

Entwicklungslinien des Software-Engineering

WS 2005/2006



Themen, betreut von
Univ.-Prof. em. Dr. H.-J. Hoffmann

(Stand 17. Oktober 2005)

Die Literaturangaben in den einzelnen Themenblättern dienen als Anleitung zur jeweiligen Thematik. Die Bearbeiter müssen sich um einen darüber hinausgehenden Überblick bemühen.

Angaben noch unvollständig !

Alle Internet-Zugriffe August – Oktober 2005 !
Irrtum und Tippfehler vorbehalten !

Anleitung zur Vorbereitung Ihres Vortrags bzw. der Ausarbeitung:

- S.I.P. Jones et al.: *How to give a good research talk*,
ACM SIGPlan Notices 28 (1993) 11, 9 - 12
- M. Deininger et al.: *Studien-Arbeiten, ein Leitfaden ...*;
(u.a.) Teubner, 1992

**Beides kann in der Bibliothek des FB Informatik
eingesehen werden !**

Etwas zur geschichtlichen Entwicklung der Informatik



Daten und Begebenheiten:

Übersicht

<http://www.cis.ksu.edu/~howell/492s04/history-impact.doc>

(leider einige Schreibfehler, insbesondere bei Namen, und an einigen Stellen einseitig aus amerikanischer Sicht)

Mit interessanten Illustrationen

http://www.computer.org/portal/cms_docs_computer/computer/timeline/timeline.pdf

Personen in Deutschland

<http://www.fbi.fh-darmstadt.de/~schneider/ba-ig/historie.pdf>

Aktivität der Gesellschaft für Informatik:

<http://www.gi-ev.de/themen/geschichte-der-informatik/>

Ein ausführlicher Tagungsbericht

<http://www.dagstuhl.de/DATA/Reports/9635/report.9635.html>



Vier Folgen

- Personen
 - #01 Zuse
 - #02 Wirth
 - #03 Parnas(weitere Personen unter anderen Themen)
 - Software-Engineering
 - #04 Der Beginn, Konferenzen in Garmisch und Rom
 - #05 *goto statement considered harmful*
 - #06 Verweise (*pointer*) und Parameter
 - #07 Fehlerbehandlung / Validierung
 - #08 *mythical man month*
 - #09 Unix / Linux
 - #10 *byte code*
 - Programmiersprachen
 - #11 Algol, ... , Ada
 - #12 Pascal / Modula / ... / C#
 - #13 Simula / Smalltalk
 - sonstiges
 - #14 Semaphore
 - #15 Petri-Netze
 - #16 Informatik in D
 - #17 Informatik in DA
- Mindestteilnehmerzahl 10.
Es werden **maximal 14 Themen** ausgegeben ! Reihenfolge, Lücken !

Konrad Zuse, 1910 – 1995



- Konrad Zuse: *Mein Lebenswerk* ; Springer, 1984
- J. Alex: *Wege und Irrwege des Konrad Zuse*; Spektrum der Wissenschaft, Januar 1997, 78 - 90
- K. Zuse: *Der Plankalkül* ; Gesellschaft für Mathematik und Datenverarbeitung, Bonn, BMBW – GMD -63, 1972
- F.L. Bauer, H. Wössner: *The "Plankalkül" of Konrad Zuse: A Forerunner of Today's Programming Languages* ; Comm. ACM 15 (1972) 7, 678 - 685
- J. Hohmann: *Der Plankalkül im Vergleich mit algorithmischen Sprachen*; Diss. TH Darmstadt, 1975
- W.K. Giloi: *Konrad Zuse's Plankalkül: The First High-Level "non von Neumann" Programming Language*; IEEE Annals History of Computing 19 (1997) 2, 17 - 24

Niklaus Wirth, 1934



- G. Pomberger et al.: *Niklaus Wirth – a Pioneer of Computer Science*;
<http://www.swe.uni-linz.ac.at/publications/pdf/TR-SE-00.08.pdf>
- N. Wirth: *Systematic programming*;
Prentice Hall, 1973
- N. Wirth: *Gedanken zur Software-Explosion*; Informatik-Spektrum, 17 (1994) 1, 5 - 10
- N. Wirth: *Ein Plädoyer für Zuverlässigkeit anstelle von Innovationitis*; Abschiedsvorlesung, NZZ 29. Jan. 1999



Folgeseite !

Niklaus Wirth, 1934 (1. Forts.)



- N. Wirth: *On the design of programming languages*; Proc. IFIP-Congress, 1974, 386 – 393 (Turing lecture !)
- K. Jensen, N. Wirth: *Pascal – User Manual and Report*; Springer 1974
- N. Wirth: *On the composition of well-structured programs*; ACM Computing Surveys 6 (1974) 4, 247 - 259
- N. Wirth: *Type extensions*; ACM TOPLAS 10 (1988) 2, 204 - 214
- M. Reiser, N. Wirth: *Programming in Oberon, steps beyond Pascal and Modula-2*; Addison-Wesley, 1992
- N. Wirth: *Pascal and its successors* .
In: M. Broy, E. Denert: *Software Pioneers*; Springer, 2002, 108 – 119

Niklaus Wirth, 1934 (2. Forts.)



- N. Wirth: *Program development by step-wise refinement*; Comm. ACM, 14 (1971) 4, 221 – 227
- N. Wirth: *Designing a system from scratch*; Structured Programming 10 (1989) 1, 10 – 18
- N. Wirth: *From programming language design to computer construction*; Comm. ACM 28 (1985) 2, 159 – 164
- Arbeitsplatzrechner *Lilith* (1978 – 1892)
- Arbeitsplatzrechner *Ceres* (1984 – 1986)
- Arbeitsplatzrechner *Ceres 2* (1988)
- Arbeitsplatzrechner *Ceres 3* (1989 - 1990)
- Experimentalrechner *Chamäleon* (1990)

siehe NZZ
Fernaussgabe
27./28. Febr.
1994,
Seite 27

David Parnas, 1941



- D.L. Parnas: *On the criteria to be used in decomposing systems into modules*, Comm. ACM 15 (1972) 12, 1053 – 1058 bzw. www.acm.org/classics/may96/
- D.L. Parnas: *On a 'Buzzword': Hierarchical Structure*, Proc. IFIP Congress 74, North-Holland, 1974, pp. 336-339
- D.L. Parnas: *Software Engineering or Methods for the multi-person construction of multi-version programs*. In G. Goos, J. Hartmanis (eds.): *Programming Methodology; LNCS 23*, Springer, 1975, 1-11
- D.L. Parnas et al.: *Precise Documentation of Well-Structured Programs*, IEEE Transact. Software Engineering, 20 (1994) 12, 948 - 976
- S.D.L. Hester et al.: *Using Documentation as a Software Design Medium*, Bell System Techn. Journal, 60 (1981) 8, 1941 - 1977

David Parnas, 1941 (1. Forts.)



Folgeseite !

- D.L. Parnas: *Why Software Jewels are Rare*; IEEE Computer, 29 (1996) 2, 57 – 60
- D.L. Parnas: *Software Engineering: An Unconsummated Marriage*; Comm. ACM, 40 (1997) 9, 128
- D.L. Parnas: *The Secret History of Information Hiding*. In M. Broy, E. Denert (eds): *Software Pioneers: Contributions to Software Engineering*; Springer, 2002, 399 – 409
- D.L. Parnas, M. Lawford: *The Role of Inspection in Software Quality Assurance*; IEEE Trans. Software Engineering, 29 (2003) 8, 16 - 20 & 674 – 676
- R. Baber et al.: *Disciplined Methods of Software Specifications: A Case Study*; Proc. Intl. Conf. Information Technology Coding and Computing, IEEE CS, 2005

siehe auch

<http://www.cas.mcmaster.ca/sqrl/DLP.publications.html>

David Parnas, 1941 (2. Forts.)



- D.L. Parnas: *Education for Computing Professionals*; *IEEE Computer*, 23 (1990) 1, 17 - 22
- D.L. Parnas: *Computing Science vs. Engineering Education*; *IEEE EDUCATIONews* 3 (1990) 1, 3 – 5
- D.L. Parnas: *Why engineers should not use Artificial Intelligence*; *INFOR* 26 (1988) 4, 234 – 24
- D.L. Parnas: *Professional Responsibilities of Software Engineers*; *Proc. IFIP 94*, Vol. II, 1994, 332 - 339

Der Beginn, Konferenzen in Garmisch und Rom



- P. Naur, B. Randell: *Software Engineering* ; Report on a Conference ... Garmisch, 7th to 11th October 1968
- J.N. Buxton, B. Randell: *Software Engineering Techniques* ; Report on a Conference ... Rome, 27th to 21st October 1969

für die beiden Berichte siehe:

<http://homepages.cs.ncl.ac.uk/brian.randell/NATO/index.html>

- B. Randell: *The 1968/69 NATO Software Engineering Reports* ; 1991
<http://homepages.cs.ncl.ac.uk/brian.randell/NATO/NATOREports/>

Und was daraus wurde:

- F.P. Brooks: *No Silver Bullet: Essence and Accidents of Software Engineering*; IEEE Computer, 20 (1987) 4, 10 - 19

"goto statement considered harmful"



- E.W. Dijkstra: Letter to the editor: *Go To Statement Considered Harmful*; Comm. ACM 11 (1968) 3, 147 - 148
- (http://portal.acm.org/ft_gateway.cfm?id=636521&type=pdf
http://www.thocp.net/biographies/papers/goto_considered_harmful.htm)
- David R. Tribble: *Go To Statement Considered Harmful: A Retrospective*; Febr. 2005, <http://david.tribble.com/text/goto.html>
- C. Böhm, G. Jacopini: *Flow diagrams, Turing machines and languages with only two formation rules*; Comm. ACM 9 (1966) 5, 366-371
- I. Nassi, B. Shneiderman: *Flowchart techniques for structured programming*; ACM SIGPLAN 8 (1973) 8, 12 – 26
(siehe auch [DIN 66261](#))

Verweise (*pointer*) und Parameter



- E. Crank: *Parameter-passing and the Lambda calculus*, 18th ACM SIGPLAN-SIGACT Symp. Principles of Programming Languages, 1991, 233 – 244, <http://portal.acm.org/citation.cfm?id=99616>
- U. Kastens: *Funktionen, Parameterübergabe*, 2005, ag-kastens.upb.de/lehre/material/gps/folien/gdp601-608.4.pdf
- R.W. Sebasta: *Concepts of programming languages*, Addison-Wesley, 2004, Chapter 9
- J. Mitchell: *The Algol family and ML*, 2003 (ML hier uninteressant), www.stanford.edu/class/cs242/slides/ml.ppt
- M.D. Crawford: *Pointers, References and Values*, 2005, <http://www.goingware.com/tips/parameters/>
- B. Goetz: *Java theory and practice: A brief history of garbage collection*, IBM, 2003, <http://www-128.ibm.com/developerworks/library/j-jtp10283/>

Folgeseite !

Verweise (*pointer*) und Parameter (Forts.)



- NN: *Value (computer science)*; 2005,
[http://encyclopedia.thefreedictionary.com/
value%20\(computer%20science\)](http://encyclopedia.thefreedictionary.com/value%20(computer%20science))
- NN: *Reference (computer science)*; 2005,
[http://encyclopedia.thefreedictionary.com/
Reference+\(computer+science\)](http://encyclopedia.thefreedictionary.com/Reference+(computer+science))
- NN: *Pointer*; 2005,
<http://encyclopedia.thefreedictionary.com/pointer>
- NN: *Smart pointer*; 2005,
<http://encyclopedia.thefreedictionary.com/Smart+pointer>
- NN: *Pointer arithmetic*; 2005,
[http://encyclopedia.thefreedictionary.com/
Pointer+arithmetic](http://encyclopedia.thefreedictionary.com/Pointer+arithmetic)
- J. Skeet: *Parameter passing in Java - by reference or by value?*;
2005, www.yoda.arachsys.com/java/passing.html
- J. Skeet: *Parameter passing in C#*; 2005
www.yoda.arachsys.com/csharp/parameters.html



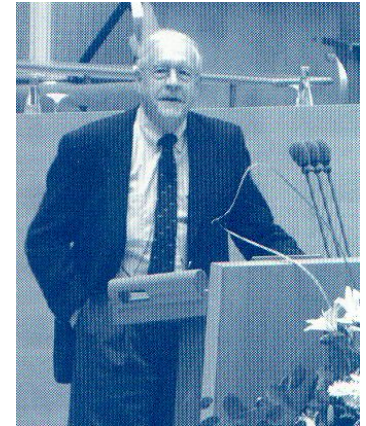
- Foto des ersten "bug", siehe <http://www.hopper.navy.mil/grace/grace.htm>
- R.W. Floyd: *Assigning meanings to programs*. In J. T. Schwartz (ed.): *Mathematical Aspects of Computer Science*; Proc. Symp. Applied Mathematics, vol. 19 (1967) 19 - 32
- C.A.R. Hoare: *An axiomatic basis for computer programming*; Comm. ACM, 12 (1969) 10, 576 – 580, 583
- C.A.R. Hoare: *Proof of a program: FIND*; Comm. ACM, 14 (1971) 1,39 – 45
- C.A.R. Hoare, N. Wirth: *An axiomatic definition of the programming language Pascal*; Acta Informatica 2 (1973) ??, 335- 355
- E.W. Dijkstra: *A discipline of programming*; Prentice Hall, 1997
- R.L. Baber: *Praktische Anwendbarkeit mathematisch rigoroser Methoden zum Sicherstellen der Programmkorrektheit*; Dissertation TU Darmstadt, 1995

Folgeseite !



- D.L. Parnas: *Can We Tolerate Software Errors?*; Proc. IFIP Congress 89, North Holland, 1989, 502
- P.J. Brown: *Error messages, the neglected area of the man/machine interface*; Comm. ACM 26 (1983) 4, 246 – 249
- J.C. Spohrer et al.: *Where the bugs are*; Proc. CHI 1985, 1985, 47 - 53
- J. Nielsen: *Error message guidelines*; Alertbox, 24. June 2001, <http://www.useit.com/alertbox/20010624.html>
- J. Siedersleben: *Errors and Exceptions – Rights and Responsibilities*, 2003, http://www.sdm.de/web4archiv/objects/download/pdf/vonline_siedersleben_ecoop03.pdf
- NN: *History of 404*, 2005, <http://www.plinko.net/404/history.asp>
- J.K. Mullin, D.J. Margoliash: *A tale of three spelling checkers*; Software – Practice & Experience, 20 (1990) 6, 625 - 630
- C. Wetherell: *Why automatic error correctors fail*; Computer Languages 2 (1977) 14, 179 - 186

"mythical man month"



- F.P. Brooks: *The Mythical Man-Month*; Addison-Wesley, 1975
- F.P. Brooks: *The Mythical Man-Month: Essays on Software Engineering*; 20th Anniversary Edition; Addison-Wesley, 1995
- F.P. Brooks: *The Mythical Man-Month after 20 Years*; IEEE Software 12 (1995) 5, 57 - 60
- R. Duncan: *The Mythical Man-Month*, a Book Review; 1996
<http://www.ercb.com/feature/feature.0001.html>
- J.M. Verner et al: *In the 25 years since The Mythical Man-Month was published what have we learned about project management?*; Information & Software Technology 41 (1999), 1021 - 1026
- NN: *The Mythical Man-Month*, a Book Review; (undated)
<http://www.robelle.com/library/smugbook/manmonth.html>
- NN: http://en.wikipedia.org/wiki/The_Mythical_Man-Month

HJH#09a
2005/2006



D. Ritchie

Unix / Linux



K. Thompson



- E. Levenez: Unix History / Unix Timeline; 2005, <http://ww.levenez.com/unix/history.html>
- D.M.Ritchie: *Writings from the Past*; <http://cm.bell-labs.com/cm/cs/who/dmr/notes.html>
- B.W. Kernighan, P.J. Plauger: *Software Tools*; Addison-Wesley, 1976 (siehe auch <http://www.amazon.com/exec/obidos/tg/detail/-/020103669X/102-4873663-3184931?v=glance>)
- B.W. Kernighan: *The Unix System and Software Reusability*; IEEE Trans. Software Engineering, 10 (1984) 5, 513 – 518
- M. Loukides: *A Brief History of UNIX*; (vermutlich) 2003 <http://www.uwsg.iu.edu/usail/external/recommended/unixhx.html>

Folgeseite !

Unix / Linux (Forts.)



Linus Torvalds

- L. Torvalds: *Linux History*; 1991/92,
<http://www.li.org/linuxhistory.php>
- NN: *History of UNIX / Linux and other variants*;
<http://www.computerhope.com/history/unix.htm>
- L. Wirzenius: *Linux: the big picture*; 2003,
<http://liw.iki.fi/liw/texts/linux-the-big-picture.html>
- Q.P. Coldiron: *Replacing Windows NT Server with Linux*; 1997,
<http://citnews.unl.edu/linux/LinuxPresentation.html>
- D. Kegel: *NT vs. Linux Server Benchmark Comparisons*; 2003,
<http://www.kegel.com/nt-linux-benchmarks.html>
- M. Loukides: *A Brief History of UNIX*; (vermutlich) 2003
<http://www.uwsg.iu.edu/usail/external/recommended/unixhx.html>



„byte code“



- K.V. Nori et al.: *Pascal-P implementation notes*. In D.W. Barron (ed.): *Pascal – The language and its implementation*; John Wiley & Sons, 1991, 125 - 170
- G. Krassner: *The Smalltalk-80 virtual machine*; Byte 6 (1981) 8, 300 – 320
- A. Goldberg, D. Robson; *Smalltalk 80: The language and its implementation*; Addison-Wesley, 1983, Kap. 26 – 30, 541 ff.
- (dito): *Smalltalk 80*; Addison-Wesley, 1989, Kap. 20, 417 ff.
- L.P. Deutsch, A. Schiffman: *Efficient implementation of the Smalltalk-80 system*; Conf. Record 11th Annual ACM Symposium Principles of Programming Languages, 1984, 297 - 302
- E. Miranda: *Overview of different implementations of Smalltalk*; (undated), <http://wiki.cs.uiuc.edu/VisualWorks/>

[Overview+of+different+implementations+of+Smalltalk!](http://wiki.cs.uiuc.edu/VisualWorks/Overview+of+different+implementations+of+Smalltalk!)

„byte code“ (1. Forts.)



- J. Gosling: *Java intermediate bytecodes*, ACM SIGPLAN Notices, 30 (1995) 3, 111 – 118
- T. Lindholm, F. Yellin: *The Java Virtual Machine Specification*; Addison-Wesley, 1996
- B. Venners: *Bytecode basics*, 1996,
<http://www.javaworld.com/javaworld/jw-09-1996/jw-09-bytecodes.html>
- C. Gittinger: *Die Unified Smalltalk/Java Virtual Machine in Smalltalk/X*; Firmenschrift eXept Software AG, 2000
www.exept.de/exept/english/Smalltalk/presentations/oop_2000/oop2000.ppt

„byte code“ (2. Forts.)



- A. Nudelman: *C# Tutorial*; 2001 – 2005,
<http://csharpcomputing.com/Tutorials/TOC.htm>
- J. Robbins: *ILDASM (Intermediate Language Disassembler) is Your New Best Friend*; MSDN Magazine, May 2001
<http://msdn.microsoft.com/msdnmag/issues/01/05/bugslayer/print.asp>
- W. Le: *A Microsoft Virtual Machine - Introduction to the .NET and C#*; 2002,
<http://www.cs.virginia.edu/~qc9b/cs851/vm-2.pp>
- D. Gruntz: *C# and Java: The Smart Distinctions*; Journal of Object Technology, 1 (2002) 5, 163 - 176
http://www.jot.fm/issues/issue_2002_11/article4
- A. Kind: *Bytecode-Interpretierung*; Informatik Spektrum 20 (1997) 2, 109 - 110
- B. Meyer: *Approaches to portability*; J. Object-oriented Programming 11 (1998) 4, 68 - 70

Algol, ..., Ada



- A. Endres: *Software und Software-Entwicklung im Wandel: Ein historischer Vergleich* ; Informatik Spektrum 16 (1993) 5, 261 - 265
- P. Naur: *Revised report on the algorithmic language Algol 60*; Comm. ACM 6 (1963) 1, 1 - 17
- NN: *Algol 60*; 2005, http://de.wikipedia.org/wiki/Algol_60
- G. Radin, H.P. Rogoway: *NPL: Highlights of a new programming language* ; Comm. ACM 8 (1965) 9 - ??
- R.C. Holt: *Teaching the fatal disease (or) Introductory programming using PL/I* ; ACM SIGPLAN Notices 8 (1973) 5, 8 - 23
- D. Beech: *A structural view of PL/I* ; ACM Computing Surveys 2 (1970) ?, 33 - 64
- A. Schulz: *Strukturiertes Programmieren in der problemorientierten Programmiersprache PL/I* ; Angewandte Informatik 19 (1977) 2, 51 - 57

Folgeseite!

Algol, ..., Ada (1. Forts.)



- A. van Wijngaarden et al.: *Revised report on the algorithmic language ALGOL 68*; Acta Informatica 5 (1975) issues 1-3
- C. H. Lindsey: *A history of ALGOL 68*; 2nd ACM SIGPLAN Conf. History of Programming Languages, 1993, 97 – 132
- C.H. Lindsey: *ALGOL 68 with fewer tears*; BCS Computer Journal, 15 (1972) 2, 176 – 188
- NN: *ALGOL 68 im Aspekt einer modernen Programmiersprache?*. In AINF – Projekt: *PROGRAMMIERSPRACHEN*; 1999, <http://www.fundus.org/referat.asp?ID=9899>
- NN: *Algol 68*; Computer Science Study Guide, 2005, <http://www.bookrags.com/sciences/computerscience/algol-68-wcs.html>
- S.M. Valentine: *Comparative notes on Algol 68 and PL/I*; BCS Computer Journal 17 (1974) 4, 325 - 331

Folgeseite !

Algol, ..., Ada (2. Forts)



- NN: *Ada 83 Reference manual* ;
<http://www.adaic.org/standards/ada83.html>
- NN (Wikimedia): *Ada Lovelace*;
http://de.wikipedia.org/wiki/Ada_Lovelace
- NN: *Ada 95 Reference manual* ;
<http://www.adaic.org/standards/ada95.html>
- NN (Wikimedia): *Ada (Programmiersprache)*; 2005,
[http://de.wikipedia.org/wiki/Ada_\(Programmiersprache\)](http://de.wikipedia.org/wiki/Ada_(Programmiersprache))
- H. Wacker: *Die Programmiersprache Ada*; 2005,
<http://www.wackerart.de/ada.html>
- D.L. Parnas: *Is ADA Too Big?* ; *Comm. ACM*, 27 (1984) 11, 1155, und *Comm. ACM*, 28 (1985) 7, 753 (Technical Correspondences)

Allgemein: NN (Wikimedia): *Timeline of programming languages*
http://en.wikipedia.org/wiki/Programming_language_timeline



- N. Wirth: *The programming language Pascal*; Acta Informatica 1 (1971) ??, 35 - 63
- N. Wirth: *On the design of programming languages*; Proc. IFIP Congress, 1974, 386 - 393
- B. Hensen: *Einfachheit von Programmiersprachen*; 2005
<http://www.fh-wedel.de/~si/seminare/ws04/Ausarbeitung/1.Sprachentwurf/prinzipien4.html>
- N. Wirth: *A brief history of Modula and Lilith* ; 1995,
<http://www.modulaware.com/mdlt52.htm>
- N. Wirth: *The Programming Language Oberon*; 1992/95,
<http://www.mathematik.uni-ulm.de/oberon/reports/report-1992.html>
- P. Floyd: *A Discussion of Oberon*; 1998,
<http://www.edm2.com/0608/oberon.html>

Folgeseite !



- A. Koenig: *C Traps and Pitfalls*; 1989,
<http://www.literateprogramming.com/ctraps.pdf>
 - J. Kopp: *Introduction to C and C++: The history of C and C++, their uses and differences*; 2005,
<http://cplus.about.com/od/beginnerctutorial/1/aa010702a.htm>
 - B. Möсли; *A Comparison of C++, FORTRAN 90 and Oberon-2 for Scientific Programming*; 1998,
<http://www.arithmetica.ch/Oberon/CFORTRANOberon.nhtml>
 - S. Feizabadi: *History of the World Wide Web / History of Java*; 1998,
http://ei.cs.vt.edu/~wwwtb/book/chap1/java_hist.html
 - C. Wille: *A Brief History of C#*; 2004,
<http://www.aspheute.com/artikel/20000713.htm>
- B. Thomsen: *How would you like to programme in 20 years?*; 2005, www.nouhauz.dk/dokumenter/120505-BentThomsen.ppt**

Simula / Smalltalk



- O.J. Dahl, K. Nygaard: *Class and subclass declarations*. In: J.N. Buxton: *Simulation Programming Languages*, North Holland, 1967, 158 - 174
- O.J. Dahl: *The roots of object orientation – The Simula language*. In: M. Broy, E. Denert: *Software Pioneers*, Springer, 2002, 78 - 90
- (Sonderheft): *Smalltalk*, Byte 6 (1981) 8
- A. Goldberg, D. Robson: *Smalltalk 80*, Addison-Wesley, 1989
- D.A. Thomas et al.: *Smalltalk with style*, Prentice Hall, 1995
- NN: *Smalltalk programming language*, (undated), http://www.4reference.net/encyclopedias/wikipedia/Smalltalk_programming_language.html
- S.R. Alpert et al.: *The Design Patterns Smalltalk Companion*. Addison Wesley, 1998
- O.L. Madsen et al.: *Object-oriented programming in the BETA programming language*, Addison-Wesley / ACM Press 1993

Semaphore u.a.

(E.W. Dijkstra, 1930 – 2002. C.A.R. Hoare, 1934)



- E.W. Dijkstra: The Structure of the "THE"-Multiprogramming System; Comm. ACM 11 (1968) 5, 341-346 (im Anhang)
- C.A.R. Hoare: *Towards a Theory of Parallel Programming*. In C.A.R. Hoare, R.H. Perrott (eds.): *Operating Systems Techniques*; Academic Press, 61-71
- C.A.R. Hoare: *Monitors – an operating system structuring concept*; Comm. ACM 17 (1974) 10, 549 – 557
- O. Eriksen: *Concurrency*; (undated)
www.daimi.au.dk/~oe/concur/Slides/02-interference.pdf
- NN:
http://de.wikipedia.org/wiki/Semaphor_%28Informatik%29
[http://www.encyclopedia-online.info/Semaphore_\(programming\)](http://www.encyclopedia-online.info/Semaphore_(programming))
<http://de.wikipedia.org/wiki/Interprozesskommunikation>



- C.A. Petri: *Kommunikation mit Automaten*; Dissertation, TH Darmstadt, 1962
- G. Seegmüller et al.: *Ein Meilenstein für die Informatik / Carl Adam Petri - eine herausragende Forscherpersönlichkeit / Carl Adam Petri und die Informatik*; Laudationes, 1991, <http://www.informatik.uni-hamburg.de/TGI/mitarbeiter/profs/petri/laudatio.html>
- A. Zimmermann: *Petri Nets*; (undated), <http://pdv.cs.tu-berlin.de/~azi/petri.html>
- H. Ehrig et al.: *Petri Net Technology for Communication-Based Systems*; Springer, LNCS 2472, 2003
- A. Oberweis: *Vom isolierten Petri-Netz-Werkzeug zur integrierten Entwicklungsumgebung für Informationssysteme*; 2005, lwi2.wiwi.uni-frankfurt.de/ftproot/pub/publikationen/sonstige/U23.pdf



- *GAMM/NTG-Empfehlungen für das Informatikstudium an den Universitäten*; 1968. Abgedruckt in W. Brauer, S. Münch: Studien- und Forschungsführer Informatik. Springer, 1996, 55 - 60
- Donth: *Aufbau der Informatik an deutschen Hochschulen*; Elektronische Rechenanlagen, **26 (1984), 223 - 228**
- W. Görke, F.-J. Brandenburg: *25 Jahre Fakultätentag Informatik - 1973 - 1998*;
http://goethe.ira.uka.de/~goerke/Brosch_Fakttag_erg.pdf
- C. Pieper: *Neue Disziplinen als Innovationsmotor? Die Entstehung des Faches "Informatik" im ost- und im westdeutschen Hochschulwesen der 60er und 70er Jahre*; 2004,
<http://www.innovationskultur.mwn.de/papers/pieper.pdf>
- Konrad-Zuse-Medaille, Preisträger
<http://www.gi-ev.de/verein/wettbewerbe/zuse-medaille.shtml>
- NN (Gesellschaft für Informatik): *Was ist Informatik?*;
<http://http://www.gi-ev.de/themen/was-ist-informatik/gi-positionspapier-was-ist-informatik.pdf>



- Zur Erinnerung an A. Walther (1998 – 1967), siehe u.a.:
http://edoc.hu-berlin.de/e_rzm/18/biener-klaus-1999-08-01/PDF/14.pdf
- Algol 60-Bericht, "ZMMD"
- Alwin-Walther-Medaille
- R. Piloty (1924), siehe u.a.:
http://www.netzwelt.de/lexikon/Robert_Piloty.html
- D.L. Parnas (1941, siehe Folien 10 - 12)
- Zur Erinnerung an G. Lustig (1929 – 1993):
<http://www.informationwissenschaft.org/hochschulverband/preis.htm>

Nachwort



If a language needs a book this thick, then it is too complex for everyday use.

"When you design a language, you should not only consider what to put in, but also what to leave out." - Niklas Wirth

C++ has become an end to itself. Developers love it because they can tinker endlessly with the programming without ever coming close to solving a real-world problem.

Granted, Ada's language spec is even thicker. And while every surprise that you might encounter is properly documented, the sheer bulk ensures you won't know about it until you see for yourself.

Try Smalltalk. Just 20 pages to define the language. The rest of a Smalltalk book describes the IDE (the refactoring browser) and the libraries. Modern Smalltalks have everything that Java has, except the need for a "More Effective Smalltalk".

<http://books.slashdot.org/article.pl?sid=05/06/07/1531214&threshold=-1&tid=156&tid=6>

Und das gilt nicht nur für Programmiersprachen !

“Allgemeine Geschäftsbedingungen”



- Referat maximal 40 Minuten
- Datenprojektor („*beamer*“) ist vorhanden
- Anschließend Diskussion
 - inhaltlich
 - Vortragsstil
- Auf Anforderung, mindestens fünf Tage vorher, kann PC für Präsentation mit PowerPoint bereitgestellt werden
- Ausarbeitung maximal acht bis zehn Seiten DIN A4;
(in der Regel) nach einer Woche abzugeben (Papier);
Literaturangaben nicht vergessen!
- Nach Bestätigung (eventl. Korrekturwünsche) durch Betreuer
Abgabe der endgültigen Ausarbeitung innerhalb einer
Woche als doc-/pdf-Datei
(maximal 2 Mbyte groß, mit Namen “WS05-06-#xx-version-ii.ext”,
“xx” elem {01 ÷ 17}, “ii” elem {1 ...})



Web-Seite des Seminars:

<http://www.informatik.tu-darmstadt.de/PU/Seminar-SW>

(dort findet sich u.a. die Vorlage für das
Deckblatt Ihrer Ausarbeitung
und der **Terminplan** !)

**Nichts als Anstand, Sie beim
Lesen nicht stolpern zu lassen.**



HJH#??

2005/2006

<Thema>



- <offen>