

Ernst-Rüdiger Olderog: A Life for Meaning

André Platzer^(✉)

Computer Science Department, Carnegie Mellon University, Pittsburgh, USA
aplatzer@cs.cmu.edu

While it is indubitably impossible to summarize all the many important aspects of Ernst-Rüdiger Olderog’s scientific contributions and to do justice to each and every one of the areas that he contributed to, it is remarkably easy to identify and characterize the common core behind his investigations. An important leitmotif in his research agenda is *semantics*, the study of meaning.

Even if a significant share of his research does not study semantics in and of itself, but rather as a means to an end, Ernst-Rüdiger Olderog stands out as having recognized the significance of semantics as an important foundation in the first place. He accurately observes how crucial the study of the meaning of a mathematical object of study is for the progress of a research area as a whole, as well as for achieving the individual results he is looking for. The most prominent results that Ernst-Rüdiger Olderog’s research enables are various forms of correctness results, as are engraved in the name of his research group: “Correct System Design”. The main analytic tool to obtain them, though, is Ernst-Rüdiger Olderog’s dedication to semantics.¹ Be it to develop an understanding of concurrency by logical communication formulas to algebraic process terms and further to Petri nets [21], to understand sequential and concurrent programs for the purpose of verification [3,2], or specification and verification techniques for real-time systems [26]. A strong devotion to semantics is a pervasive aspect of his research throughout his career [27,4,28,5,21,3,22,12,26,11], whether on communicating processes [28,27], concurrency at large [5,4,21], real-time systems [26,22] and specification languages for richer realtime systems [15,12], traffic agents [9,8], or games for distributes systems synthesis [11]. Ernst-Rüdiger Olderog’s attention to semantics is quite prominently featured already in the first column of a paper on his first research results [20]:

To explore the applicability of this idea, we have to formalize what “capturing the true partial correctness formulas” means. [20]

In that paper he investigates completeness of Hoare calculi and limits thereof for programs with procedures [20,19] based on his dissertation [18], refining earlier results of Clarke [7]. It is quite an achievement for a proper treatment of semantics to span such a wide range of topics, each with its own different conceptual and technical challenges.

Most of his Ph.D. students got accustomed to Ernst-Rüdiger Olderog’s quest for semantics quite quickly. So quickly that there was ample opportunity to overhear the following question among the first raised during many random hallway

¹ As reflected in the former name of Ernst-Rüdiger Olderog’s group: “Semantics”.

conversations among students about their most recent (sometimes ingenious but certainly still immature) discovery:

“*Was soll denn das heissen???*”

which roughly translates to: “But what’s that even supposed to mean???”

Of course, Ernst-Rüdiger Olderog himself used an infinitely more polite and modest way of rephrasing this question, albeit in a semantics-preserving way. But by carefully thought-out remarks and polite questions, Ernst-Rüdiger made sure his students ultimately understand that everything else crumbles apart unless the semantics holds it all neatly together. It takes the right semantics to start off a scientific development. If the semantics is inappropriate, the best theorems about it do not help. If the semantics expresses the right aspects, but exposes them in inapt ways, then proofs about the semantics become tedious.

Ernst-Rüdiger Olderog does not just ask whether a semantics is “right”. He asks whether it is “exactly right”. Does the semantics fit to the intuitive expectations for the domain of discourse? Does it lend itself to extrapolating appropriate verification and reasoning techniques? Does it pass muster under the respective principles of semantics, such as compositionality principles in denotational semantics or progress properties in operational semantics? Does it lead to simple and elegant proofs? A semantics that is exactly right will satisfy all these criteria, which leads to a spectrum of quite finely nuanced semantical choices that the uninitiated might struggle with.

Indeed, Ernst-Rüdiger’s focus on getting the semantics right also makes all the sense in the world in another context. Higher-order proof assistants such as Isabelle [17], Coq [16], and Nuprl [1] make it possible to develop rich theories with machine-checked proofs. Since that makes mistakes in proofs impossible (if the prover kernel is implemented correctly), getting the semantics right as reflected in the basic definitions themselves is what probably matters the most.

Ernst-Rüdiger Olderog deserves particular admiration for his dauntless investigation of semantical nuances in ever more demanding challenges. Rather than settling for the (already very challenging) world of concurrent processes [21], he went on to a semantic study of programs [3,2], even concurrent programs, with all their extremely subtle semantic interactions. In shared-variable parallel programs and synchronization in parallel programs, a fine line separates a permissive semantics that allows all kinds of concurrent interactions but makes analysis and predictions impossible from a semantics that makes verification straightforward but is so overly restrictive that it hardly does justice to the intended purposes of concurrent programs. Rather than settling for the challenges of concurrency, however, Ernst-Rüdiger went on to study the semantical challenges of real-time systems [26], and did not shy away from subtle and challenging integrations of real-time reasoning with concurrency and rich data structures [15].

These directions of real-time systems as well as of semantically integrating real-time systems with complex systems aspects were one instrumental part of the AVACS project *Automatic Verification and Analysis of Complex Systems* [6]. Ernst-Rüdiger Olderog played a major role in AVACS, where he has been serving as the coordinator for the whole Real-Time Systems group R as well as for one

of its 3 subprojects on real-time systems with complex aspects such as rich data [23] ever since its beginning in 2004 until today. AVACS is a Sonderforschungsbereich/Transregio funded by the German Science Foundation (DFG) spanning real-time systems (the R subprojects), hybrid systems (H), and systems of systems verification (S) in a major research initiative of researchers at the University of Oldenburg, the University of Freiburg, the University of Saarbrücken as well as the Max-Planck Institute in Saarbrücken. AVACS involved around 10 subprojects, each with around 5 principle investigators and even more researchers.

While real-time systems have received significant attention by the research community at large, Ernst-Rüdiger Olderog noticed early on that real-time systems are not just important in isolation but that real-time aspects arise in systems with a proper software structure [29,24,30,12,15,25,26,23]. These considerations include transformations from real-time specifications to program specifications [29], decompositions of Duration Calculus formulas into untimed systems communicating by timers [24], and a verification and design approach for real-time software in Programmable Logic Controllers (PLCs) [22], which was held together by a backbone of a semantic link via Duration Calculus between Constraint Diagrams and PLCs, and which had applications in railway signaling systems. This approach has been implemented in the verification tool Moby/PLC [10], which led to Moby/RT [25], a more general verification tool for real-time systems with a similar basis. Ernst-Rüdiger Olderog pursued his interest in combinations of specification techniques further in the development of CSP-OZ-DC [15], an integration of Communicating Sequential Processes with Object-Z and Duration Calculus, which formed an important foundation for the real-time efforts throughout AVACS to understand the interaction of concurrent process behavior, infinite data structures, and continuous real-time dynamics [23].

In addition to serving as scientific director for the real-time efforts in AVACS, Ernst-Rüdiger Olderog also contributed to efforts on the hybrid systems side, especially in verification efforts for traffic agents obeying certain cooperation principles, both in railway systems [8] as well as in car platooning [9]. The basic observation that made an analysis more feasible, was that such traffic applications can often be simplified by partitioning its operations into phases where traffic agents are still far away, then when they come closer and need to negotiate a safe action, as well as when they are correcting their actions to avoid or mitigate safety risks. Of course, the corresponding verification principles are backed up by a detailed semantical analysis how, generically, the components of such a system can interact to justify its correctness. Careful considerations exploiting the abstract structure of traffic also led to a dedicated logic for multilane scenario reasoning in cars by abstracting the motion qualitatively in spatial interval logic [14,13] again, of course, with a dedicated semantics suitable for the application domain.

I cannot say for sure whether Ernst-Rüdiger Olderog found the meaning of life yet, but with all his dedication to giving it all a semantics, I am confident that he is leading a life for meaning. Congratulations, Ernst-Rüdiger Olderog!

References

1. Allen, S.F., Constable, R.L., Eaton, R., Kreitz, C., Lorigo, L.: The nuprl open logical environment. In: McAllester, D. (ed.) CADE-17. LNCS, vol. 1831, pp. 170–176. Springer, Heidelberg (2000). http://dx.doi.org/10.1007/10721959_12
2. Apt, K.R., de Boer, F.S., Olderog, E.R.: Verification of Sequential and Concurrent Programs, 3rd edn. Springer (2010)
3. Apt, K.R., Olderog, E.R.: Verification of Sequential and Concurrent Programs. Texts and Monographs in Computer Science, 2nd edn. Springer (1997)
4. de Bakker, J.W., Meyer, J.C., Olderog, E., Zucker, J.I.: Transition systems, infinitary languages and the semantics of uniform concurrency. In: Proceedings of the 17th Annual ACM Symposium on Theory of Computing, Providence, Rhode Island, USA, May 6–8, pp. 252–262 (1985)
5. de Bakker, J.W., Meyer, J.C., Olderog, E., Zucker, J.I.: Transition systems, metric spaces and ready sets in the semantics of uniform concurrency. *J. Comput. Syst. Sci.* **36**(2), 158–224 (1988)
6. Becker, B., Podelski, A., Damm, W., Fränzle, M., Olderog, E., Wilhelm, R.: SFB/TR 14 AVACS - automatic verification and analysis of complex systems (der sonderforschungsbereich/transregio 14 AVACS - automatische verifikation und analyse komplexer systeme). *IT - Information Technology* **49**(2), 118–126 (2007)
7. Clarke, E.M.: Programming language constructs for which it is impossible to obtain good Hoare axiom systems. *J. ACM* **26**(1), 129–147 (1979)
8. Damm, W., Hungar, H., Olderog, E.-R.: On the verification of cooperating traffic agents. In: de Boer, F.S., Bonsangue, M.M., Graf, S., de Roever, W.-P. (eds.) FMCO 2003. LNCS, vol. 3188, pp. 77–110. Springer, Heidelberg (2004)
9. Damm, W., Hungar, H., Olderog, E.R.: Verification of cooperating traffic agents. *International Journal of Control* **79**(5), 395–421 (2006)
10. Dierks, H., Tapken, J.: Tool-supported hierarchical design of distributed real-time systems. In: 26th Euromicro Conference on Real-Time Systems. IEEE Computer Society, Los Alamitos (1998)
11. Finkbeiner, B., Olderog, E.: Petri games: synthesis of distributed systems with causal memory. In: Proceedings Fifth International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2014, Verona, Italy, September 10–12, pp. 217–230 (2014)
12. Fischer, C., Olderog, E.-R., Wehrheim, H.: A CSP view on UML-RT structure diagrams. In: Hussmann, H. (ed.) FASE 2001. LNCS, vol. 2029, pp. 91–108. Springer, Heidelberg (2001)
13. Hilscher, M., Linker, S., Olderog, E.-R.: Proving safety of traffic manoeuvres on country roads. In: Liu, Z., Woodcock, J., Zhu, H. (eds.) Theories of Programming and Formal Methods. LNCS, vol. 8051, pp. 196–212. Springer, Heidelberg (2013)
14. Hilscher, M., Linker, S., Olderog, E.-R., Ravn, A.P.: An abstract model for proving safety of multi-lane traffic manoeuvres. In: Qin, S., Qiu, Z. (eds.) ICFEM 2011. LNCS, vol. 6991, pp. 404–419. Springer, Heidelberg (2011)
15. Hoenicke, J., Olderog, E.: CSP-OZ-DC: A combination of specification techniques for processes, data and time. *Nord. J. Comput.* **9**(4), 301–334 (2002)
16. The Coq development team: The Coq proof assistant reference manual. LogiCal Project (2004). <http://coq.inria.fr> (version 8.0)
17. Nipkow, T., Wenzel, M., Paulson, L.C.: Isabelle/HOL: A Proof Assistant for Higher-order Logic. Springer, Heidelberg (2002)

18. Olderog, E.R.: Charakterisierung Hoare-scher Systeme für Algol-ähnliche Programmiersprachen. Ph.D. thesis, Universität Kiel (1981)
19. Olderog, E.: Sound and complete Hoare-like calculi based on copy rules. *Acta Inf.* **16**, 161–197 (1981)
20. Olderog, E.: A characterization of Hoare’s logic for programs with pascal-like procedures. In: *Proceedings of the 15th Annual ACM Symposium on Theory of Computing*, Boston, Massachusetts, USA, April 25–27, pp. 320–329 (1983)
21. Olderog, E.R.: *Nets, Terms and Formulas: Three Views of Concurrent Processes and Their Relationship*. Cambridge University Press (1991)
22. Olderog, E.-R.: Correct real-time software for programmable logic controllers. In: Olderog, E.-R., Steffen, B. (eds.) *Correct System Design*. LNCS, vol. 1710, pp. 342–362. Springer, Heidelberg (1999)
23. Olderog, E.-R.: Automatic verification of real-time systems with rich data: an overview. In: Agrawal, M., Cooper, S.B., Li, A. (eds.) *TAMC 2012*. LNCS, vol. 7287, pp. 84–93. Springer, Heidelberg (2012)
24. Olderog, E.-R., Dierks, H.: Decomposing real-time specifications. In: de Roever, W.-P., Langmaack, H., Pnueli, A. (eds.) *COMPOS 1997*. LNCS, vol. 1536, pp. 465–489. Springer, Heidelberg (1998)
25. Olderog, E., Dierks, H.: Moby/RT: A tool for specification and verification of real-time systems. *J. UCS* **9**(2), 88–105 (2003)
26. Olderog, E.R., Dierks, H.: *Real-Time Systems: Formal Specification and Automatic Verification*. Cambridge Univ. Press (2008)
27. Olderog, E., Hoare, C.A.R.: Specification-oriented semantics for communicating processes. In: Diaz, J. (ed.) *Automata, Languages and Programming*. LNCS, vol. 154, pp. 561–572. Springer, Heidelberg (1983)
28. Olderog, E., Hoare, C.A.R.: Specification-oriented semantics for communicating processes. *Acta Inf.* **23**(1), 9–66 (1986)
29. Olderog, E.R., Schenke, M.: Design of real-time systems: the interface between duration calculus and program specifications. In: Desel, J. (ed.) *Structures in Concurrency Theory*, pp. 32–54. WICS. Springer (1995)
30. Schenke, M., Olderog, E.: Transformational design of real-time systems part I: from requirements to program specifications. *Acta Inf.* **36**(1), 1–65 (1999)

Correct System Design

Symposium in Honor of Ernst-Rüdiger Olderog on the
Occasion of His 60th Birthday, Oldenburg, Germany,
September 8-9, 2015, Proceedings

Meyer, R.; Platzer, A.; Wehrheim, H. (Eds.)

2015, XII, 279 p. 69 illus. in color., Softcover

ISBN: 978-3-319-23505-9