A Biography of Ole-Johan Dahl

Olaf Owe, Stein Krogdahl, Tom Lyche*

Department of Informatics, University of Oslo, Norway

On 12 October 1931, Ole-Johan Dahl was born to the family of a ship captain in Mandal, the southernmost city of Norway. In 1952, three years after beginning his studies at the University of Oslo, he was introduced to computers at the Norwegian Defense Research Establishment (NDRE) where he fulfilled his military service obligation. Jan V. Garwick was responsible for the field of mathematical analysis and calculations, and Ole-Johan was placed in the "computing room" led by Garwick's assistant, Kristen Nygaard. It is quite likely that, in this setting, Garwick, Nygaard and Dahl were the first in Norway to develop programs on "large" digital computers. In the years to come, NDRE cultivated a scientific collaboration with the pioneering computer group at the University of Manchester and the electronics company Ferranti. As a result of this tie, NDRE got the first version of Ferranti's Mercury computer in 1957.

Ole-Johan's next project was to develop and implement a "high level" language for numerical computation, called MAC (Mercury Automatic Coding). While Kristen changed focus to the area of operations research in the mid-1950s, Ole-Johan became Garwick's main collaborator. Together they formed the first real programming group in Norway.

In 1958, Ole-Johan completed the Candidatus Realium degree at the University of Oslo in mathematics. His thesis entitled "Multiple Index Countings on the Ferranti Mercury Computer" [94], was formally in the area of numerical analysis, but actually in computer science and programming, a field that had not yet emerged. In fact, he was one of the first to acquire a relevant and modern education in computer science.

In 1960 Kristen had become research director at the Norwegian Computing Center (NCC). He decided to make an attempt solve two main problems in operations research, namely, *the lack of concepts and language for description and communication about large and complex systems*, and *the lack of a specialized programming language for simulation tasks*. Realizing that he could not do this by himself, he looked to Ole-Johan, a specialist in programming language, as the obvious collaborator.

They started working together in 1961, and in 1963 Ole-Johan joined Kristen full-time at NCC. Together they created the Simula 1 language (1961-1965) and Simula 67 (1965-1968), introducing the concepts of class, subclass and inheritance, virtual binding of operations, and dynamic creation of objects. The Simula concept of quasi-parallelism reflected that objects may in principle be independent processes running in parallel. Implicit forms of information hiding,

* in O.Owe et al (Eds.): "From Object-Orientation to Formal Methods", LNCS 2635 Festschrift, dedicated to the Memory of Ole-Johan Dahl.

through the subclass mechanism, were later also complemented by explicit forms of information hiding.

These concepts, which constitute what today is called object-orientation, have greatly influenced modern programming languages, programming methodology and modeling languages, including UML. The class related concepts in Simula were clearly ahead of their time; it took some 20 years until they gained understanding and popularity. Languages such as Smalltalk, Beta, C++, Eiffel, Java, and C#, have directly adopted Simula's fundamental concepts about objects and classes. Object-orientation is today the dominant style of programming, description, and modeling.

To quote Bjarne Stoustrup: "Simula isn't just an innovative programming language. From the beginning it came with a philosophy of design based on modeling that has had impact far beyond the realm of programming and programming languages." The object-oriented philosophy also underlies the modern use of windows and the graphical interfaces that we all use.

How could Ole-Johan and Kristen, at such an early stage, design a language with all the mechanisms that today form the "object-oriented" paradigm for system development? An important part of the answer is obviously that they were extraordinary talented researchers in the field, and with rather different perspectives and personalities, which frequently led to confrontations and heated discussions. There is a famous story illustrating their style of working: A new employee at NCC came worriedly running down to the receptionist and cried out: "We must do something! There are two men fighting upstairs in front of the blackboard." The receptionist listened for a moment, and replied: "No don't worry. It is just Kristen and Ole-Johan discussing Simula!"

In addition, it was probably very fortunate that they first designed a language for simulation (Simula 1), and later generalized the concepts in a general purpose language (Simula 67). Ole-Johan has expressed it this way: "A reason for this may be that in developing a complicated simulation model it is useful to decompose it in terms of 'objects', and to have an explicit mapping from external objects to program constructs. A natural way of developing real systems is not much different." Kristen emphasized that an essential motivation behind Simula was system description, and the need for a language to model real world concepts.

The final recognition of Ole-Johan and Kristen as the founders of object orientation has been established through two prestigious awards, both given during their last year of life: In November 2001, they were awarded the IEEE John von Neumann Medal "for the introduction of the concepts underlying object-oriented programming through the design and implementation of Simula 67," and in February 2002, they were given the A. M. Turing Award by the ACM "for ideas fundamental to the emergence of object-oriented programming, through their design of the programming languages Simula 1 and Simula 67" [tribute].

Earlier, in June 2000, they were awarded Honorary Fellowships for "their originating of object technology concepts" by the Object Management Group, the main international standardization organization within object-orientation. In August 2000 they were appointed Commanders of the Order of Saint Olav by the King of Norway (one of the highest national awards), and in 1999 they became the first to receive the Rosing Honorary Prize, awarded by the Norwegian Data Association for exceptional professional achievements. Both were elected members of the Norwegian Academy of Science. Ole-Johan was also a member of IFIP Working Group 2.3 and of Academia Europaea, and he received an honorary doctorate from the University of Lund, Sweden.

A characteristic of Ole-Johan as a researcher was his quest for simplicity, elegance, and purity rather than ad hoc solutions with their associated exceptions and compromises. This is reflected in the class concept of Simula. From an educational point of view, its elegance, generality and simplicity make Simula well suited for teaching object orientation.

We include below a version of the last paper Ole-Johan wrote [36][1], which provides a detailed summary of the research leading up to Simula, as well as afterthoughts and viewpoints of its cultural impact. Other sources of information on the development of Simula are [23], and more recently [Hol94, Kro03, Bös03].

Research at the University of Oslo

In 1968 Ole-Johan was invited to be a full professor to create the discipline of computer science at the University of Oslo. He spent enormous efforts building up a curriculum in computer science in the late 1960s, writing textbooks at night while teaching during the day, and supervising up to 20 graduate students at a time. For 10 years from 1968 he was the only professor in computer science in Oslo. For many years, he had only one lecturer and a few research assistants helping him. The courses he designed met the highest international standards of the time and for many years to come. Most of the courses were offered for 20 years or more with only minor changes; some are still being taught.

After beginning his career at the university, Ole-Johan deliberately stopped working on further development of Simula as such. In his new position he felt that it was essential to build up computer science to be an accepted academic discipline, and establish a theoretical foundation for basic concepts of computer science and programming languages. He made important advances in programming methodology, introducing techniques for program structuring and conceptual modeling, based on the experiences of the design and implementation of Simula. Early works in this direction include the papers [1, 4, 10, 13, 14, 16, 17, 63] with his work on *Hierarchical Program Structures* as the best known and most influential.

Inspired by Tony Hoare's logic for program reasoning [Hoa69], he continued research in the area of program architecture, programming and specification languages, as well as verification methods and techniques. Most of this work is related to the area of formal methods, where the idea is to use mathematical methods to specify, develop and reason about programs. Because of his computer science background and education, his theoretical work was accompanied by con-

[1] References labeled with numbers refer to the bibliography of Ole-Johan.

cern for practical relevance and usefulness. Exploiting the advantages of both, he advocated combined use of top-down development and bottom-up development.

This led to research on abstract data types, a concept inspired by the class concept of Simula and Tony Hoare's paper entitled *Proof of correctness of data representation* [Hoa72]. In particular, Ole-Johan focused on generator induction, inspired by the work of John Guttag and Jim Horning, subtyping, inspired by Simula's subclass mechanism and by the work on order sorted algebra, and integration of applicative and imperative class-based reasoning. Based on mechanizable evaluation and reasoning, he developed a theory for "constructive" types and subtypes centered around a long term project, called Abstraction Building, Education Language (ABEL), which served as a research testbed and source of student projects [5, 21, 25, 26, 33, 34, 43, 44, 46, 53], and resulted in several dr. scient theses supervised by Ole-Johan [Owe80, Nos84, Mel86, Lys91].

Ole-Johan was teaching formal methods, and their practical use for 30 years. He believed that computer science students should know the principles of program reasoning, and that this would make them better programmers even without performing detailed verification. The course work has resulted in the book, *Verifiable Programming* [5], which includes much of his own research and results. He supervised a large number of students, ten of whom became university professors.

Formal methods, object-orientation and concurrency

Work towards the understanding and formalization of what is now called object-orientation, was carried out already in the early 1970s [15, 18], and with the thesis of Arne Wang, one of Ole-Johan's first students [Wan74].

Ole-Johan's early approach to reasoning about object-oriented systems builds on the idea of limiting direct access to attributes of an object from outside, either disallowing all remote variable access or allowing access to some variables (seen as an abbreviation for implicit read and write operations). This means that one can give local invariants in a class and prove that the class invariant is established and maintained by looking at the text of the class itself and possibly superclasses. When subclassing is restricted so that super-invariants are respected, reasoning about objects can be done without looking at the global state space: Hoare style reasoning can be done locally in each class.

Early on it was recognized that this kind of class-based reasoning was fruitful for the Monitor concept of Hoare [Hoa74] and for aspects of operating systems concerning process control. The co-routine and monitor papers were part of this research direction [15, 18, 19, 20, 21, 29, 35, 64] and the thesis of Stein Gjessing, who had been supervised by Ole-Johan [Gje83].

In contrast to those object-oriented approaches where "everything is an object", Ole-Johan believed in object classes side by side with data types. Ole-Johan's view was that objects should reflect mutable data structures, handled by references, and data types should reflect immutable, but copyable, data [28]. This called for user defined data types. In ABEL, a functional sublanguage was defined for definition of abstract data types, whereas classes were defined in an imperative style. Thus Ole-Johan considered functional programming as a complement to object-oriented programming rather than a competitor.

According to the original ideas of Simula, an object would in general have its own activity, as well as data and procedures. Objects with activity were reflecting "independent processes", and objects without activity (but still with data and procedures) were called passive. In Simula 67 these ideas were realized by allowing objects to be co-routines, the natural way at that time to imitate concurrent processes and a useful simulation mechanism. In today's world a natural adaptation would be to let objects be concurrent processes, and one would obtain a distributed system by a set of objects running in parallel and interacting by remote method calls (only). This is the approach taken in Ole-Johan's later works. See for instance [26, 36].

Ole-Johan's work on abstract specification of concurrent objects by means of histories presents a techniques for "black box" interface specification of process classes. The abstract state of a concurrent object is represented by its communication history, i.e., the trace of all visible communication events involving the object, such as method calls. As the abstract state at any given time is reflected by a finite history, Ole-Johan developed specification techniques based on finite traces, and a central idea was to use right-append as a trace generator in order to describe new actions in terms of the current history, thereby avoiding recursive process definitions as found e.g. in CSP. Reasoning about concurrent objects in terms of such histories is compositional and integrates well with object-orientation. Ole-Johan developed a style of history specification where specifications of a certain form can easily be refined into an imperative object-oriented implementation. The use of histories was initiated in the early 1970s and remained an important research topic for him throughout the 1990s [21, 22, 26, 41, 58]. When he retired, Ole-Johan was writing a book on concurrency based on research in connection with a course on concurrency and process control [69, 54].

The above principles for object-oriented programming, specification, and reasoning, constitute what we may call "the Dahl School." A further introduction is given in the paper by Johnsen and Owe in this volume.

International visits

Due to Ole-Johan, several prominent researchers visited Oslo. In particular we mention the one year stay of Donald Knuth from Stanford University in 1972 to 1973. Knuth had a great impact on the computer science development in Oslo at an early and crucial period in time.

Knuth, who immediately understood the benefits of the Simula ideas, gave up work on his own simulation language (SOL), and became a supporter of Simula. He could have made Simula quite well-known by teaching it at Stanford, but when asking for a inexpensive compiler for academic use at Stanford, he was unfortunately turned down by NCC (despite the strong arguments of both Kristen and Ole-Johan, who fully understood the importance of this opportunity).

In the late 1970s Reiji Nakajima, a post doc at that time, made a one year visit to Oslo, which led to a number of interesting discussions around abstract data types and ABEL. In the early 80s, Neelam Soundarajan visited Oslo for a year, working together with Ole-Johan on reasoning with histories. Ole-Johan enjoyed this cooperation very much, appreciating Neelam's clarity. Neelam came for a second year in the early 1990s. Zhenjia Lei, University of Xi'an also visited Ole-Johan, which resulted in a return visit by Ole-Johan in Xi'an in the mid-1990s. Apart from this visit, Ole-Johan had only one sabbatical leave during his career, which was spent at Stanford University in the late 1970s.

Ole-Johan also enjoyed the many shorter visits by a large number of scientists, including Tony Hoare, Hans Langmaack, Dines Bjørner, Eugene Kindler, Cliff Jones, Manfred Broy, David Luckham, Jean-Pierre Jouannaud, Pierre Lescanne, and Willem-Paul de Roever, most of whom also enjoyed music evenings and dinners in Ole-Johan's home.

Personal interests

Ole-Johan was a music lover and an excellent amateur pianist. In fact, he was one of the best "prima vista" amateur pianists in Norway. He knew the world of classical music well, and that of chamber music in particular. Much of his free time was filled by music, and he was a central member of the Board of Oslo Quartet Association and a driving force behind the yearly chamber music courses at the Nansen School in Lillehammer. He regularly arranged house concerts at the department, often playing together with visitors who happened to be musicians, or with his wife or daughter.

In addition he enjoyed and excelled in many kinds of games including chess and bridge, and spent much time as a student pursuing these interests. At conferences he was known for his skills in classical billiards (3 balls) in addition to the piano. A personal bibliography is written by his wife [Dah03].

References

[Bös03] Laszlo Böszörmenyi, Stefan Podlipnig: People behind Informatics: In memory of Ole-Johan Dahl, Edsger W. Dijkstra and Kristen Nygaard. Book written for the memorial exhibition at the international conferences JMLC 2003 and EuroPar 2003, Institute of Information Technology, University of Klagenfurt, Austria, August 2003. `http://www-itec.uni-klu.ac.at/~laszlo/Memorial/memorial_exhibition.htm`

[Dah03] Tove Dahl: A brief biography of Ole-Johan Dahl. 2003. In [Bös03].

[Gje83] Stein Gjessing: Aspects of Semantics and Verification of Monitors, dr. philos. thesis, Department of Informatics, University of Oslo, 1983.

[Hoa69] C.A.R. Hoare: An axiomatic basis for computer programming. *Comm. ACM* 12 (1969), pp. 576–580.

[Hoa72] C.A.R. Hoare: Proof of the Correctness of Data Representations. *Acta Informatica*, vol. 1, 1972, pp. 271–281.

[Hoa74] C.A.R. Hoare: Monitors: an Operating System Structuring Concept. *Comm. ACM* 17(10)(1974), pp. 549-557.

[Hol94] Jan Rune Holmevik: Compiling Simula: A Historical Study of Technological Genesis. *IEEE Annals of the History of Computing*, Vol. 16 no. 4, 1994.

[Kro03] Stein Krogdahl: The birth of Simula. In the final proceedings of HiNC 1, Trondheim, June 2003. IFIP. To appear.

[Lys91] Olav Lysne: term rewriting. dr. scient. thesis, Department of Informatics, University of Oslo, 1991.

[Mel86] Sigurd Meldal: On Hierarchical Abstraction and Partial Correctness of Concurrent Structures. dr. scient. thesis, Department of Informatics, University of Oslo, 1986.

[Nos84] Decision Algorithms for Program Verification. Dr. scient. thesis, Department of Informatics, University of Oslo, 1984.

[Owe80] Olaf Owe: A specification technique with idealization. Dr. scient. thesis, Department of Informatics, University of Oslo, 1980.

[Wan74] Arne Wang: Generalized Types in High-Level Programming Languages. Research Report in Informatics no. 1, 1974, cand. real. thesis. Department of Mathematics, University of Oslo.

[tribute] The Department's page of tribute: `http://www.ifi.uio.no/adminf/tribute.html` with links to Ole-Johan's and Kristen's own homepages (Kristen's is more rich on information). For Kristen there is a memorial page at `http://www.ifi.uio.no/in_memoriam_kristen/` See also `http://www.jot.fm/issues/issue_2002_09/eulogy`

A Bibliography of Ole-Johan Dahl

Olaf Owe, Stein Krogdahl, Tom Lyche

Department of Informatics, University of Oslo, Norway

This bibliography is partly based on information collected by Berit Strange, from the Library of the Department of Informatics, University of Oslo. The website `www.ifi.uio.no/~olejohan` contains links to some papers.

Books

1. *Structured Programming.* O.-J. Dahl, E.W. Dijkstra, C.A.R. Hoare. Academic Press, London, 1972. (220 pages). Also as: *A.P.I.C. Studies in Data Processing* no. 8. ISBN 0-12-200550-3, 0-12-200556-2
2. *SIMULA begin.* G.M. Birtwistle, O.-J. Dahl, B. Myhrhaug and K. Nygaard. Auerbach Publishers Inc, 1973 (391 pages) ISBN 91-44-06211-7. 2. edition published by Studentlitteratur, Stockholm & Chartwell-Bratt Ltd, England, 1973
3. *Syntaks og semantikk i programmeringsspråk (Syntax and Semantics in Programming Languages).* O.-J. Dahl. Studentlitteratur, Lund, Sweden, 1972 (103 pages) ISBN 91-44-07111-6 (In Norwegian)
4. *Algoritmer og datastrukturer (Algorithms and Data Structures).* O.-J. Dahl and Dag Belsnes. Studentlitteratur, Lund, Sweden, 1973 (170 pages) ISBN 91-44-06991-x (In Norwegian)
5. *Verifiable Programming.* O.-J. Dahl. Prentice Hall (International series in computer science), New York, 1992, revised edition 1993 (269 pages) ISBN 0-13-951062-1

Papers in Proceedings, Journals, and Books

6. Automatisk koding – et prosjekt ved forsvarets forskningsinstitutt [English: Automatic coding – a project at the Norwegian Research Defense Establishment]. O.-J. Dahl. (In Norwegian) *In: Proceedings of NordSAM*, May 1959, Karlskrona. (The Nordic Symposium on Application of Mathematical Machines) Carl-Erik Frøberg and Yngve Rollof (eds.), 1959, pages 135-141.
7. Litt om symbolbruken i Algol [English: A Note on the Use of Symbols in Algol]. O.-J. Dahl. *In: BIT* (Nordic Journal of Information Processing) 2(1962) no. 1, pages 7-8. (In Norwegian)
8. SIMULA – a Language for Describing Discrete Event Systems. K. Nygaard and O.-J. Dahl. *In: Proceedings of the IFIP Conference, 1965.* Vol. 2, Spartan Books, Washington, D.C.; Macmillan, New York, pages 554-555.
9. SIMULA – an ALGOL-Based Simulation Language. O.-J. Dahl and K. Nygaard. *In: Communications of the ACM* 9(1966), pages 671-682.

10. Discrete Event Simulation Languages. O.-J. Dahl. *In: Programming Languages.* G. Genuys (ed). NATO Advanced Study Department. Academic Press, 1968, pages 349-395.
11. Class and Subclass Declarations. O.-J. Dahl and K. Nygaard. *In: Simulation Programming Languages,* J.N. Buxton (ed.), Proceedings of the IFIP working conference on simulation programming languages, Oslo, May 1967, North-Holland, Amsterdam, 1968, pages 158-174.
12. Some Features of the SIMULA 67 Language. O.-J. Dahl, B. Myhrhaug and K. Nygaard. *In: Digest of the Second Conference on Application of Simulation,* Dec. 1968, New York, IEEE (cat no. 68C60-SIM) pages 29-32.
13. Decomposition and Classification in Programming Languages. O.-J. Dahl. *In: Linguaggi nella società e nella tecnica, Convegno promosso dalla Ing. C. Olivetti & C., S. p. A. per il centenario della nascita di Camillo Olivetti,* Milano, 1970. (Saggi di cultura contemporanea 87) pages 371-383.
14. Programming Languages as Tools for the Formulation of Concepts. O.-J. Dahl. *In: Proceedings of the 15th Scandinavian Congress*, Oslo 1968, K.E. Aubert and W. Ljunggren (eds.). *Lecture notes in Mathematics* no. 118, Springer, 1970, pages 18-29.
15. Coroutine Sequencing in a Block Structured Environment. Arne Wang and O.-J. Dahl. *In: BIT* 11(1971) pages 425-449.
16. Hierarchical Program Structures. O.-J. Dahl and C.A.R. Hoare. *In: Structured Programming.* Academic Press, 1972, pages 175-220 (see [1] above)
17. Programming Discipline. O.-J. Dahl. *In: Proceedings of the 1974 CERN School of Computing,* Godøysund, Aug. 1974. CERN Report 74-23, Geneva, pages 426-435.
18. An Approach to Correctness Proofs of Semicoroutines. O.-J. Dahl. *In: Mathematical Foundations of Computer Science,* 3rd Symposium, Jadwisin near Warsaw, 17-22 jun, 1974. *LNCS* no. 28, Springer, 1975, pages 157-174.
19. Analysis of an Algorithm for Priority Queue Administration. A. Jonassen and O.-J. Dahl. *In: BIT* 15(1975) no. 4, pages 409-422.
20. A Model for Controlling a Network of Processors and Storage Units. O.-J. Dahl. *In: Théorie des algorithmes, des langages et de la programmation: Textes des exposés du séminaire organisé par l'Institut du Recherche d'Informatique et d'Automatique (IRIA), Rocquencourt, Octobre 1974.* M. Nivat (ed.). IRIA, Rocquencour, France, 1975, pages 83-94.
ISBN 2-7261-0120-8
21. Can program proving be made practical? O.-J. Dahl. *In: Les fondements de la programmation,* M. Amirchahy and D. Néel (eds.). IRIA, Le Chesnay, France (Text in English.) pages 57-114. ISBN 2726101844
22. Time Sequences as a Tool for Describing Program Behaviour. O.-J. Dahl. *In: Abstract Software Specifications, January 22–February 2, 1979, Copenhagen Winter School,* D. Bjørner (ed.). *LNCS* no. 86, Springer 1979, pages 274-290.
23. The Development of the Simula Languages (Chapter IX: SIMULA Session). K. Nygaard and O.-J. Dahl. *In: History of Programming Languages.* Richard L. Wexelblat (ed.), Academic Press, New York, 1981, pages 439-493.
Also in: ACM Sigplan History of Programming Languages Conference, June, 1978. *ACM Sigplan Notices* 13(1978) no. 8, pages 245-272.

24. Object Oriented Specification. O.-J. Dahl. *In: Research Directions in Object-Oriented Programming,* Bruce Shriver and Peter Wegner (eds.), MIT Press, Cambridge, 1987, pages 561-576.
(Presented at the Object-Oriented Programming Workshop, June, 1986.)
25. Generator induction in order sorted algebra. O. Owe & O.-J. Dahl. *In: Formal Aspects of Computing* 3(1991) pages 2-20.
26. Formal Development with ABEL. O.-J. Dahl and O. Owe. *In: Proceedings of Formal Software Development Methods.* VDM '91, Oct. 1991. *LNCS* no. 552, Springer, 1991, pages 320-362.
27. Object Orientation and Formal Techniques (extended abstract). O.-J. Dahl. *In: VDM '90 VDM and Z – Formal Methods in Software Development. LNCS* no. 428, Springer, 1992, pages 1-11.
28. Value Types and Object Classes. O.-J. Dahl. *In: ASU Newsletter* 20(1992) no. 1, pages 8-20.
29. Monitors Revisited. O.-J. Dahl. *In: A classical Mind: Essays in Honour of C.A.R. Hoare.* A.W. Roscoe (ed.). Prentice Hall, 1994, pages 93-103.
30. Relating a Simulation Model to an Applicative Specification. O.-J. Dahl. *In: Modelling and Simulation: Proceedings ESM, Praha, 1995,* M. Snorek, M. Sujansky, A. Verbraeck (eds.) Society for Computer Simulation International, 1995, pages 633-638.
31. *Recent Trends in Data Type Specification: 11th Workshop on Specification of Abstract Data Types jointly with the 8th COMPASS Workshop, Oslo, Norway, September 19-23, 1995, Selected papers.* Magne Haveraaen, O. Owe, O.-J. Dahl (eds.). *LNCS* no. 1130, Springer, 1996, (550 pages)
ISBN 3-540-61629-2
32. Data Access Safety and Storage Economy in Programming Languages. O.-J. Dahl. (8 pages) 23rd ASU Conference, August 1997, Stara Lesna, Slovakia. *ASU Newsletter* vol. 24, no. 2, Feb. 1998.
33. On Introducing Higher Order Functions in ABEL. O.-J. Dahl and Bjørn Kristoffersen. *In: Nordic Journal of Computing* 5(1998) pages 50-69.
34. Subtyping and Constructive Specification. O.-J. Dahl, O. Owe, Tore J. Bastiansen. *In: Nordic Journal of Computing* 5(1998) pages 19-49.
35. A Note on Monitor Versions: an Essay in the Honour of C.A.R. Hoare. O.-J. Dahl. *In: Millennial Perspectives in Computer Science* (Proceeding of the 1999 Oxford-Microsoft Symposium in Honour of Sir Tony Hoare). Jim Davis, Bill Roscoe and Jim Woodcock (eds.). PALGRAVE, in the series Cornerstones of Computing, 2000, pages 91-98.
36. The Roots of Object Orientation: the Simula Language. O.-J. Dahl. *In: Software Pioneers: Contribution to Software Engineering.* Manfred Broy, Ernst Denert (eds.), Springer Verlag, 2002. pages 78-90. (Talk given at conference in Bonn, June 2001) ISBN 3-540-43081-4

Reports from University of Oslo

Early reports are from Department of Mathematics, section D (for numerical analysis and computer science). Later reports are from Department of

Informatics (IFI), which was formed in 1978. Reports marked by a star (*) represent early versions or revisions of papers mentioned above.

Research Reports

37. Analysis of an Algorithm for Priority Queue Administration. Arne Jonassen and O.-J. Dahl. Research Report in Informatics no. 3, 1975 (68 pages) ISBN 82-553-0203-4 *
38. An Approach to Correctness Proofs of Semicoroutines. O.-J. Dahl. Research Report in Informatics no. 13, 1977 (20 pages) ISBN 82-90230-00-1 *
39. Can Program Proving be Made Practical? O.-J. Dahl. Research Report in Informatics no. 33, 1978 (57 pages) Lectures presented at the EEC-CREST course on Programming Foundations, Toulouse 1977 (revised May 1978). ISBN 82-90230-26-5 *
40. Time Sequences as a Tool for Describing Program Behaviour. O.-J. Dahl. Research Report no. 48, 1979 (17 pages) ISBN 82-90230-43-5 *
41. Partial Correctness Semantics of Communicating Sequential Processes. Neelam Soundararajan and O.-J. Dahl. Research Report no. 66, 1982 (29 pages) ISBN 82-90230-62-1
42. Notes on a LIFO Disciplined Simplex Algorithm. O.-J. Dahl. Research Report no. 79, 1984 (7 pages) ISBN 82-90230-76-1
43. Logic of Programming and Specification. O.-J. Dahl. Research Report no. 84, 1984 (48 pages) ISBN 82-90230-83-4
44. A Presentation of the Specification and Verification Project ABEL. O.-J. Dahl and O. Owe. Research Report no. 90, 1984 (9 pages) ISBN 82-90230-89-3. Also in the proceedings of the 3. Verification Workshop, Watsonville, CA, Feb. 1985.
45. Specification and Reasoning about Discrete Simulation Models: A Case Study. O.-J. Dahl. Research Report no. 94, 1985 (10 pages) ISBN 82-90230-93-1. Lecure at the IMACS World Congress on System Simulation and Scientific Computation, Oslo, Aug. 1985.
46. Preliminary Report on the Specification and Programming Language ABEL. O.-J. Dahl, Dag F. Langmyhr, O. Owe. Research Report no. 106, 1986 (86 pages) ISBN 82-7368-006-1
47. Object Oriented Specification. O.-J. Dahl. Research Report no. 108, 1987 (18 pages) ISBN 82-7368-009-6 *
48. Generator Induction in Order Sorted Algebras. O.-J. Dahl and O. Owe. Research Report no. 122, 1989 (17 pages) ISBN 82-7368-027-4 *
49. Object Orientation and Formal Techniques (extended abstracts). O.-J. Dahl. Research Report no. 138, 1990 (11 pages) ISBN 82-7368-044-4 *
50. Formal Development with ABEL. O.-J. Dahl and O. Owe. Research Report no. 159, 1991 (43 pages) ISBN 82-7368-066-5 *
51. Value Types and Object Classes. O.-J. Dahl. Talk presented at the 25 years SIMULA 67 Anniversary Conferences, Nordwijkerhout 11-12 June, 1992 and Oslo 22 June. Research Report no. 170, 1992 (13 pages) ISBN 82-7368-079-7 *

52. Monitors Revisited. O.-J. Dahl. Research Report no. 175, 1993 (12 pages) ISBN 82-7368-084-3 *
53. Co- and Contravariance in Functional Subtypes: Contribution to IFIP WG 2.3, June 1994. O.-J. Dahl. Research Report no. 191, 1994 (7 pages) ISBN 82-7368-101-7
54. Hoare-style Parallel Programming: Foils for a student course, IN305. O.-J. Dahl. Research Report no. 192, 1994 ISBN 82-7368-102-5
55. On the Use of Subtypes in ABEL. O.-J. Dahl and O. Owe. Revised version. Research Report no. 206, 1995 (20 pages) ISBN 82-7368-117-3 *
56. On Introducing Higher Order Functions in ABEL. O.-J. Dahl and Bjørn Kristoffersen. Research Report no. 210, 1995 (18 pages) ISBN 82-7368-123-8 *
57. Subtyping and Constructive Specification. O.-J. Dahl, O. Owe and Tore J. Bastiansen. Research Report no. 228, 1996 (38 pages) ISBN 82-7368-142-4 *
58. Formal Methods and the RM-ODP. O.-J. Dahl and O. Owe. Research Report no. 261, 1998 (17 pages) ISBN 82-7368-192-0

Compendiums and Lecture Notes

59. Kompendium til DB-2 (maskinspråk og operasjonssytem). O.-J. Dahl. Dept. of Mathematics, 1969 (48 pages) (In Norwegian).
60. Listestrukturer: Ch. 7 from compendium Ikke-numeriske metoder. O.-J. Dahl. Dept. of Mathematics, 1970 (46 pages) (In Norwegian).
61. Forelesninger i DB 2. O.-J. Dahl & co. 2. edition, Dept. of Mathematics, 1971 (171 pages) (In Norwegian).
62. Top-Down Parsers Expressed in a High-Level Language. O.-J. Dahl. Dept. of Mathematics, 1972 (12 pages).
63. Hierarchical Program Structures. O.-J. Dahl. Lecture Note no. 6, Dept. of Mathematics, 1973 (57 pages). The predecessor of [1].
64. Two Lectures for the Graduate Course: Parallel Programming and Operating Systems. O.-J. Dahl and C.A.R. Hoare. Lecture Note no. 7, Dept. of Mathematics, 1973 (15 pages).
65. Runtime organisasjon for SIMULA/ALGOL. O.-J. Dahl. Compendium no. 11. Dept. of Informatics, 1980 (36 pages) (In Norwegian).
66. Stochastiske simuleringsmodeller. O.-J. Dahl. (36 pages) Compendium 13, Dept. of Informatics (In Norwegian).
67. Program Specification and Verification Techniques. O.-J. Dahl. Part I, 1988, Part II and III 1990. Compendium 42, Dept. of Informatics (153 pages).
68. Substitusjonssystemer. O.-J. Dahl. Lecture notes (12 pages) Aug. 1989 (In Norwegian)
69. Parallell programmering (Parallel Programming). O.-J. Dahl. Compendium 45, Dept. of Informatics (69 pages), Revised Aug. 1995 (In Norwegian). Kompendium 46, Aug. 1996 (In Norwegian).
70. Stochastic Simulation Modelling. O.-J. Dahl. Compendium 69, Dept. of Informatics, April 1996 (36 pages).

Reports from the Norwegian Computing Center (NCC), Oslo

71. Preliminary Presentation of the Simula Language and some Examples of Network Descriptions. O.-J. Dahl and K. Nygaard. NCC Doc., May 18th 1963.
72. SIMSCRIPT Implementation. Vic Bell and O.-J. Dahl. NCC Doc. (31 pages), Nov. 1963.
73. The SIMULA Storage Allocation Scheme. O.-J. Dahl. NCC Doc. 162, Nov. 1963 (9 pages)
74. SIMULA Status Report. O.-J. Dahl. NCC Doc. 1.1, 1964 (10 pages)
75. The SIMULA Data Structures. O.-J. Dahl. NCC Doc. March 1964 (23 pages)
76. The SIMULA Language: Specifications 17 March 1964. O.-J. Dahl and K. Nygaard. NCC Doc. March, 1964 (30 pages)
77. The SIMULA Project: Technical Progress Report 1. O.-J. Dahl and K. Nygaard. NCC Doc. July 1, 1964 (7 pages)
78. SIMULA status report. O.-J. Dahl. NCC Doc. 1.10, 1964 (24 pages)
79. SIMULA – A Language for Programming and Description of Discrete Event Systems: Introduction and User's Manual. O.-J. Dahl and K. Nygaard. NCC Publ. no. 11, May 1965 (103 pages). Revised versions: 1966. III, 108 pages. 5th ed. NCC, 1967 (124 pages)
80. Basic Concepts of SIMULA – an ALGOL Based Simulation Language. O.-J. Dahl and K. Nygaard. NCC Doc., 1965 (17 pages)
81. SIMULA, an ALGOL Based Simulation Language. O.-J. Dahl and K. Nygaard. NCC Doc., April 1966 (26 pages) *
82. Discrete Event Simulation Languages: Lectures Delivered at the NATO Summer School, Villard-de-Lans, Sept. 1966. O.-J. Dahl. NCC Doc., 1966 (63 pages) *
83. SIMULA: Simula Tracing System. O.-J. Dahl, B. Myhrhaug and K. Nygaard. NCC Doc., 1966.
84. Class and Subclass Declarations. O.-J. Dahl and K. Nygaard. NCC Publ. no. 93, (Presented at IFIP Working Conference on Simulation Languages, Lysebu, Oslo, May 1967). March 1967 (17 pages) *
85. SIMULA 67 Common Base Proposal. O.-J. Dahl and K. Nygaard. NCC Doc., May 1967 (10 pages)
86. Proposals for Consideration by the SIMULA 67 Common Base Conference. O.-J. Dahl and K. Nygaard. NCC Doc., June 1967.
87. SIMULA 67 Common Base Definition. O.-J. Dahl and K. Nygaard. NCC Doc., June 1967 (31 pages).
88. SIMULA 67 Common Base Language. O.-J. Dahl, B. Myhrhaug, K. Nygaard. NCC Publ. S-2, 1968 (141 pages).
    Revised editions: (1970, 145 pages, SIMULA information: NCC Publ. S-22). (1982, 127 pages, NCC Publ. no. 725). (1984, 172 pages, NCC Publ. no. 743). ISBN 82-539-0225-5 [2]

[2] All versions of the common base language definition assume knowledge of Algol 60. A final and complete version of the SIMULA 67 language definition without this

89. Some Uses of the External Class Concept in SIMULA 67. O.-J. Dahl, B. Myhrhaug, K. Nygaard. NCC Doc., 1968. (Presented at the NATO sponsored conference on Software Engineering, Garmisch, Germany, October 1968)
90. SIMULA 67 – Basic Information. O.-J. Dahl and K. Nygaard. NCC Publ. no. S-3, 1968 (12 pages).
91. SIMULA 67 – Implementation Guide. O.-J. Dahl, B. Myhrhaug. NCC Publ. no. S-9, 1969. Rev. March 1973, NCC Publ. no. S-47 (146 pages).
92. The Development of the SIMULA Languages. K. Nygaard and O.-J. Dahl. NCC Doc. (Publication 603), 1978 (28 pages) ISBN 82-539-0072-4 (Note: includes a bibliography.) *

Reports from the Norwegian Defense Research Establishment (NDRE), Kjeller, Norway

93. An Automatic Coding Scheme for the Ferranti MERCURY Computer. O.-J. Dahl. NDRE Report IR-F-286, 1956.
94. Multiple Index Countings on the Ferranti MERCURY Computer. O.-J. Dahl. Norwegian Defense Research Establishment, 1957. (NDRE Report 23) & Cand real thesis, University of Oslo, Dept. of Mathematics, 1957.
95. Programmer's Handbook for the Ferranti MERCURY Computer Frederic at the Norwegian Defense Research Establishment. O.-J. Dahl, Jan V. Garwick. NDRE. 2. edition published by Merkantile Trykkeri, Oslo, 1958. 3. edition, NDRE, 1962 (117 pages).
96. Mac Bulletin I. O.-J. Dahl. NDRE report (S-15), 1960 (10 pages)[3]
97. Mac Bulletin II. O.-J. Dahl. NDRE report (S-17), 1960 (3 pages)
98. Mac Bulletin V: Macros. O.-J. Dahl NDRE report (S-57), 1963 (13 pages)

Video Recorded Talks

99. ACM SIGPLAN History of Programming Languages Conference, January 1978, Los Angeles. SIMULA Session. Speakers: Nygaard and Dahl. 1 videotape (VHS) (60 min), ACM, 1980.
100. The talk given at Software Pioneers Conference, Bonn, June 2001 (supplied with the book [36] above).
101. Introduction to SIMULA (talk given in Norwegian), to be available through the department homepage: `www.ifi.uio.no`.

---

assumption is found in: Standard SIMULA, as approved by the SIMULA Standards Group Aug. 26, 1986. (176 pages) ISBN 91-7162-234-9.

[3] MAC was an acronym for MERCURY Automatic Coding, a high level programming language for the MERCURY computer, and a compiler.