoC plus additional 20 LoC per each input-parameter.

Correlation-coefficient r:

$$r = 0.878$$

The correlation-coefficient r shows a strong linear correlation between the size of the wsdl-file and the number of used input-parameters.

- Statistical calculations for figure 5 (x – output-parameters, y – LoC of the wsdl-file):

  Linear regression model:

  $$y = a + bx = 83.77 + 32.48x$$

From this equation we can say, that in the case of one provided output-parameter the wsdl size is 84 LoC plus additional 32 LoC per each output-parameter.

Correlation-coefficient r:

$$r = 0.717$$

The correlation-coefficient r shows a light linear correlation between the size of the wsdl-file and the number of used output-parameters.

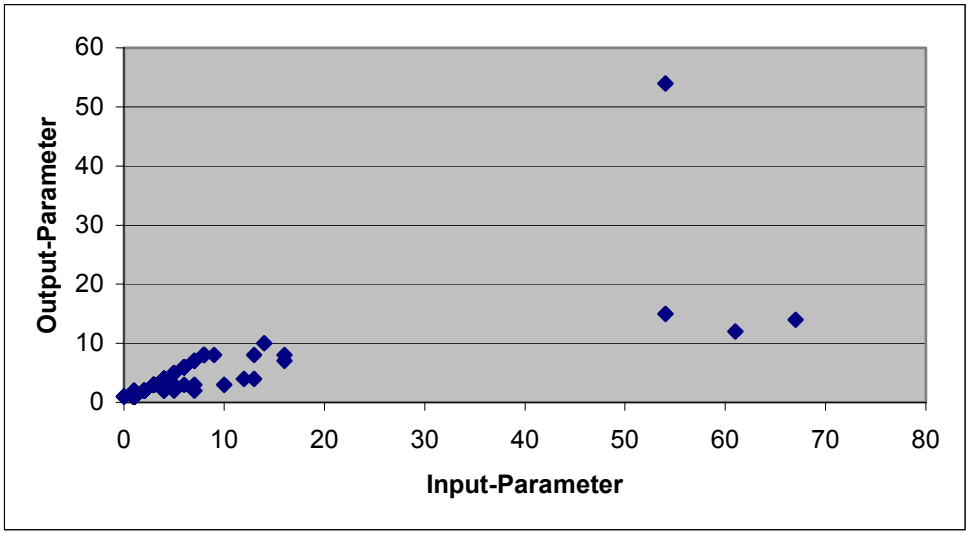


Figure 6: Relations between input- and output-parameters

An further point are the potential relations between input and output-parameters, shown in Figure 6. This information is in particular interesting for the evaluation of the Web Service granularity, e.g. high granularity means few input parameters but many output parameters.

- Statistical calculations for figure 6 (x – input-parameters, y – output-parameters):

  Linear regression model:

  $$y = a + bx = 1.43 + 0.37x$$

From this equation we can say, that in the case of one used input parameter the number of output parameters is 1.43 plus additional 0.37 per each input parameter.

Correlation-coefficient r:

$$r = 0.738$$

The correlation-coefficient r shows a light linear correlation between the number of input parameters and the number of output parameters.

The value of the correlation coefficient r ranges between –1 and +1. We used the following interpretations: less than 0.6 no correlation, less than 0.7 poor correlation, less than 0.8 light correlation, less than 0.9 strong correlation, less than 1.0 direct correlation.

In evaluation of the carried out investigations by the use of regression models, the following hypotheses can be derived. (only some selected points).

- The wsdl-file contains only a description of the provided methods.
- The granularity of provided Web Services is currently low.
- The arithmetic mean of a typical wsdl-file is currently 192 LoC
- The number of methods correlates direct with the number of output-parameters (r = 0.99).
- The functionality of Web Services is comparable to EJB business components.
- The arithmetic mean of the input-parameters is 5.05 and 3.34 for the output-parameters.

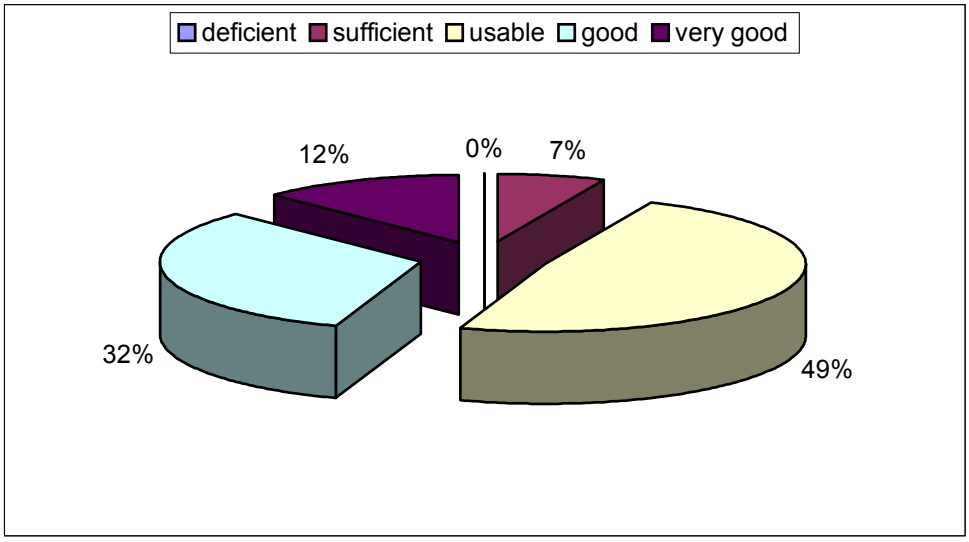


Figure 7: Evaluation of the Web Service description

The successful use of a Web Service within an industrial context depends strongly on the maturity of the Web Service description. This description must contain at least information about the provided functionality described by the wsdl-file and the location (e.g. a URL) of the Web Service within the internet. From our point of view the wsdl-specification is not enough to describe the whole behaviour of a Web Service. For the effective use further information are required. The following points describes a complete description of a Web Service:

- Only wsdl-specification and access point – **a**.
- General textual description about possible functions - **b**.
- Contact person in the case of problems - **c**.
- Contributed clients for the Web Service - **d**.
- Example of the application of the Web Service - **e**.
- Information about possible errors - **f**.
- Information about the quality behaviour - **g**.
- Explanation of semantic aspects - **h**.

The used evaluation model can be described as follows:

- *Deficient* – The description covers only aspect a and b. The use of the Web Service within an industrial context is not possible.
- *Sufficient* – The description covers aspect a, b and c. The use of the Web Service within an industrial context is possible but combined with risks.
- *Usable* – The description covers aspect a, b, c and d. The use of the Web Service within an industrial context is possible.
- *Good* – The description covers aspect a, b, c, d and f. The use of the Web Service within an industrial context can be recommended.
- *Very good* – The description covers all mentioned aspect (a until h). Furthermore considers the Web Service commercial aspects and provides service level agreements (SLA).

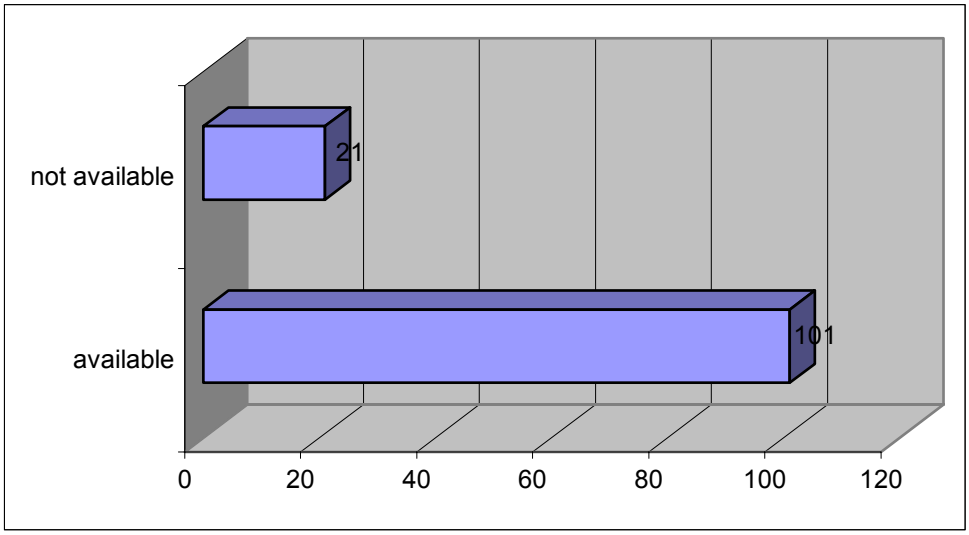


Figure 8: Web Service availability (22.04.2003)

The realised investigation considers also a test of the availability (see figure 8). In this case, the offered functions of the Web Service were carried out. 17% of all the analysed Web Services were not available. However, it is only a random sample here. For secured statements, it is necessary to observe the Web Services for a longer period.

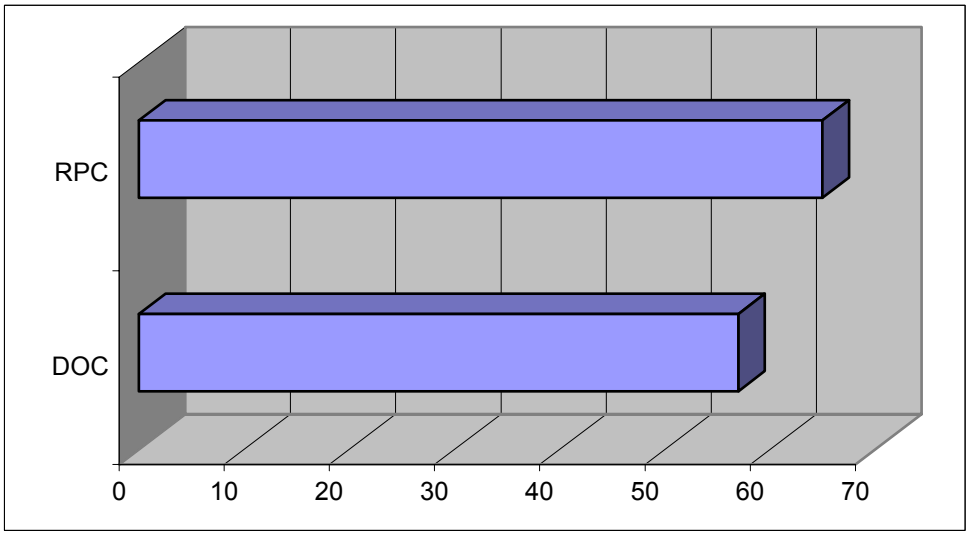


Figure 9: SOAP implementation types RPC vs. DOC

An other investigation considers the specific kind of the implementation. The SOAP-protocol can be used by the use of the document exchange model (short DOC) and the remote procedure call model (short RPC). The DOC model describes only the transfer of a message type (not the internal structure), the RPC model describes the specific structure of a method call and also the response format. Figure 9 shows the results of the investigation, currently both techniques are used in the same manner. But, the most of the currently new registered Web Services considers the DOC-model.

The empirical analysis represented here were realized within approx. 1 month. Within this time period the number of Web Service under XMethods increased from 332 Web Services (state on the 20th April 2003) to 360 Web Services (state on the 3rd

May 2003) at the end of the analysis. Weekly were there approx. 10 to 15 new service registered.

Conclusions and further works

In this paper we have given a first introduction about the external view of within the internet provided Web Services. This view was support by the use of empirical analysis of more than 100 Web Services. Under consideration of these results a first appraisal can be carried out.

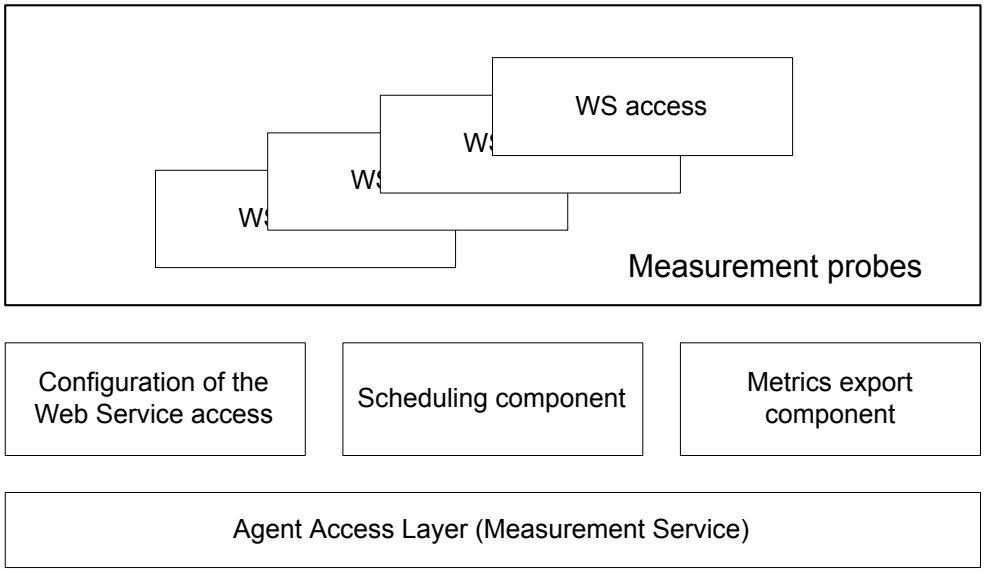


Figure 10: Planed Measurement Agent

From our received experience we derived many further activities. The most important goal of further investigations is its automatization. The represented measurements represent only a random sample. To receive valid trends over a timed period it is planed to implement a agent based measurement system. This system under development has the possibility to measure the availability, the performance, the functionality and the complexity of a specific Web Service from the users (or better integrators) point of view. Figure 10 shows an overview about the architecture of the whole measurement system. By the use of this measurement agent we plan to expand the empirical evaluation to other UDDI-compatible directories. Furthermore, an enlargement of the used statistics evaluations is planned. E.g. consideration of the multiple linear regression techniques and non linear relationships between the used parameters.

References

[Abrams 2002] Abrams, C.: Web Services Scenario: Setting and Resetting Expectations, Gartner European Symposium/ITxpo, Cannes/France, November 2002

[Chappell 2002] Chappell, D.; Jewell, T.: .Java Web Services, O Reilly & Associates, Sebastopol/CA, 2002

[Heafel 2002] Monson-Haefel, Richard: EJB 2.1 Web Services: Part 1, Monson-Haefel's Guide to Enterprise Java Beans, http://www.theserverside.com/resourcen/arti-cels/MonsonHaefel-Column2

[Juric 2001] Juric, M. B.; Basha, S. J.; Leader, R.; Nagappan, R.: Professional J2EE EAI. Wrox Press Ltd., Birmingham/UK

[Schmietendorf 2002] Schmietendorf, A.; Dumke, R.: Metrics-based Analysis of Enterprise Java Beans Components, in Dumke, R. et al.: Software Measurement and Estimation - Proc. of the 12th IWSM, S.137-152, Shaker-Verlag Aachen, Oktober 2002

[Schmietendorf 2003] Schmietendorf, A.; Lezius, J.; Dimitrov, E.; Reitz, D.: Web-Service based EAI-solutions, Chapter in Knuth, M. (Eds.): Web Services, Software & Support Verlag, Frankfurt/Deutschland, 2. Quartal 2003 (only in German language)

[XMethods 2003] Hong, T.; Hong, J.: Web Service directory, www.xmethods.com, San Jose, California/USA, Mai 2003

Awareness for professional IT Projekt Estimation

- A Presentation for convincing Management

presented by
Manfred Bundschuh

1

Agenda

- **Capers Jones‘ Experiences**
- **Be aware of Resistence**
- **To do or not to do ?**
- **The Case for Function Points**
- **An Agenda for the Discussion of the most important Estimation Themes**

2

Capers Jones‘ Experiences

- **Introduction of professional Project Estimation costs an Organisation far less then only one failed Project.**
- **Project Leaders who do not accomplish Project Estimation professionally, do act gross negligent!**

3

Be aware of Resistence

Resistance

Is natural and unavoidable	**Expect Resistance!**
Appears offer undercover	**Find Resistance!**
Has many causes	**Understand Resistance!**
Discuss the scruples, not the arguments!	**Confront Resistance!**
Consider that there is always more than one way to resolve Resistance!	**Manage Resistance!**

4

To do or not to do ?

- Are your Project Leaders telling you, that they do have no time for Function Point Counting and / or Project Estimation?
  - ➔ Then they are like an Architect who does not have time to calculate the Statics of the Bridge, which he has to construct!
- If a Project Leader really does not have time for Function Point Counting and Project Estimation
  - then there is something awfully wrong at the beginning of his Project and he should consider to stop it!
- Unprofessional Estimation is a top risk!
- Estimation is like firefighting
  - professional done will it help you, otherwise it will burn you!

5

The Case for Function Points

- Function Point Documentation is End User Documentation
- Function Point System Border Diagramms can be reused as Architecture Diagrams:
  - Processes: Screens, Batches
  - Data: Files, Reports
  - Interfaces
- Function Points are a Measure of
  - Size
  - End User Functionality
- Function Points are a Base for
  - Interface Complexity Measurement
  - Estimation
  - Test Case Definitions
  - Metrics
  - Benchmarking

6

An Agenda for the Discussion of the most important Estimation Themes

- Object of Estimation
- Timing for Estimation
- Precision of Estimation
- Effort for Estimation
- Monitoring / Tracking of Estimation (Requirements- /Scope Creep, Bifurcation)
- Tools for Estimation
- Honesty of Estimation
- Experience of Estimation
- Culture for Estimation

7

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

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.

Schmietendorf, Andreas:

- ***Process concepts to guarantee the software performance engineering in enterprice IT organizations***

Shaker-Verlag., Aachen, 2001 (194 pages)
ISBN 3-8265-9590-4

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“

Albert Endres & Dieter Rombach:

- ***A Handbook of Software and Systems Engineering***

Pearson Education Limited, Essex, 2003 (327 pages)
ISBN 0-321-15420-7

Computers are the most pervasive tools of modern society. Their development relies on advanced methods of software and systems engineering. Based on repeated and consistent observations, key lessons of these fields can now be formulated into rules or even laws, providing initial building blocks towards a theoretical foundation that is essential for further research, for teaching and for the practice of software development.

Intended as a handbook for students and professionals alike, this book is the first to identify and discuss such rules and laws. They are largely independent of technologies, and thus form a basis for the principles underlying software and system engineering. Software and system engineers should be aware of this proven body of knowledge, to ensure professionalism and due diligence in their work.

The book is structured around the software development lifecycle. It begins with requirements definition and goes on to maintenance and withdrawal. In different process models, these tasks have different importance or are applied in a different sequence, or even cyclically. The book provides the reader with:

- clear statement of software and systems engineering laws and their applicability
- empirical evidence that proves the usefulness of the material covered
- unique knowledge to apply in an industrial setting.

John C. Munson, PH.D.:

Software Engineering Measurement

Auerbach Publications, Boca Raton, Florida , 2003 (443 pages)
ISBN 0-8493-1503-4

The author describes how to manage software development measurement systems, how to build software measurement tools and standards, and how to construct controlled experiments using standardized measurement tools.

The book answers three fundamental questions. First, exactly how do you get the measurement data? Second, how do you convert the data from the measurement process to information that you can use to manage the software development process? Third, how do you manage all of the data?

By demonstrating how to develop simple experiments for the empirical validation of theoretical research and showing how to convert measurement data into meaningful and valuable information, Software Engineering Measurement will show you how to use your measurement information for immediate, software process improvement.

Features:

- Explains how to apply scientific method to software measurement
- Develops static and dynamic measurement techniques

Discusses modelling techniques to establish relationships between software attributes that can be measured and those that cannot.

Conferences Addressing Metrics Issues

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 Enginnering**
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**

PE2003:

4. Workshop Software Performance Engineering
17. Juni 2003 in Magdeburg,
see: http://ivs.cs.uni-magdeburg.de/~schmiete/peak/peak.htm

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**

CONQUEST 2003:

Conference on Quality Engineering in Software Technology
September 17 - 19, 2003, Nuremberg, Germany
see: http://www.asqf.de/deu/termine/termine.php?ID=147

IWSM2003:

13th International Workshop on Software Measurement
September 23 - 25 in Montreal, Kanada,
see: http://lrgl.uqam.ca/iwsm2003/

IFPUG 2003:

IFPUG 2003 Annual Conference
September , 2003, Scottsdale, AZ, USA
see: http://www.ifpug.org/conferences/

Conferences Addressing Metrics Issues

ISESE 2003:

IEEE International Symposium on Empirical Software Engineering
September 30 - October 1, 2003, Roma, Italy
see: `http://ese.uniroma2.it/events/eseiw2003/isese2003/`

UML 2003:

Fourth International Conference on the Unified Modelling Language
October 20 - 24, 2003, San Francisco, USA
see: `http://www.umlconference.org/`

Metrikon 2003:

DASMA Metrik Kongress
10.-11. November an der FH Neu-Ulm,
see: `http://www.dasma.org/`

EuroSPI 2003:

European Conference on Software Process Improvement
December 10 - 12, 2003, Birmingham, UK
see: `http://www.eurospi.net/`

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

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

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

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

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

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

- **http://divcom.otago.ac.nz:800/com/infosci/smrl/home.htm**
  The Software Metrics Research Laboratory at the University of Otago (New Zealand).

- **http://ivs.cs.uni-magdeburg.de/sw-eng/us/**
  Homepage of the Software Measurement Laboratory at the University of Magdeburg.

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

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

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

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

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

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

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

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

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

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

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

News Groups

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

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

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

Software Measurement Associations

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

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

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

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

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

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

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

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

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

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

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

Software Metrics Tools (Overviews and Vendors)

Tool Listings

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

- **http://mdmetric.com/meastl1.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