



Dahl and Nygaard at the time of Simula's development

Simula '67

- Class and object. The class concept as a template for creating instance – objects.
- Subclass. Classes may be organized in a classification hierarchy by means of subclasses.
- Virtual methods. A Simula class may define virtual methods that can be redefined in subclasses.
- Active objects. An object in Simula may be the head of an active thread – technically it is a coroutine.
- Action combination. Simula has an inner-construct for combining the action-parts of a class and its subclass.
- Processes and schedulers. It is easy in Simula to write new concurrency abstractions including schedulers.
- Frameworks. Simula provided the first OO framework in form of Class Simulation. The simulation features of Simula I was made available through Class Simulation.
- Automatic memory management. Simula had automatic memory management, including garbage collection.



Ole-Johan Dahl

The Birth of Object Orientation: the Simula Languages

June 2001

"A major new impact area opened with the introduction of workstations and personal computers. Alan Kay and his team at Xerox PARC developed Smalltalk, [15], an interactive language building upon Simula's objects, classes and inheritance. It is oriented towards organising the cooperation between a user and her/his personal computer."



Alan Kay

"If you look at software today, through the lens of the history of engineering, it's certainly engineering of a sort—but it's the kind of engineering that people without the concept of the arch did. Most software today is very much like an Egyptian pyramid with millions of bricks piled on top of each other, with no structural integrity, but just done by brute force and thousands of slaves."



Adele Goldberg

"During the PARC days, the opportunity to work with children and other nontechnical users kept us focused on how to use rigorously what people already know informally about objects. But the most thrilling experience for me was to work with ParcPlace customers in both large and small companies, and see how our technology enabled them to finally break the barrier between business understanding and systems implementation."



Kuhn, Thomas S.

The Structure of Scientific Revolutions (3rd. Edition)
(The University of Chicago Press, 1962, 1970, 1996)

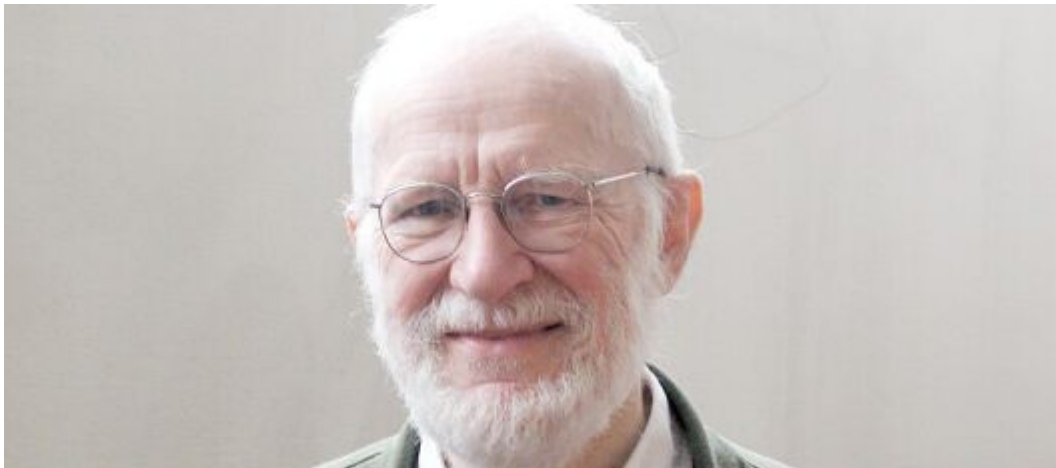
“...in much of the book the term “paradigm” is used in two different senses. On the one hand, it stands for the entire constellation of beliefs, values, techniques, and so on shared by the members of a given community. On the other, it denotes one sort of element in that constellation, the concrete puzzle-solutions which, employed as models or examples, can replace explicit rules as a basis for the solution of the remaining puzzles of normal science.”

Postscript – 1969. Page 175.



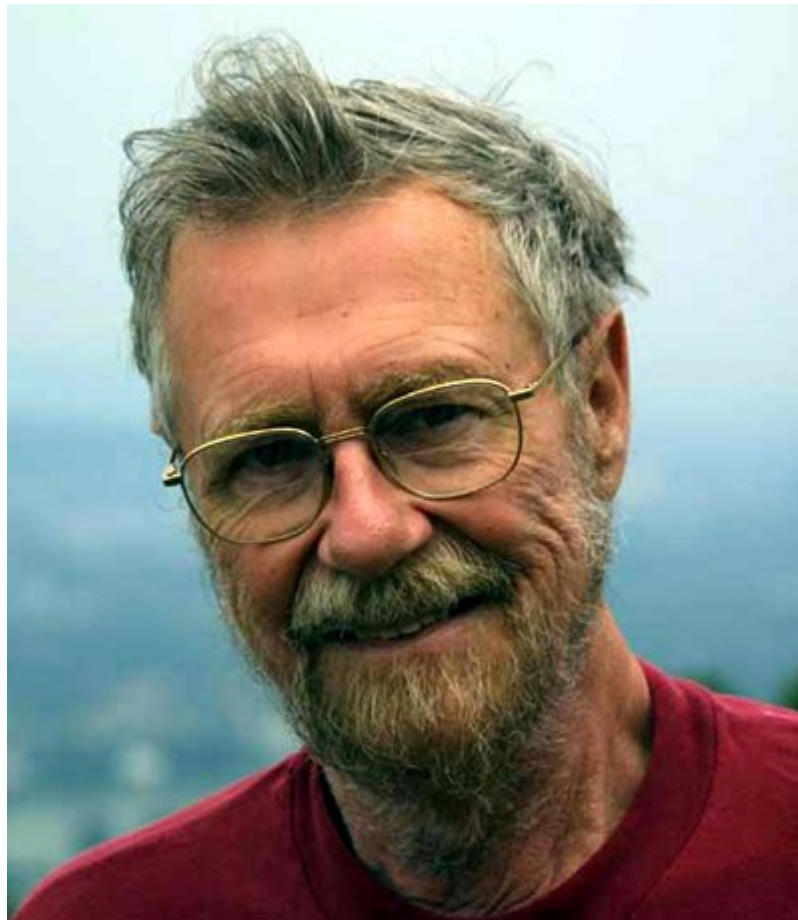
Kristen Nygaard

All these three perspectives should be supported within any general system description and programming language in the future: no perspective will "win" as some people seem to believe. (For this reason the term "paradigm" should be avoided, since the meaning established by Thomas Kuhn for the term is that of a basic perspective within a science, irreconcilable with alternative paradigms. [Kuhn, T., 1970])



Peter Naur

As a major result of these studies I described programming as a human activity: theory building [27-1985]. By this description the core of programming is the programmer's developing a certain kind of understanding of the matters of concern.



Edsger Wybe Dijkstra

Finally: from my words some of you may have concluded that I am just spiteful and bitter. Let me reassure you: I am neither spiteful, nor bitter. Not yet although I must admit that the major part of computing Science (or perhaps more accurately: Programming Folklore) often strikes me as unchallenged prejudices, repeated over and over again as articles of faith. But in my introduction I have announced that I should say what I wanted to say. I know that I have used strong language and harsh words - but the time to be gentle has passed: the situation is much too serious to be covered by politeness.

The floor is now open for discussion and I thank you for your patience.

Eindhoven, dec. 1969.

