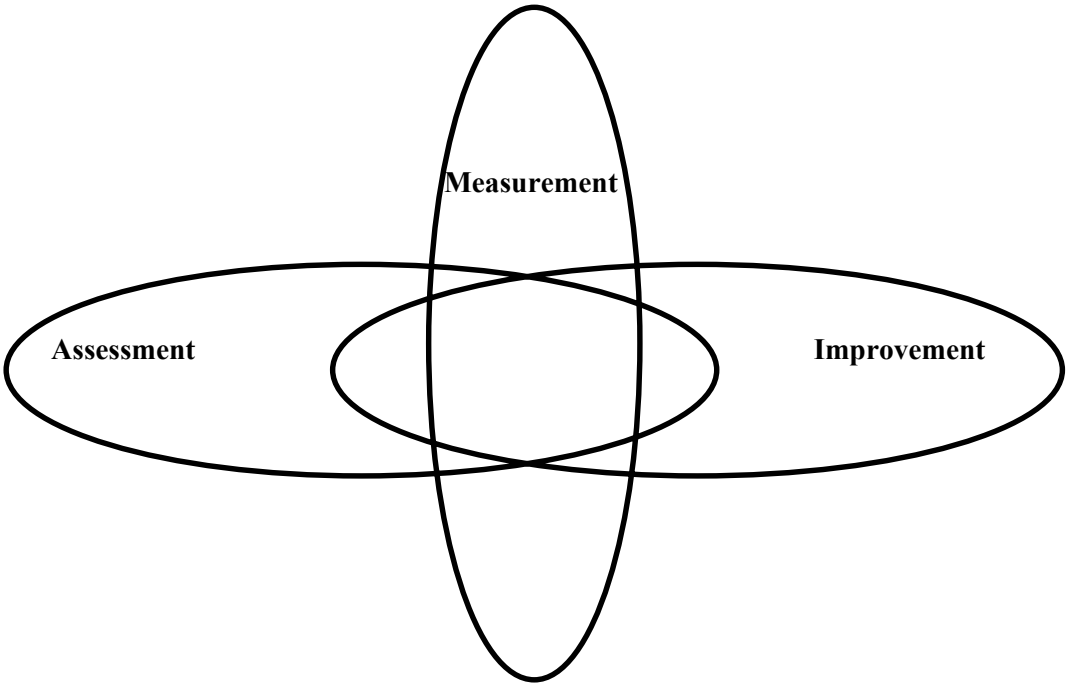
Volume 1, Number 1, September 1996

METRICS NEWS

Journal of the GI-Interest Group on Software Metrics



Editors: ***R. Dumke, C. Ebert, E. Rudolph, H. Zuse***

Otto-von-Guericke-Universität

Magdeburg
The *METRICS NEWS* can directly be ordered from the Editorial Office (for address see below).

Editors:

Reiner Dumke
Professor on Software Engineering,
University of Magdeburg, Faculty of Informatics,
Postfach 4120, D-39016 Magdeburg, German
Tel.: +49-391-6718868, Fax: +49-67-12810
email: dumke@irb.cs.uni-magdeburg.de

Christoph Ebert
Dr.Ing. in Computer Science
Alcatel Telecom, Switching Systems Division,
Fr. Wellensplein 1, B-2018 Antwerpen, Belgium
Tel.: +32-3-240-4081, Fax: 32-3-240-9935
email: ebertc@btmaa.be.alcatel.be

Eberhard Rudolph
Professor on Software Engineering
Hochschule Bremerhaven, FB2 System Analysis,
Mozartstr. 37, D-27570 Bremerhaven, Germany
Tel.: +49-471-26142, Fax: +49-471-207389
email:

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

Editorial Office: Otto-von-Guericke-University of Magdeburg, Faculty of Informatics, Postfach 4120, 39016 Magdeburg, Germany
Tehnical Editor: Dr. Achim Winkler
The journal is published in one volume per year consisting of two numbers.
The price 1996 for a single copy is DM 30,- for institutions and DM 20,- for private persons incl. postage handling.

EDITORIAL

This is the first issue of a new scientific journal in the field of software metrics and related to qantitative aspects, the

METRICS NEWS.

The title including the kind of this journal as a summarizing of software metrics top themes (as position papers), chosen papers of our metrics workhops, and *news* (as informations about the software metrics research area in the world, the new books and conferences on this area). The editors are working many years of the software metrics domains as measurement frameworks, function point analysis, mesurement theoretical view, and practical applications.

The background of the METRICS NEWS contents are the member of the GI-interest group on software metrics founded on 1991. All members from the industry or academia are invited to present their experience or research results on the area of software quality assurance, software metrics, process management, software measurement frameworks etc.

The english language was chosen in corresponding the international character of our research contacts and results and as European initiative.

The editors are grateful to the Otto-von-Guericke University of Magdeburg for publishing this journal.

We hope that the new journal will be helpful and successful to improve the software development processes and products.

The Editors

Call for Participation

6th International Workshop on Software Metrics

of the German-Quebec Working Group,
of the German Interesting Group on Software Metrics,
of the German Interesting Group on Software Quality Assurance
in Object-Oriented Systems

September 19 - 20, 1996
University of Regensburg

Thursday Morning Schedule:

9:30 - 10:00 : Welcome: Prof. Lehner, Prof. Abran, Prof. Dumke

10:00 - 13:00 : New Software Measurement Applications and Paradigms

(11:30 - 11:45 : Coffee and Refreshments)

Ebert (Alcatel, Belgien): *Applying Knowledge-Based Techniques to Software Quality Management*

Foshag (SES Muenchen): *Metrics for Evaluation of Maintenance Service*

Schmelz (University of Jena): *The Use of Factorial Analysis in the Area of Software Metrics*

Baisch/Ebert (Alcatel, Stuttgart): *On a Neural-Fuzzy Technique with GAOptimization for Software Quality Models*

Desharnais/Pierre (CRIM,Canada): *Post-Measurements Validation Procedures of Function Point Counts*

13:00 - 14:30 : Lunch

Thursday Afternoon Schedule:

14:30 - 18:30 : Object-Oriented Software Metrics

(16:00 - 16:15 : Coffee and Refreshments)

Fetcke/Abran/Nguyen (CRIM, Canada): *The Mapping of OO-Jacobson Method to the Measurement Concepts of Function Points Analysis*

Dumke (University of Magdeburg): *Really Object-Oriented Software Metrics*

Hitz/Stiller (University of Wien): *A Versatile Framework for the Collection of Object-Oriented Metrics*

Call for Participation

Brian, L.C: *Defining and Validating Design Coupling Measures in Object-Oriented Systems*

Morschel, I.: *Quality and Productivity Improvement in Object-Oriented Software Development - Experiences and Lessons Learned*

Further Lecture to Object-Oriented Software Measurement

20:00 : Social Event

Friday Morning Schedule:

9:00 - 12:30 : Measurement Frameworks

(10:30 - 10:45 : Coffee and Refreshments)

Zuse (TU Berlin): *A Measure Information System*

Winkler/Dumke/Koeppe (University of Magdeburg): *Efficiency and Maintainability of JAVA Applications*

Abran/Maya/Bourque (CRIM, Canada): *The Extension of Function Points Technique to the Measurement of very small Maintenance Requests*

Mittelmann (VOEST Linz): *Implementation of a Measurement Plan in an Industrial Environment*

Gilles St/Amant (CRIM, Canada): *A Proposal of a Quality Model for Web Content*

12:30 - 14:00 : Lunch

Friday Afternoon Schedule:

14:00 - 15:30 : Projects Planning and Conclusions

Position Papers

Growth and Acceptance of Function Points

by David H. Longstreet
and Raffaela Ibba

Abstract

Systems continue to grow in size and complexity. They are becoming more and more difficult to understand. Improvement of coding tools allows software developers to produce large amounts of software to meet an ever expanding need from users. As systems grow a method to understand and comunicate size needs to be used. Funcion Point Analysis is a structured technique of problem solving. It is a method to break systems into smaller components, so they can be better understood and analyzed. This article describes Function Points Analysis and industry trends using function points.

Introduction

Human beings solve problems by breaking them into smaller understandable pieces. Problems that may appear to be difficult are simple once they are broken into smaller parts -- dissected into classes. Classifiying things, placing them in this or that category, is a familiar process. Everyone does it at one time or another -- shopkeepers when they take stock of what is on their shelves, librarians when they catalog books, secretaries when they file letters or documents. When objects to be classified are the contents of systems, a set of definitions and rules must be used to place these objects into the approapriate category, a scheme of classification. Function Point Analysis is a structured technique of classifying components of a system. It is a method to break systems into smaller components, so they can be better understood and analyzed. It provides a structured technique for problem solving.

In the world of Function Point Analysis, systems are divided into five large classes and general system characteristics. The first three classes or components are External Inputs, External Outputs and External Inquires each of these components transact against files therefore they are called transaction. The next two Internal Logical Files and External Interface Files are where data is stored that is combined to form logical Information. The general systems characteristics assess the general functionality of the system.

Brief History

Function Point Analysis was developed first by Allan J. Albrecht in the mid 1970s. It was an attempt to overcome difficulties associated with lines of code as a measure of software size, and to assist in developing a mechanism to predict effort associated with software development. The method was first published in 1979, then later in 1983. In 1984 Albrecht refined the method and since 1986, when the International Function Point User Group (IFPUG) was set up, several versions of the Function Point Counting Practices Manual have been published by IFPUG.

Position Papers

Growth and Acceptance of Function Point Analysis

The acceptance of Function Point Analysis continues to grow. This is indicated by the growth of the International Funcion Point User Group (IFPUG). Since 1987 membership in IFPUG has grown from 100 members to nearly 600 members in 1994. Additionally, in less than six years conference attendance has grown from 125 in 1988 to 315 by 1994. Examination of IFPUG clearly indicates that the mejority of ist is members are from North America, but Function Point analysis growth outside North America is strong. This is evident by the growing numer of function point oranizations worldwide. There are eight affiliate oranizations of IFPUG listed below.

Affiliate	*Country/Region*	*Organization Name*
ASMA	Australia	Australian Software Metrics Association
CIM	Quebec	Centre d interet sur les metriques
DASMA	Germany	Deutschsprachige Anwendergruppe for Software Metrik und Aufwandschätzung
EFPUG	Europe	European Association of Function Point Users Group
FFPUG	France	French Function Point Users Group
GUFPI	Italy	Gruppo Utenti Function Point Italia
NEFPUG	Netherlands	Netherlands Function Point Users Group
UFPUG	United Kingdom	UK Function Point User Group

Using Italy as an example, in 1990 GUFPI was formed with 20 business partners and has grown to 50 at the present time. GUFPI organized the first Italian workshop on Software Metrics in 1993 with 120 people attending the meeting and representing 80 different organizations. Last November, the Second Italian workshop on Software Metrics was held with 150 attendants. The aim of these meetings is to promote the use of function points and provide experiences in reals world environments.

Additionally, GUFPI and IFPUG are organizating a joint conference on software metrics to be held in Rome in February 1996. This conference is expected to be the largest European conference on software metrics to date.

International Function Point User Group (IFPUG)

The primary mission of IFPUG is to promote and encourage the effective management of application software development and maintenance activities through the use of Function

Point Analysis and other software measurement techniques. IFPUG accomplishes this through conferences, pre conference workshops, committees and associated publications.

IFPUG has two conferences per year (Spring and Fall). The objective of the conferences is to facilitate the exchange of Knowledge and ideas for improved software measurement techniques. Additionally, the conferences provede an environment that stimulates the personal and professional development of attendees.

The Education Curriculum Committee (ECC) was formed at the beginning of the 1992 to develop, coordinate, and promote high-quality pre-conference workshops. The committee is responsible for selecting and developing courses, obtaining qualified instructors, ensuring the quality of all training materials, and coordinating with other committees additions and modifi-

cations to course material based on changes in counting practices, management reporting practices, etc.

Other IFPUG Committees

Counting Practices Committee (CPC)

The CPC works to develop a consistent standard for Albrecht-based function point counting practices. The committee identified a need in 1988 to develop and publish and IFPUG standard built on the foundations of Albrecht 1984. The result was the publication of the IFPUG Function Point Counting Practices Manua, curently in use under Release 4.0, released in 1994.

Management Reporting Committee (MRC)

This committee was established to priovide IFPUG members with a published set of quidelines for using Function Points and other software metrics in management reporting. MRC publishes and maintains the Function Ponts as an Asset. In 1991, the MRC began work on a Management Reporting Workshop curriculum targeted at understanding and using metrics for managing reporting software quality and productivity. This workshop was successfully piloted at the Charleston conference and remains part of IFPUG`s pre conferenc etraining curriculum.

Certification Committee (CC)

After the release of the Counting Practices Manual, the need of a professional certification program related to the newly defined IFPUG counting standard was identified. It was determined that professional certification was needed for counting practitioners. In 1991, the Certification Committee was formed to create and administer certification exams. A certification test for function point counters has been developed and is administrated prior to conterences.

New Environments Committee (NEC)

The committee was formed in 1989 to target the growing number of engineering-oriented members. The committee examines the application of Function Point Analysis object-oriented systems, CASE, grahics, and other new technologies.

Benchmarking Committee (BC)

The committees was formed in 1993 to develop a process to allow organizations (both public and private) to share metrics information. It has been expanded to include organization worldwide.

Objectives of Function Point Analysis

Function Points measure software by quantifying ist functionality provided to the user based primarily on the logical design. Frequently the term end user or user is used without specifying

what is meant. In this case, the user is a sophisticated user. Someone that would understand the system from a functional perspective --- more than likely someone that would provide requirements or does acceptance testing. In other words, Function Points make it possible to measure the size of the information systems according to what the user sees and interacts with.

Since Function Points measures systems from a functional perspective they are independent of technology. Regardless of Language, development method, or hardware platform used, the number of Function points for a system will remain constant. The only variable is the amount of effort needed to deliver a given set of function points; therefore, Function Point Analysis can be used to determine whether a tool, an environment, a language is more productive compared with others within an organization or among organizations. This is a critical point and one of the greatest values of Function Point Analysis.

Function Point Analysis can provide a mechanism to track and monitor scope creep. Funkction Point Counts at the end of requirements, analysis, design, code, testing and implementation can be compared. The function point count at the end of requirements and/or designs can be compard to function points actually delivered. If the project has grown, there has been scope creep. The amount of growth is an indication of how well requirements were gathered by and/or commicated to the project team. If the amont of growth to projects declines over time it is a natural assumption that communication with the user has improved.

Characteristic of Quality Function Point Analysis

Function Point Analysis should be performed by trained and experience personnel. If Function Point Analysis is conducted by untrained personnel, it is reasonable to assume the analysis will done incorrectly. The personnel counting function points should utilize the most current version of the Function Point Counting Practices Manual (at the moment version 4.0).

Current application documentation should be utilized to complete a function point count. For example, screeen formats, report layouts, listing of interface with other systems and between systems, logical and/or preliminary physical data models will all assist in Function Point Analysis.

The task of counting function points should be included as part of the overall project plan. That is, counting function points should be scheduled and planned. The first function point count should be developed to provide sizing used for estimating.

Summary of benefits of Function Point Analysis

- Function Points can be used to size software applications accurately. Sizing is an important component in determining productivity (outputs/inputs).
- They can be counted by different people, at different times, to obtain the same measure within a reasonable margin of error.
- Function Points are easily understood by the non technical user. This helps communicate sizing information to a user or customer.

Position Papers

- Function Points can be used to determine whether a tool, a language, an environment, is more productive when compard with others.

Conclusions

Accurately predicting the size of software has plagued the software industry for over 45 years. As indicated by the growth of IFPUG, Function Points are becoming widely excepted as the standard metric for measuring software size. Now that Function Points have made adequate sizing possible, it can now be anticipated that the overall rate of progress in software productivity and software quality will improve. Understanding software size is the key to understanding both productivity and quality. Without a reliable sizing metric relative changes in productivity (Function Points per Work Month) or relative changes in quality (Defects per Function Point) can not be calculated. If relative changes in productivity and quality can be calculated and plotted over time, then focus can be put upon an oranizations strengths and weaknesses. Most important, any attempt to correct weaknesses can be measured for effectiveness.

Due to space limitations this article could never be a comprehensive guide to counting function points. If the reader wants a comprehensive guide to counting function points, the authors recommend contacting the International Function Point User Group (IFPUG) to obtain the most current version of the Function Point Counting Practices Manual.

IFPUG Executive Offices
Blendonview Office Park
5008-28 Pine Creek Drive

Westerville, OH 43081-4899 USA
(614) 895-7130, Fax: (614) 895-3466

The authors can be reached at.

David Longstreet 2207 West WalnutBlue Springs, MO 64015 USA Voice: (816) 229-0303, Fax: (617) 273-5176 david@spr.com or at CompuServe 75557.437 Raffaela Ibba, SOGEI, S.p.A. via Mario Carucci, 9900143 Roma Italy Voice: +39-6-50252266, Fax: +39-6-50957201.

About the authors:

David Longstreet is a graduate of Texas A&M University. He has written articles on software metrics, maintenance, and testing. He ist the authors of Tutorial on Software Maintenance published by IEEE Computer Society Press. He is listed in Marquee's Who's Who in America. He has been a member of the Board of Directors of IFPUG since 1992. He is a consultant for Software Productivity Research.

Raffaela Ibba is a graduate of the University of Rome. She has written articles on software metrics, maintenance and testing. She is a project leader for a metrics group at Sogei, a company of the Finsiel Group, where she has been working since 1983. She is an officer of GUFPI holding the position of Treasurer. She is a certified function point counter.

Metrics Interest Groups

GI-Interest Group "Quality Improvement"

by Martin Schader
and Kathrin Baumann

Motivation

The use of software in all fields of every day life and the solution of increasingly complex and security- or reliability-related problems results in the importance of the ability to rely on the quality of applications. Furthermore, the change of markets requires a flexible reaction to new demands.

Goals

With the interest group "quality improvement" that was established at the second meeting of the GI-special group 2.1.9 for "object-oriented software engineering" in Berlin, a forum shall be created

- for the discussion of the importance of object orientation for software quality
- for an exchange of ideas and the discussion of current problems and research results
- for the transfer of results and experience between science and practice

- for cooperation among each other
- to impart the state of research
- to facilitate entry into these topics for interested persons

Topics

The interest group "quality improvement" is being assembled at the moment. The following list contains the topics, which have been of special interest during the meeting of the GI-special group 2.1.9 on March 25, 1996. Proposals for further topics are always welcome.

- The object-oriented software engineering process: analysis, judgement and improvement
- Importance of object-orientation for the quality of the development process and the developed products (especially the comparison between common and object-oriented software engineering as well as experiences with the transition)
- Guidelines for object-oriented design
- Procedure for continuous quality improvement (including documentation of experiences)
- Quality improvement by reuse
- Test and maintenance of object-oriented applications
- Prediction of object-oriented systems, estimation methods

Metrics Interest Groups

- Software measures (object-oriented product-, process- and resource-measures)
- Quality pattern (reliable quality-improving solutions of problems and strategies for the solution of problems)
- Quality-standards and their use in pracitce

Recent Information

If you are interested in the interest group "quality improvement" your name can be included in our list of interested persons. Just write an e-mail to

Majordomo@rz.uni-mannheim.de

with the following content

subscribe ak-ooqv <address>

or a note to Frau Baumann. So you will receive automatically recent information about the works of the interest group "quality improvement". Furthermore, you will get information in WWW at the following link:

http://www.bwl.uni-mannheim.de/Schader/beschreibung/ak-ooqv.html

Participation

A lively participation in the meeting is welcomed. The first meeting of the interest group will be integrated into the 6th Workshop of the *GI-AG Softwaremetriken*. The most important points in the near future will be phrased then to build up a basis for further co-operation. Interested persons who are willing to take an active part are requested to prepare a short lecture (10-15 minutes lecture plus discussion).

It would be very useful for the preparation, if you could tell us

- if you are interested in participating in the first meeting of the interest group and
- if you wish to give a lecture

(E-mail or a written note to Frau Baumann; possivly with the lectures' topic).

Further meetings of the interest group can be held either in the form of workshops concerned with the main topics or like a conference with different contributions which will cover a large spectrum. (This will be a point of discussion at the first meeting too.)

Contact and Address

Lehrstuhl für Wirtschaftsinformatik III
Prof. Dr. Martin Schader
Dipl.-Math. oec. Kathrin Baumann
L 5,6 Tel.: +49 (0621) 292-5075, -5375
D-68131 Mannheim, Schloß Fax: +49 (0621) 292-5701

E-Mail: baumann@wifo.uni-mannheim.de
www: http:/www.bwl.uni-mannheim.de/Schader/beschreibung/ak-ooqv.html

Metrics Interest Groups

Software Metrics Research Positions in Montreal

by Pierre Bourque

Software Engineering Management Research Laboratory
Universite du Quebec a Montreal

The mission of the Software Engineering Management Research Laboratory is to develop, for our software engineering industry, the analytical models and measurements instruments to enable them to improve their decision-making processes in order to meet their business objectives.

Director: Alain Abran, Ph.D.

Assistant Director:Pierre Bourque, M.Sc.

Project Partners:

- Universite du Quebec a Montreal
- Bell Canada
- Natural Sciences and Engineering Research Council of Canada

3 year project entitled VALID: 1996-1998

Project Research Issues:

- Measurement programs in software engineering
- Software maintenance measurement
- Function Point Analysis

The full job descriptions of these positions will conform to Universite du Quebec a Montreal standards on research employment.

Full-time employment for a minimum of 12 months, renewable in 1997 and in 1998.

Lead Researcher

Responsabilities include:

- Lead researcher on projects which are related to the Laboratory's research directions
- Research methodology support
    - ♦ Graduate student support on proper data collection methodologies for software metrics research.
    - ♦ Graduate student support on data analysis.
- Design and implementation of an architecture of an integrated software metrics database
    - ♦ Design and implementation of a database architecture for the data collected across the research projects.

    - ♦ Orderly evolution of this database architecture across projects and in accordance with the progress of the completed research.
- Writing of scientific papers
- Collaboration with our international research partners, the Laboratory's Research Associates, and the Laboratory's graduate students.

Requirements:

- Masters degree specialized in software engineering measurement techniques
- Industrial research experience
- Working knowledge of French
- Good English writing skills is preferred

Please submit to the address listed below:

- your resume
- examples of previous work

Post-doctoral research fellowship

Responsabilities include:

- Lead researcher on projects which are related to the Laboratory's research directions
- Software engineering research methodology
    - ♦ Graduate student support on proper research methodologies for software engineering research.
    - ♦ Graduate student support on proper data collection methodologies for software engineering research.
- Writing of scientific papers

Requirements:

- PhD. in software engineering, in computer science or in information systems;
- Industrial research experience
- Good English writing skills
- Workin knowledge of French is preferred

Please submit to the address listed below:

- your resume
- examples of previous work

Contact and Address

Pierre Bourque, Directeur adjoint, Laboratoire de recherche en gestion des logiciels
Departement d'informatique, Universite du Quebec a Montreal
Case postale 8888, succursale Centre-Ville, Montreal (Quebec) Canada
H3C 3P8, Tel.: (514) 987 - 0315, Fax: (514) 987 - 8477
e-mail: bourque.pierre@uqam.ca
www: http://saturne.info.uqam.ca/Labo_Recherche/lrgl.html

Metrics Interest Groups

more in the next journal about:

- ***ami (Application of Metrics in Industry)***
- ***QUALMS***
- ***STEV (Austria)***

- ***etc.***

New Books on Software Metrics

Ebert, C., Dumke, R.:Software Metriken in der Praxis: Einführung und Anwendung von Software Metriken in der industriellen Praxis

Springer Verlag, Berlin, 1996, 314 pages and 108 figures; in German language; hardcover; available from all bookstores world-wide. ISBN 3-540-60372-7.

Software metrics as indicators for cost, delivery accuracy or quality are important instruments for software project management. The recently published book "Software Metrics in Practice" of Ebert and Dumke provides theoretical and practical background of software metrics based on real-world projects. While the two editors alone already cover industry and academia and have more than 20 years of metrics experience altogether, the background of the other 14 authors is even more colourful. Their individual background covers almost all areas of software development and managment. Application domains range from electronics and telecommunication to car manufacturing, services and traditional IT providers. Sizes of thein companies are from small software and consulting companies up to big global players. Several of the authors companies are ranked in the Fortune 100. Several authors have directed international software metrics initiatives, such as AMI, BOOTSTRAP, COSMOS, or GQM.

The topics described in the book include fundamental measurement theory, metrics tools, benchmarking and project management. The application of the mentioned areas in the context of real projects and based on typical problems of industrial day-to-day business focuses on software development, effort estimation, criticality prediction, or reliability models.

Most interesting however is the fact that not only the authors have impressive industrial backgrounds but as a result of this, the different articles also provide examples form very different settings and environments. Case studies for instance describe how a metrics program is started, what to consider when planning an assessment, or how to select metric tools. Hands-on examples introduce software quality management, test tracking, quantitative process managment, project managment, or metrics training. State-of-the-art experiences finally include object-oriented metric suites and software process improvement.

Besides this overview of different areas and application domains of software metrics the book includes an annotated literature list with almost 50 entries. Current URLs to metric related world wide web sites conclude this boot that claims being an overview on introduction and application of software metric, but indeed provides insight into the wide world of metrics, without getting voring or putting too much weight on irrelevant topics. No critique without any negative flavour. What is surely missing in this book is more fundamentals on measurement theory and of course metrics definitions. Keeping in mind that the book was not intended as a catalogue on metrics, but rather an introduction and overview, we think that this lack is not a real problem. In fact several chapters provide lists of the most relevant metrics, that are rypicaly rather simple: effort, errors, or size are seemingly more often used than any sophisticated complexity metrics.

Tho book quotes Howard Rubin with stating that more than 60 % of all metric projects fail in a sense that they do not get beyond the piloting phase. Taking the book and remembering the good advice in the beginning, namely to first assess one's situation, then determine where to go, head for actions to get there and only afterwards decide on metrics that show how far the

targets have been achieved, should help avoiding all the mentioned pitfalls and mitigating the occurring risks. Immediate access to the authors via e-mail might help in getting answers to
problems that can otherwise not be resolved. By doing so, perhaps the book acts as a catalysator for yet another metric success story that could be added in a future edition.

Dumke, R.; Foltin, E.; Koeppe, R.; Winkler, A.: Softwarequalität durch Meßtools - Asessment, Messung und intrumentierte ISO 9000

Vieweg Publ., 1996, ISBN 3-528-005527-8

This book gives an overview about the software metrics tools for all phases of the software development process. The metrics tools are defined as CAME tools (*Computer Aissisted Software Measurement and Evaluation).* The introduction describes the essential aspects of the software measurement. The description of the CAME tools includes the cost estimation tools, Capabiblity Maturity Model evaluation tools, metrics tool for software specification,

design and code, and tools for the software testing and maintenance (including network performance). Some tables helps to decision of choosing the useful tool integration for the software quality assurance process.

Garmus, D.; Herron, D.: Measuring the Software Process - A Practical Guide to Functional Measurement

Yourdon Press Computing Series, 1996

Allan Albrecht has been known as the inventor of Function Points since the method was first introduced in 1979. The book describes the Function-Point method in detail. The authors write: We have had the good fortune to work with Allan over many years. *He has been gracious enough to share with us and many others his time, knowledge, and insights into this invaluable software measurement technique*.

van den Berg, K.: Software Measurement and Functional Measurement

Thesis University of Twente Enschede, ISBN 90-9008251-4

Software metrics have been investigated for the assessment of programs written in a functional programming language. The external attribute of programs considered in this thesis is their comprehensibility to novice programmers. This attribute has been operationalized in a number of experiments.

The internal attribute of software which is examined is the structure. Two models for the structure of software have been employed: callgraphs and flowgraphs. The proposed control-flow model captures the operational semantics of function definitions. The objective measurement of the attributes has been supported by tools.

The validation of structure metrics has been addressed in certain experiments for programming-in-the-small. The structure of type expressions in functional programs has been

analysed in a case study. A simple framework for software metrication proved to be useful. The validation of metrics has been linked with axioms from the representational measurement theory.

The control-flow model for functional programs showed ist value in the setup of an experiment regarding the influence of the structure on the comprehensibility. A programming style rule on the use of guards in function definitions has been validated by the findings in this experiment.

Takang, A.A.; Grubb, P.A.: Software Maintenance - Concepts and Practice

Thomson Computer Press, 1996

The purpose of this book is to explore the key conceptual and practical issues underpinning software change and discuss how they impact on the implementation of changes to software systems. The motivation for this book stems from the challenges that software engineers face when modifying complex software systems after the systems become operational. The extent of this challenge is indicated in the cost of accommodating evolving requirements by modifying software. This cost can be up to 70% of the total life-cycle cost. Software maintenance is increasingly being recognized as an important area in software engineering. Despite this, many mainstream software engineering courses in Universities concentrate on issues concerning the development of new software systems while only playing lip service to the concerns of effecting changes to these systems after they become operational.

Kuvaja, P.; Similä, J.; Krzcinik, L.; Bicego, A.; Saukkonen, S.; Koch, G.: The BOOTSTRAP Approach

Blackwell Business, 1994

The value of software process assessment and improvement has been reconized for some time. Initiatives such as the SEI's (Software Engineering Institute in the USA) or ISO 9001 have selectively addressed the needs of the software engineering community.

This book is about BOOTSTRAP, an assessment and improvement methodology that addresses perhaps the widest sample of software producing organisations referred to throughout this book as SPUs (Software Producing Units). It offers integrated methods and tools for assessment and improvement applicable to an extensive selection of SPUs.

This book addresses primarily BOOTSTRAP, but also emphasizes general aspects of software process assessment and improvement. Therefore, we hope it will be of practical use to a wide readership. There are no specific entry requirements for the reader - who may b - a software process specialist or a novice in the discipline.

New Books on Software Metrics

Melton, A. (Editor): Software Measurement

Thomson Computer Press, 1996

The idea for this book was conceived in the fall of 1991 as I was organizing a panel on software metrics for the 20th Annual Computer Science Conference. Given that there would be software metrics experts at the conference, it was deceided to organize a one-day workshop on software metrics. The worksop was entitled: Software Metrics: Understanding Software Engineering. This book is in a sense the conclusion of that workshop.

Software Productivity Consortium: The Software Measurement Guidebook

Thomson Computer Press, 1995

The Software Measurement Guidebook provides practical guidance for the measurement driven management of software development and maintenance projects; it applies to software intended for both government and nongovernment users. Guidance is provided about the collection and validation of process and product metrics data and in the use of quantitative information to support project control and process improvement. This guidebook describes how to select quantifiable goals for a software project and metrics to support those goals. It presents practical methods for estimating software size, development and support cost, and development schedule. Also presented are measure of software quality and models for estimating and predicting software defects.

Henderson-Sellers, B.: Object-Oriented Metrics - Measures of Complexity

Prentice Hall Publisher, 1996

Object-oriented metrics are an integral part of object technology and of good software engineering. Careful implementation of a metrics program can offer significant range of advantages:

- A quantitative understanding of both the architecture and the detailed design of the system so that successes can be built on and repeats of failures avoided. Without metrics, risk levels are significantly higher.
- A quantivative evaluation of the value of object technology, for example, in terms of productivity, effort, time, and maintainability of systems.
- An objective evaluation as to whether a particular disign / program has all the hallmarks of a good object-oriented design / program or whether it is "sheep dressed as lamb".
- The basis for good cost estimation of object-oriented projects.
- An integral part of a well thought out management strategy for object-oriented projects.

This book presents the state-of-the-art on object-oriented metrics.

New Books on Software Metrics

Wiklund, M.E. (Editor): Usability in Practice - How Companies Develop User-Friendly Products

AP Professional, 1994

Michael Wiklund directs the usability engineering group at American Institutes for Research, which delivers user interface education, research, design, and evaluation services to commercial and government clients. He has advised many companies on how to establish and

maintain a successful usability program. His design portfolio includes numerous products for medical, scientific, business, and home use. Since 1987, he has been a visiting industry Professor at Tufts University, where he teaches a project-oriented course-titled Applied Software User Interface Design.

Lehner, F.: Software-Dokumentation und Messung der Dokumentations-qualität

Die Software-Dokumentation - als Tätigkeit und als Ergebnis dieser Tätigkeit - ist seit einigen Jahren starken Veränderungen ausgesetzt. Die Veränderungen ergeben sich durch verbesserte Techniken und Werkzeuge zur Dokumentation, durch Entwicklungen in der Informatik, aber auch durch die allmähliche Aufnahme des Wissens aus anderen Disziplinen (z.B. Linguistik, Psychologie, Lernforschung). Die Gründe für Mängel bei der Dokumentation, für ihren unzureichenden Stellenwert und für die geringe wissenschaftliche Auseinandersetzung m it dem Thema sind vielfältig: die Tätigkeit der Software-Entwicklung wird höher eingeschätzt, Ausbildungsdefizit der Software-Entwickler, der interdisziplinäre Charakter der Dokumentation, Personalmangel usw. Ziel des Buchs ist es, vor diesem Hintergrund Hilfestellung und Orientierung bei den Aufgaben der Software-Dokumentation sowie bei der Beurteilung und Verbesserung der Dokumentationsqualität zu geben.

Shepperd, M.: Foundations of Software Measurement

Prentice Hall, 1995

This book has as ist aim the better integration of measurement into practice of software engineering. In order to accomplish this, the reader is provided with sufficient background to understand, develop and apply metrics to software engineering projects. A particular concern is the use of measurement within the context of models. It is strongly argued that modelling is a powerful aid for understanding what is to be measured, together with the meaning of the measurement, and that without some form of measurement can degenerate into an aimless collection of numbers.

New Books on Software Metrics

Will Appear:
Zuse, H.: A Framework of Software Measurement

The central goal of this textbook is to provide readers with a framework of software measurement. Software measurement is not a mature science, today. On the one side, there is a lack of a theoretical framework for software measurement, and on the other side, there is a

lack of education of scientists, practitioners and students in the area of software measurement. The book is written with the intention to investigate software measures and to give theoretical and practical guidelines for software measurement. By the time the reader reached the and of this book, scientists, teachers, practitioners, and students should be able to define the basic terminology of software measurement, to explain the definition of a measure; the definition of a homomorphism, the ideas / secrets behind software measures, the empirical conditions behind software measures, the scale types, the use of qualitative and quantitative methods, the role of software measures during the software life-cycle, the idea behind software cost estimation models, the idea behind the Function-Point method, the exact definitions of validation of software measures, the foundations of prediction models, the use of meaningful statistics, and the reader should be able to explain problems in the area of software measurement.

Metrics including Comferences

- **First Euromicro Working Conference on Software**

Maintenance and Reengineering, Berlin, March 17-19, 1997 (Deadline for Papers: September 15, 1996)

- **Fourth International Software Metrics Symposium,** March 1997, Boston (incl. with the ICSE'97)

- **3rd International Conference on Reliability, Quality & Safety of Software-Intensive Systems (ENCRESS'97),** May 29-30, 1997, Athens, Greece (Deadline for Papers: October 15, 1996)

- **Fifth International Symposium on Assessment of Software Tools and Tehnologies,** Pittsburgh, May 1997 (Deadline for Papers: December 1, 1996)

- **European Software Control and Metrics Conference** was continued after the Wilmslow (May 1996) Conference

- metrics themes are also discussed in the yearly **OOIS, ECOOP** and **ESEC** conferences

Metrics in the World-Wide Web

For further informations about software metrics related topics see:

- **http://www.mccabe.com** to the McCabe & Association,

- **http://www.esi.es** to the European Software Engineering Laboratory

- **http://www.cs.tu-berin.de/~zuse** to Dr. H. Zuse for the software metrics bibliography and other more

- **http://irb.cs.uni-magdeburg.de/se/metrics_eng.html** to the software measurement laboratory at the Univerisity of Magdeburg

- **http://www.sbu.ac.uk/~csse/publications/OOMetrics.html** to Prof. R. Whitty, London, at the object-oriented metrics papers

METRICS NEWS

VOLUME 1 **1996** **NUMBER 1**

CONTENTS

ISSN 1431-8008