

White Box Estimation – Theorie und Praxis

Cornelius Wille¹, Andreas Schmietendorf², Reiner Dumke³

¹Fachhochschule Bingen, wille@fh-bingen.de

²Fachhochschule für Wirtschaft und Recht Berlin, schmiete@fhw-berlin.de

³Otto-von-Guericke-Universität Magdeburg, dumke@ivs.cs.uni-mgdeburg.de

Zusammenfassung

Der folgende Beitrag erläutert zunächst einige Aspekte der Aufwandschätzung und die wesentlichen Merkmale der verbreitetsten Methoden.

Danach wird der Inhalt bzw. die Motivation für ein so genanntes White-Box-Estimation diskutiert. Ähnlich dem Web 2.0 als „Mach-mit-Web“ lebt das White Box Estimation von der aktiven Teilnahme möglichst vieler IT-Entwickler und Projektverantwortlicher.

Abschließend wird eine aktuelle Initiative der COSMIC und der ISBSG vorgestellt, die dazu aufruft, die internationale Datenbasis für Projekterfahrungen mit erweitern zu helfen und damit für weitere Anwendungen noch besser zu qualifizieren.

1. Aufwandschätzung

Die Aufwandschätzung im IT-Bereich ist nach wie vor eine der wichtigsten Aufgaben für die Gewährleistung einer Software-Entwicklung im Budget und in der Zeit. Die folgende Abbildung zeigt vereinfacht die allgemeine Form einer Schätzaufgabe als „Wer schätzt was womit für wen?“

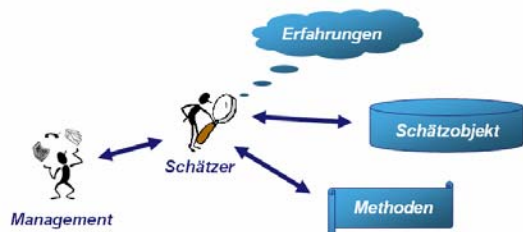


Abbildung 1: Allgemeine Schätzaufgabe

Beim „Wer“ geht es um die Qualifikation, die Erfahrung aber auch um die Motivation des Schätzers. Das „Was“ ist in diesem Umfeld zunächst der Produkt- bzw. Systemumfang und weiterhin die Entwicklungsdauer bis hin zum Entwicklungsaufwand bzw. den -kosten.

Beim „Womit“ geht es um die Schätzmethode selbst. Zu den allgemeinen Schätzmethode gehören beispielsweise ([Biffel 2000], [Boehm 2000], [Bundschuh 2008], [Jones 2007], [Laird 2006], [McConnell 2006], [Putnam 2003])

- *Algorithmische Modelle* (Constructive Cost Models (COCOMO); Cost-based Quality Model (COQUAMO), Software Lifecycle Management (SLIM) usw.)
- *Expertisen* (Delphi, Wideband-Delphi usw.)
- So genannte *Point-Metriken* (International Function Point User Group Function Point Analysis (IFPUG FPA), Mark II FPA usw.)
- *Analogieschlussmethoden* (Proportionalitätsverfahren, Pendant-Schätzung usw.)
- *Technologiebezogene Methoden* (Weighted Average of Individual Offsets (WOA), Constructive Cost Model for COTS (COCOTS), Object Points usw.)
- *Proprietäre Lösungen* (firmenspezifische Lösungen, firmenbezogene Adaption vorhandener (standardisierter) Methoden usw.)

Beim „Wen“ geht es schließlich darum, den eigentlichen Adressaten einer Schätzung, i. a. das Management, „zufrieden zu stellen“.

Eine – wenn auch klassische – Übersicht zu den so genannten *Point-Metriken* lautet nach [Lothar 2007] wie folgt.

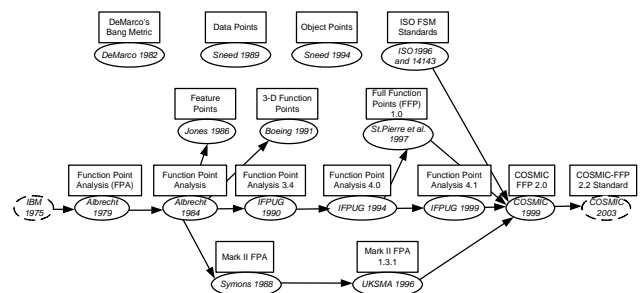


Abbildung 2: Point-Metriken im Überblick

Die Grundlage hierfür ist im allgemeinen der so genannte Function Size Measurement (FSM) Standard ISO/IEC 14143.

Neben diesen klassischen Formen existieren weitere spezifische Formen als Erweiterung der IFPUG-Methode in den Niederlanden (der NESMA), in Finnland (der FISMA) sowie als spezielle Form einer UML-basierten Software-Entwicklung als so genannte Use Case Points (UCP) [Frohnhoff 2008].

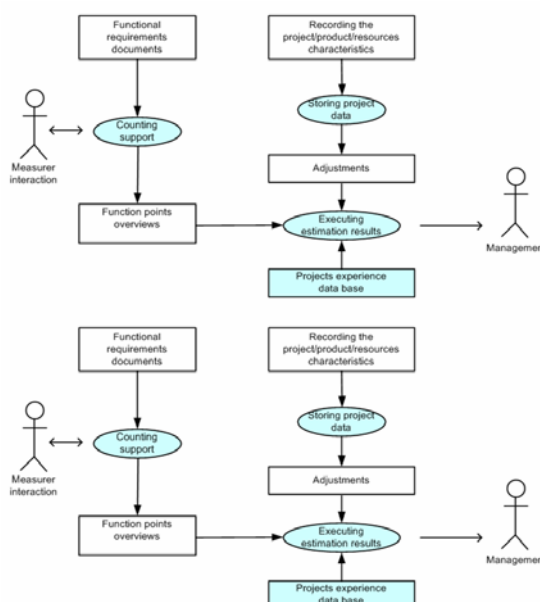
Im Bereich der Point-Metriken ist allerdings die so genannte *COSMIC Full Function Point* (COSMIC FFP) [COSMIC 2007] nicht nur die Methode, die auch speziell für *eingebettete Systeme* zugeschnitten ist, sondern auch die bisher einzige, die auf Grundlage der Messtheorie validiert für die Funktionsumfangbestimmung eine *Verhältnisskalierung* aufweist mit der Maßeinheit CFP [Zuse 2003].

Das *Common Software Measurement International Consortium* (COSMIC) wurde 1998 gegründet und beschäftigt sich neben der Unterstützung der Einführung von Messprozessen in der IT-Industrie vor allem um die Anwendung und Weiterentwicklung der COSMIC FFP Methode als ISO 19761 Standard. Die inzwischen vorliegende Version 3.0 ist seit Dezember 2007 im Web verfügbar (siehe <http://www.cosmicon.com>) und enthält neben dem Manual auch Beispiele für den Business-Systembereich oder speziellen Formen der Aufwandschätzung.

2. White Box Estimation

Der Begriff des White Box Estimation wurde erstmals von Abran definiert [Abran 2007]. Anlass war die Kritik an vorhandenen, zumeist die IFPUG FPA unterstützenden Tools hinsichtlich der Aufwandschätzung auf der Grundlage „eingearbeiteter“ Projekterfahrungsdaten von „unzähligen“ Projekten.

Die Point-Metrikbasierte Aufwandschätzung wird allgemein nach folgendem Grundprinzip vorgenommen (siehe Abbildung 3).



lage so genannter *Kostentreiber* vorgenommen wird, die sich auf Projekterfahrungen beziehen bzw. diese bei jeder Schätzung erweitern.

Generell gelten die in der folgenden Abbildung genannten Regeln für Aufwandschätzverfahren (siehe auch die 118 Tips in [McConnel 2006]). Sie sollten

- in frühen Phasen anwendbar sein,
- Erfahrungen bereits realisierter Projekte nutzen,
- transparent für den Auftraggeber sein,
- unterschiedliche Aufwandsarten berücksichtigen,
- noch nach Jahren vergleichbar sein,
- usw.

Die Qualität einer umfangsbasierten Schätzung wird vor allem von der Qualität der projekterfahrungsbezogenen Merkmale bestimmt, d. h. in wie weit sie dem zu schätzenden Projekt entsprechen. Die Idee eines White Box Estimation besteht nun darin, diesen Erfahrungshintergrund „sichtbar“ zu machen, das heißt die Projektbeschreibungen für alle nachnutzbar explizit zu erfassen, um die Auswahl geeigneter Vergleichswerte für eine Schätzung zu qualifizieren. Natürlich ist die Erfahrung eigener Projekte im eigenen Technologie- und Systemumfeld durch Nichts zu ersetzen. Allerdings bedeutet dies im Zuge immer globaler, verteilter Entwicklungen auch eine erschwerte (internationale) Vergleichbarkeit. Wir wollen diese Einschätzungen wie folgt begrifflich zusammenfassen.

Black Box Estimation:

Anwendung eines Schätzverfahrens für die Bestimmung des System- oder Software-Produktumfangs und Durchführung der Schätzung auf der Grundlage umfangreicher Projekterfahrung, die jedoch nicht näher spezifiziert ist bzw. identifiziert werden kann. Allerdings ist hierbei auch der Vorteil internationaler Vergleichbarkeit gegeben im Rahmen der Skaleneigenschaft der jeweiligen Methode (siehe [McConnel 2006] oder [Bundschuh 2008]). Für die IFPUG FPA wird hierbei eine Vergleichbarkeit über gleichartige Systementwicklungen gewährleistet.

Gray Box Estimation:

Die Anwendung einer existierenden Methode bzw. eines Standards ohne verhältnisskalierter Umfangsbestimmung (also bereits Aufwandschätzung von Anfang an) unter Zugrundelegung eigener Projekt-

erfahrungen und der damit möglichen nahezu vollständigen Vergleichbarkeit im eigenen Entwicklungsumfeld (siehe z. B. [Frohnhoff 2008] für die UCP-Methode).

White Box Estimation:

Anwendung eines Schätzverfahrens für die Bestimmung des System- oder Software-Produktumfangs und Durchführung der Schätzung auf der Grundlage umfangreicher Projekterfahrung, die nach allgemeinen Kriterien ausgewählt und somit näher bestimmt werden bzw. auf die eigene Projektsituation zugeschnitten werden kann. Auch hier ist der Vorteil der internationalen Vergleichbarkeit i. a. gegeben (siehe [COSMIC 2007] oder [Ebert 2007]). Dabei besteht die Vergleichbarkeit vor allem auch über unterschiedliche Systemarten, was bei zunehmend heterogenen Architekturen immer notwendiger wird.

Glas Box Estimation:

Anwendung einer *eigenen Methode* für die Bestimmung des System- oder Software-Produktumfangs und Durchführung der Schätzung auf der Grundlage umfangreicher Projekterfahrung, die ausschließlich aus den eigenen Projekten entnommen wird und somit eine optimale Situationserfassung gewährleistet. Allerdings ist hierbei eine (internationale) Vergleichbarkeit nicht gegeben und somit die Möglichkeit der Nutzung anderer Erfahrungen nicht möglich ist.

Unabhängig von der jeweiligen Schätzmethode spielen die „geeigneten“ Projekterfahrungen also eine sehr wichtige Rolle.

3. Die COSMIC-ISBSG-Initiative

Von der *International Software Benchmarking Standards Group (ISBSG)* werden bereits seit über 10 Jahren Projektdaten auf freiwilliger Basis erfasst, die anderen helfen sollen, Vergleichsschätzungen projektspezifisch vornehmen zu können. Inzwischen sind Projektdaten von über 4000 Projekten enthalten (sog. Release 10). Die folgende Abbildung zeigt eine Auswertungsform dieses Erfahrungsdatenbasis (siehe [Ebert 2007]).

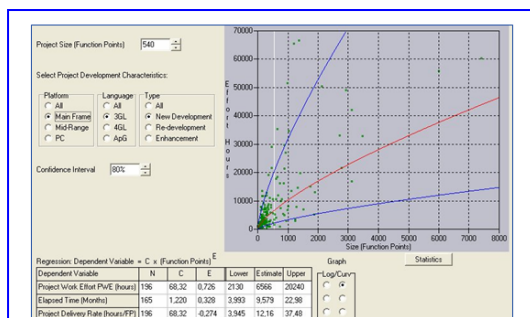


Abbildung 4: ISBSG Portal

Die diesjährige Initiative der COSMIC und der ISBSG soll nun dazu aufrufen, diese internationale Erfahrungsdatenbasis in besonderer Form zu erweitern und damit für alle noch besser (auch im Sinne eines White Box Estimation) anwendbar zu gestalten. Deshalb sollten sich Interessenten an den Web-Link

<http://isbsg.org>

bzw. Information direkt beim „admin@isbsg.org“ abfragen.

Die folgenden Informationen sollen einen Eindruck zu den zu erfassenden Daten geben, die im Repository gespeichert werden.

A. Submitter Information

- Contact information for the questionnaire submitter
Contact person:
Organisation:
Country:
E-mail:
- Your identifying name or ID for this submitted project.
Project ID:
Date Submitted:
- ISBSG Office Use Only
ID:
Date Rec.:
Initials:

B. Process

- What type of software project was your project?
New Development
Re-development
Enhancement
Other (specify):
- Is the development team involved in a process improvement program?
Yes No
Which if any, software process or quality standards was the project performed under?

C. Technology

- What was the primary technology used to build or enhance the software? i.e. that used for most of the build effort.
Programming Language
Operating System
Integrated Development Environment
Debugging
Database
Object/Component Server
HTML/Web Server
E-Mail or Message Server
Other (specify):

62. What was the primary implementation platform of the software product? i.e. that which the software was implemented into.

- Device Embedded
- Personal Computer
- Mid range
- Mainframe
- Multi Platform

If 'device embedded', please specify the target environment:

- Automotive
- Aviation
- Domestic appliance
- Games device
- Machine tool
- Mobile phone
- PDA Music device Telecommunications
- Other (specify):

D. People and Work Effort

DEVELOPMENT TEAM

63. In which country did the development team perform most of the project work?

Other(s):

In which country was the project implemented?

Other(s):

69. Development team effort (in hours) expended in each major activity of the 'generic' project process, and the number of team members involved in each activity. (Please provide summary values at least.)

Enter numbers of people & their effort for each activity (as Dev. Team People and Totals Effort hours for)

- Plan
- Specify
- Design
- Build
- Test
- Impl.
- Or summary values for the whole project

CUSTOMERS / FUNCTIONAL USERS / END USERS

70. In which industry is the software used (or do the software's end users primarily work)?

- Aerospace/Automotive
- Agriculture, Forestry etc.
- Banking
- Chemicals
- Communications
- Community Services
- Computers & Software
- Construction
- Consumer Goods
- Defence
- Education
- Institution Electricity,
- Gas & Water
- Electronics
- Food Processing
- Finance & Business Service
- Government
- Insurance
- Manufacturing
- Media
- Medical/Health Care

WORK EFFORT VALIDATION

80. Do the effort figures of Q's 69 and 74 account for all the work done on the project?

- Yes (skip the next question)
- No

81. If no (prev. question), what do you estimate the uncollected effort to be?

- Less than 5% of recorded
- 5 – 10% of recorded effort
- Other (specify): Unable to estimate

84. How would you rate the quality of the work effort data?

- Poor
- Adequate
- Good
- Excellent

85. Why did you assign the above quality rating?

E. Product

86. What type of software has the project produced, or is enhancing?

- 3D modelling or animation
- Catalogue or register of things or events
- Customer billing or relationship management
- Document management
- Device or interface driver Electronic data interchange
- Financial transaction processing & accounting
- Geographic or spatial information systems
- Graphics & publishing tools or system
- Image, video or sound processing
- Software for machine control
- Job, case, incident or project management
- Logistic or supply planning Management or performance
- Mathematical modelling
- & control reporting (finance or engineering)
- Network management Online analysis and
- Operating system or reporting software utility
- Personal productivity (e.g. word processor, spreadsheet)
- Process control Software development tool
- Stock control & order Trading
- Workflow support & processing management
- Other (specify):

91. If there was reuse of software development work products on this project, what was the amount of functionality provided by reused work products (measured or approximated)?

- Size:
- Unit/Method:
- Other Method:

F. COSMIC Functional Size

92/97. Which COSMIC functional sizing standard was applied?

Version: Specific local customisation? Yes No

Layer/Component (Item) Name Description

- 1.
- 2.
- 3.

DEVELOPMENT/REDEVELOPMENT SOFTWARE SIZE

96. Size Information

Item

Functional processes count (created)

Total size of ENTRIES

Total size of EXITS

Total size of READS

Total size of WRITES

Layer/Comp. total size

Total development/redevelopment size in CFP (or Cfsu)

ENHANCEMENT SOFTWARE SIZE

102. Added Functionality – Size Information

Item

Functional processes count (created)

Total size of ENTRIES

Total size of EXITS

Total size of READS

Total size of WRITES

Layer/Comp. total size

Total added size in CFP (or Cfsu)

103. Changed Functionality – Size Information

Item Functional processes count (modified)

Item

Functional processes count (created)

Total size of ENTRIES

Total size of EXITS

Total size of READS

Total size of WRITES

Layer/Comp. total size

Total modified size in CFP (or Cfsu)

104. Deleted Functionality – Size Information

Item

Functional processes count (created)

Total size of ENTRIES

Total size of EXITS

Total size of READS

Total size of WRITES

Layer/Comp. total size

Total deleted size in CFP (or Cfsu)

107. Total size of the Enhancement = Totals of added + modified + deleted sizes = CFP (or Cfsu)

108. After which of the following activities was this measurement performed?

- Planning
- Specification
- Design
- Build
- Test
- Implementation

111. Does the functional size entered in this section F (questions 92-107) match the functionality that was developed by the project effort entered in section D (questions 63-85)?

Yes No

If 'No' describe any additional functionality that was:

developed by the project:

delivered, but not developed, by the project:

Additional functionality may occur in software components or layers that were not addressed by the functional size measurement entered in 'Software size', for example the development of device drivers.

Additional functionality may be delivered but not developed, for example purchased software.

G. Project Completion

122. On what date did the software go into operation? (e.g. 05-Apr-09) (dd-mmm-yy)

123. What was the total project elapsed duration (including inactivity)? (months)

124. If there was any time of total inactivity, what was its duration? (months)

125. Are there any factors that you think had a **positive** impact on the project performance or outcomes?

126. Are there any factors that you think had a **negative** impact on the project performance or outcomes?

127. What was the number of defects recorded during the first month of the software's operation (first 30-days after the date on which the software began operation)?

Minor:

Major:

Extreme:

Or Total Defects:

4. Literatur

[Abran 2007] Abran, A.; Ndiaye, I, Bourque, P.: *Evaluation of a Black-box Estimation Tool: A Case Study*. Software Process Improvement and Practice, 12(2007), pp. 199-218

[Biffel 2000] Biffel, S.: *Using Inspection Data for Defect Estimation*. IEEE Software, Nov./Dec. 2000, pp. 36-43

- [Boehm 2000] Boehm, B. W.: *Software Cost Estimation with COCOMO II*. Prentice Hall, 2000
- [Bundschuh 2008] Bundschuh M.; Dekkers. C.: *The IT Measurement Compendium*, Springer-Verlag, 2008
- [COSMIC 2007] COSMIC-FFP: *Measurement Manual – The COSMIC Implementation Guide For ISO/IEC 19761:2003. v 3.0*, <http://www.cosmicon.com>, 2007
- [Ebert 2007] Ebert/Dumke: *Software Measurement – Establish, Extract, Evaluate, Execute*. Springer-Verlag, 2007
- [Frohnhoff 2008] Frohnhoff, S.; Engeroff, T.: *Field Study: Influence of Different Specification Formats on the Use Case Point Method*. In: Dumke et al.: *Software Process and Product Measurement*, Springer-Verlag, 2008, S 62-75
- [ISBSG 2003] *Software Project Estimation – A Workbook for Macro-Estimation of Software Development Effort and Duration*. Melbourne, 2003
- [Jones 2007] Jones, C.: *Estimating and Measuring Software Costs: Bringing Realism to Estimating*. McGraw-Hill Verlag, 2007
- [Kitchenham 1997] Kitchenham et al.: *Evaluation and assessment in software engineering*. Information and Software Technology, 39(1997), pp. 731-734
- [Laird 2006] Laird, L. M; Brennan, M. C.: *Software Measurement and Estimation: A Practical Approach*. John Wiley & Sons Publ., 2006
- [Lothar 2007] Mathias Lothar: *From Software Measurement to e-Measurement*. Shaker-Verlag, 2007
- [McConnell 2006] McConnell, S.: *Software Estimation*. Microsoft ubl., 2006
- [Putnam 2003] Putnam, L. H.; Myers, W.: *Five Core Metrics – The Intelligence Behind Successful Software Management*. Dorset House Publishing, New York, 2003
- [Zuse 2003] Zuse, H.: *What can Practitioners learn from Measurement Theory*. In Dumke et al.: *Investigations in Software Measurement*, Proc. of the IWSM 2003, Montreal, September 2003, pp. 175-176