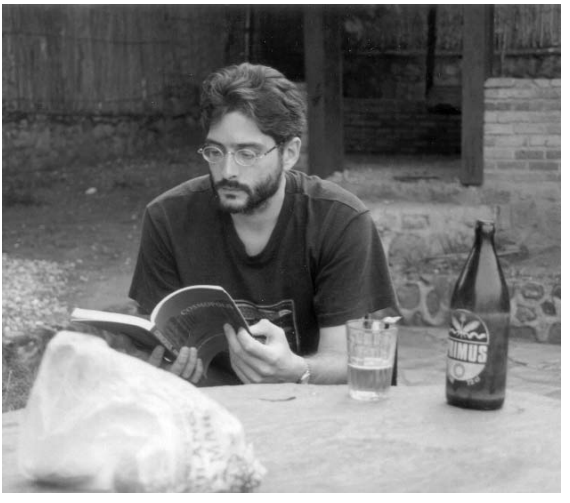


Kijereshi, Tanzania, 1994

HET KOMT WEL VOOR

"Gedaan in het gesloten
seizoen der dromen"
Lucebert

het komt wel voor
dat de deur in het slot valt
de stilte plotseling intreedt
de angst toeslaat
men in een leegte staart –

het komt wel voor
dat een man van in de veertig
met ontstelde ogen toeziet
hoe de vingers zich in elkaar
wringen, voelen moet hoe de
adem stokt in de keel –

het komt wel voor
dat men dan de dood ontmoet

Wim Gerth
28 November 2003

Rob Tristan Gerth: 1956–2003

On Friday November 28, 2003, computer scientist and logician Rob Gerth died from sudden cardiac arrest. It is impossible to say all that the loss of Rob means to Intel, to the verification community, and to each of us personally.

As a corporation, Intel has lost a world-renowned authority on formal modeling and verification. Rob brought an independence of thought and agility of mind that are the hallmarks of a rare intellect. He was one of the few people with the knowledge and experience to meet the challenges we face in verifying micro-architectural protocols; we hope we can find a way to carry on his legacy.

As a community, we have lost not just a key technical contributor but part of our scientific conscience. It was in Rob's nature to ask "Why?", even when the answer seemed obvious. He brought a commitment to truth and high standards that was alternately challenging, frustrating, and inspiring; we will sorely miss it.

As family, friends and colleagues, we will remember and cherish Rob's warmth and love, his knowledge and openness in sharing it, his dry sense of humor, the way only he could read Toon Tellegen stories, and – perhaps most of all – his contagious laughter.

Rob's passion for knowledge reached far beyond computer science and logic: it embraced science, history, literature, art and music. The 2004 Proceedings of the 16th International Conference on Computer Aided Verification are dedicated to him. This article in the Proceedings surveys highlights of Rob's scientific career, followed by excerpts taken from some of the many tributes sent for Rob's funeral by friends and colleagues from around the world. The list of contributors, though too long to be printed here, is itself a tribute to Rob's impact on us all.

May 2004

John O'Leary & Marly Roncken
Intel

R. Alur and D.A. Peled (Eds.): CAV 2004, LNCS 3114, pp. 1–14, 2004.

Rob Gerth, Program Verifier par Excellence: His Research

Going through Rob Gerth's contributions to program semantics and verification, and tools for the latter, one is struck by his almost incredible versatility, total commitment, and scientific depth.

Starting in 1981 with a proof system for Brinch Hansen's monitor-based language Distributed Processes [1], he followed with an important paper on "Transition Logic: How to Reason about Temporal Properties in a Compositional Way" [2], coauthored papers on "Compositional Semantics for Real-Time Distributed Computing" for the predecessor of LICS [3], with its Full-Abstraction proof [4], a (sound and relatively complete) "Proof System for Concurrent Ada Programs" in *Science of Computer Programming* [5], and a far reaching paper in 1986 "Monitors Revisited: A First Step towards Verifying Object-Oriented Systems" [6], far before the general research public in computer science realized the importance of concepts like compositionality, formalized real-time semantics, or relatively complete proof systems for monitor-based and object-oriented languages.

This list continues to be extended in the same break-through baffling fashion: Coauthor of full-abstract semantics for Statecharts and concurrent Prolog [7, 8], "Rooting UNITY" [9], his Program Refinement and Self-Modifying Code [10, 11], coauthored papers for CONCUR [12, 13] or Reliable Distributed Systems [14]. Who else in those years cared for investigating fully abstract program semantics, verifying fault-tolerant distributed programs, self-modifying programs, interface refinement, and the now crucial concept of compositional refinement?

Having reached the limits of what can reasonably be verified *in principle* by hand, Rob saw that the time of automated verification had arrived and devoted himself with the same ruthless interest and intrepidity to developing foundations of abstract interpretation, model checking and automated refinement tools – to allow applications at industrial scale. The focus in his work now shifts to handling state explosion in (real-time) model checking.

To this end, he developed partial-order techniques [15, 16] and other advanced techniques like on-the-fly model checking [17] and abstraction [18] – the latter with his "very own" Ph.D. student Dennis Dams and with Orna Grumberg. The foundational aspects of abstract interpretation in model checking were first presented at a memorable PROCOMET workshop in San Miniato [19], and culminated in 1997 in a TOPLAS seminal paper [20] and application paper [21].

At the end of this period, Rob organised the ESPRIT project VIRES (Verifying Industrial Reactive Systems) where the developed theory and tools were applied to the verification of an industrial bus protocol. This project was his legacy to Europe, and the beginning of a new era in the verification of large systems. When Rob left the Technical University Eindhoven for Intel at the end of 1997, VIRES found a suitable new project lead in his former Ph.D student Dennis Dams.

Rob's "warming up" exercise at Intel was the ambitious exploratory project of formally verifying arithmetic hardware units to their IEEE specifications [22]. The technology and methodology developed in that project are used routinely

today to verify floating-point functionality in Intel's Microprocessor design projects.

He subsequently went back to a more familiar domain, developing a linear-time temporal logic – dubbed ForSpec [23]. The uniqueness of Forspec lies in its many special features for hardware specification, motivated by many Intel engineering years of both edge-cutting and practical design and validation. Rob's unique blend of theoretical knowledge and practical experience in formal semantics and model checking invariably prevented inconsistencies and efficiency hazards, and helped solve the most difficult roadblocks. ForSpec is widely used in Intel, and has been incorporated in OpenVera^{TM}, a language for hardware testbenches and assertions that is supported by a broad network of CAD companies.

Just weeks before his death, Rob presented the first results of what he called his "bit-vector compiler" – software he had been developing to relate protocol models over abstract data types to ditto models over "hardware" datatypes, i.e. bits and bit-vectors. Following up on [24], the compiler is part of Rob's protocol verification programme and perhaps the most cherished and ambitious project of his life. A life which he did not anticipate would end any time soon.

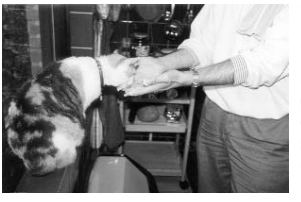
If Edsger Wiebe Dijkstra, another Dutchman, opened the eyes of computer scientists for the value of proving programs correct, then in Rob Tristan Gerth a worthy modern-day successor and polymath was incarnated, whose own unique vision furthered the feasibility of Dijkstra's programme. We shall all miss Rob's enthusiasm, ruthless intellectual curiosity, and relentless energy.

Selected List of Publications

1. M. Roncken, R. Gerth, and W. P. de Roever. A proof system for Brinch Hansen's distributed processes. In *Proceedings of the GI Jahrestagung*, pages 88–95, 1981.
2. R. Gerth. Transition logic: How to reason about temporal properties in a compositional way. In *16th ACM Symp. on Theory of Computing*, pages 39–50. ACM Press, 1984.
3. R. Koymans, R. K. Shyamasundar, W. P. de Roever, R. Gerth, and S. Arun-Kumar. Compositional semantics for real-time distributed computing. In R. Parikh, editor, *Logic of Programs*, volume 193 of *LNCS*, pages 167–189. Springer, 1985.
4. C. Huizing, R. Gerth, and W. P. deRoever. Full abstraction of a real-time denotational semantics for an OCCAM-like language. In *14th ACM Symp. on Principles of Programming Languages*, pages 223–236. ACM Press, 1987.
5. R. Gerth and W. P. de Roever. A proof system for concurrent Ada programs. *Science of Computer Programming*, 4(2):159–204, 1984.
6. R. Gerth and W. P. de Roever. Proving monitors revisited: a first step towards verifying object oriented systems. *Fundamenta Informatica*, 9(4):371–399, 1986.
7. C. Huizing, R. Gerth, and W. P. de Roever. Modeling statecharts behaviour in a fully abstract way. In M. Dauchet and M. Nivat, editors, *13th Colloq. on Trees in Algebra and Programming*, volume 299 of *LNCS*, pages 271–294. Springer, 1988.
8. R. Gerth, M. Codish, Y. Lichtenstein, and E. Y. Shapiro. Fully abstract denotational semantics for flat concurrent Prolog. In *3rd Symp. on Logic in Computer Science*, pages 320–335. IEEE Computer Society, 1988.

9. R. Gerth and A. Pnueli. Rooting UNITY. *ACM SIGSOFT Software Engineering Notes*, 14(3):11–19, 1989.
10. R. Gerth. Foundations of compositional program refinement - safety properties. In J. W. de Bakker, W. P. de Roever, and G. Rozenberg, editors, *Stepwise Refinement of Distributed Systems: Models, Formalisms, Correctness*, volume 430 of *LNCS*, pages 777–807. Springer, 1989.
11. R. Gerth. Formal verification of self modifying code. In *Int. Conf. for Young Computer Scientists*, pages 305–313. International Acad. Publishers, China, 1991.
12. R. Gerth, R. Kuiper, and J. Segers. Interface refinement in reactive systems. In R. Cleaveland, editor, *3rd Int. Conf.on Concurrency Theory*, volume 630 of *LNCS*, pages 77–93. Springer, 1992.
13. S. Zhou, R. Gerth, and R. Kuiper. Transformations preserving properties and properties preserved by transformations in fair transition systems. In E. Best, editor, *4th Int. Conf. on Concurrency Theory*, volume 715 of *LNCS*, pages 353–367. Springer, 1993.
14. H. Schepers and R. Gerth. A compositional proof theory for fault tolerant real-time distributed systems. In *12th Symp. on Reliable Distributed Systems*, pages 34–43. IEEE Computer Society, 1993.
15. D. Dams, R. Gerth, B. Knaack, and R. Kuiper. Partial-order reduction techniques for real-time model checking. *Formal Aspects of Computing*, 10(5-6):132–152, 1998.
16. R. Gerth, R. Kuiper, D. Peled, and W. Penczek. A partial order approach to branching time logic model checking. *Inf. and Comp.*, 150(2):132–152, 1999.
17. R. Gerth, D. Peled, M. Y. Vardi, and P. Wolper. Simple on-the-fly automatic verification of linear temporal logic. In *15th IFIP WG6.1 Int. Symp. on Protocol Specification, Testing and Verification*, volume 38 of *IFIP Conference Proceedings*, pages 3–18. Chapman & Hall, 1995.
18. D. Dams, O. Grumberg, and R. Gerth. Generation of reduced models for checking fragments of CTL. In *CAV*, LNCS, pages 479–490. Springer, 1993.
19. D. Dams, O. Grumberg, and R. Gerth. Abstract interpretation of reactive systems: Abstractions preserving $\forall$CTL*, $\exists$CTL* and CTL*. In E.-R. Olderog, editor, *IFIP WG2.1/WG2.2/WG2.3 Working Conf. on Programming Concepts, Methods and Calculi*, IFIP Transactions. North-Holland/Elsevier, June 1994.
20. D. Dams, R. Gerth, and O. Grumberg. Abstract interpretation of reactive systems. *ACM Trans. on Programming Languages and Systems*, 19(2):253–291, 1997.
21. D. Dams and R. Gerth. The bounded retransmission protocol revisited. In F. Moller, editor, *2nd Int. Workshop on Verification of Infinite State Systems*, volume 9 of *Electronic Notes in Theoretical Computer Science*. Elsevier, 1997.
22. J. O'Leary, X. Zhao, C.-J. H. Seger, and R. Gerth. Formally verifying IEEE compliance of floating-point hardware. *Intel Technical Journal*, First Quarter 1999.
23. R. Armoni, L. Fix, A. Flaisher, R. Gerth, B. Ginsburg, T. Kanza, A. Landver, S. Mador-Haim, E. Singerman, A. Tiemeyer, M. Y. Vardi, and Y. Zbar. A new temporal property-specification language. In J.-P. Katoen and P. Stevens, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, volume 2280 of *LNCS*, pages 296–311. Springer, 2002.
24. R. Gerth. Sequential consistency and the lazy caching algorithm. *Distributed Computing*, 12(2-3):57–59, 1999.

Rob and Masja, Eindhoven, 1993



Rob Gerth: Personal Recollections

I had the pleasure of knowing Rob for over 10 years. He first appeared in the department of Computer Science at the Technion when I was doing a Phd, for a visit in Israel, while he collaborated with Liuba Shrira.

When I finished the Phd, during my work at Bell Labs, Rob invited me several times to Eindhoven. We worked on several topics, and published together two papers. We worked with Ruurd Kuiper, Wojciech Penczek, Moshe Vardi and Pierre Wolper. It was always a lot of fun and great inspiration.

We all know these people that we try to communicate with about our research, and they are busy and postpone their reply indefinitely, so after some time we give up and never hear from them again. Rob was the striking exception to that: it took me some time to understand that I can always and fully trust that even if we did not achieve a fully published results on a visit, there will be one day, where Rob is going to send some really interesting new and original thread of ideas. I learned that I do not need to remind him. He is just thinking of it, and when he realizes he has something substantial to say, he will. Rob was a perfectionist. We worked on a third subject, and decided the results are not so great. Rob would never send a paper unless he thinks it's an important result and the writing is clear.

I was looking forward to meet Rob again soon. We talked about me visiting Intel in December. I also waited for him to clear his invited talk about the verification of the new Intel processor in the CAV 2004 conference in Boston with his management. I would have reminded anyone else, but with Rob I knew he is taking care of things. Rob has influenced the CAV community with his important contributions. We will remember him.

Doron Peled
University of Warwick

It is with great pain that I struggle with the unacceptably tragic news of Rob's death. For me, Rob always signified the spirit of youth and intellectual rebellion – uncompromising insistence on finding out things for himself and establishing his own truth in the way that would satisfy him personally. This independence of thought and idealistically high standards often led him in paths infrequently traveled achieving many original, ingenious, and fresh insights that were Rob's unique trade mark in science.

It is extremely difficult to reconcile oneself with the thought that this youthful and energetically fresh creative person has been cut in mid stream. The loss to science, industry, and each of us personally, is too great to encompass in few words.

Amir Pnueli
New York University & Weizmann Institute

I have known and shared many happy times with Rob. He and I shared a grant from the EU for many years, which let us eat, drink, and (sometimes) work around Europe. Most of all I remember fierce arguments, highlighted by the intensity of Rob's in-your-face burning eyes. These were all totally enjoyable, and happily resolved over a more-than-few drinks.

Mike Reed
Oxford University

Fig. 1. Rob, Amir Pnueli and Mike Reed in Loutro, Crete, 1995.

Remembering Rob Gerth, I am deeply saddened by the loss of a real friend. He was my first Ph.D. student. The difference between him as supervisee and me, his supervisor, quickly evaporated because of his rare, incisive intellect, his passionate devotion to our field – program verification – his integrity, his admirable courage, and his honesty. He was a man to steal horses with.

He surpassed me scientifically shortly after finishing his thesis. Both of us knew that, and for me this was a reason for pride and enjoyment – The Torch of Scientific Research had been passed on to a Worthy Successor.

Rob, Marly Roncken, Niek van Diepen, and I met in the beginning of the 1980's as part of an informal research group in program verification at the University of Utrecht. We had a proof theory for Brinch Hansen's monitor-based language Distributed Processes as goal in mind. Marly contributed our first representative example, her Monkey-Bananas puzzle, and Rob observed that its correctness proof involved Owicki's interference-freedom test. These efforts led eventually to our first joint publication in 1982.

This goal of a proof theory for monitors subsequently evolved into the subject of Rob's thesis, finished after my group had already moved to Eindhoven. But Rob wanted to obtain his Ph.D. degree at "his own" University, that of Utrecht. Jan van Leeuwen made this possible. I still remember sitting along one of Utrecht's beautiful medieval canals after the ceremony, with my wife Corinne and our daughter Jojanneke, trying to convince Rob that a meal for his doctoral committee and his guests would be the appropriate end to a wonderful day. Marly convinced him in the end, and I finally noticed they had become a real couple.

Fig. 2. Jan van Leeuwen handing over Rob's Ph.D. certificate, 8 May 1989 (left). After the ceremony (right), Rob exchanged his tails for a T-shirt version – a present from WP in response to Rob's (reluctant) yielding to the University's ceremonial dress code.

In the meantime we had started participating in the ESPRIT research programme of the European Commission. This was made possible by an eminent team of research students, consisting of Rob, Ruurd Kuiper, Kees Huizing, Ron Koymans, and Jozef Hooman at Eindhoven, and Frank Stomp and Job Zwiers at Nijmegen University. Amir Pnueli commented later that they constituted the most gifted single group of Ph.D. students he had ever collaborated with.

In my second EU research project Rob became Eindhoven's team leader. Those projects led, among others, to the birth of the now dominant CAV series of conferences on Computer-Aided Verification. Frequent guests of the chair were Allen Emerson, Amir Pnueli, Joseph Sifakis, and many others who were to lift program verification later to its current level. Within a couple of years Rob Gerth had evolved from a proud citizen of Utrecht to a researcher having a world-wide view of program verification with an established scientific reputation, and had changed his main line of research to computer-aided verification.

His parting gift to Eindhoven University was the ESPRIT project VIRES, the proposal of which he wrote single-handedly, on the verification of a mobile phone protocol. That project was subsequently led at Eindhoven University by his first Ph.D. student, Dennis Dams. It constitutes still a topic for invited lectures!

I had moved in the meantime to the University of Kiel, Germany, Rob moved later to Intel, and Dennis Dams to Bell-Labs. An era of scientific excellence and vision in program verification in the Netherlands had come to an end.

Willem-Paul de Roever
University of Kiel

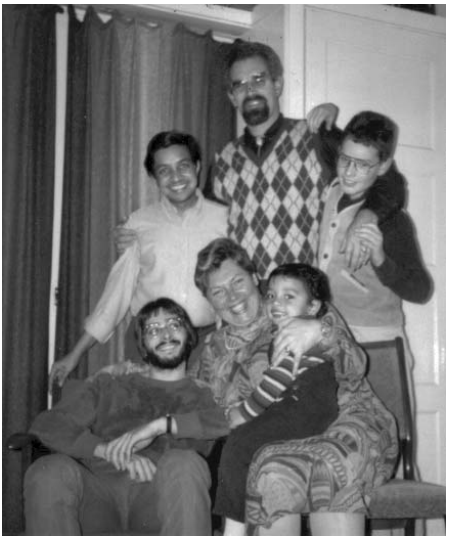


Fig. 3. Two Ph.D. generations: Dennis Dams with Rob at the PROCOMET workshop in Italy, 1994 (left), and Rob with Willem-Paul, Corinne and Sanne de Roever and the Shyamasundar family in Utrecht, 1983 (right).

I had not seen Rob for two years, but when I got the message of his death he was suddenly there again, in a long sequence of memories. The many years in Eindhoven, first as M.Sc. advisor, then co-promotor of my PhD thesis. In that sense he was my "scientific father", a role that he fulfilled with dedication.

Thinking back, what I found most characteristic about Rob was his calmness. Always was he willing, and took ample time, to listen patiently to what I had to tell him. We discussed long, and often late hours about work and many other things. He could continue working on a joint paper, unstirred, until just before, or after the submission deadline. And I remember how relieved I was when he finally showed up on the platform escalator of Eindhoven station, half a minute before the departure of our international train. His face showing only a slight surprise about my state of panic.

With that self-evident calmness he allowed me to continue working on my PhD thesis, even well beyond the regulatory four years. I do not actually remember whether I ever told him how grateful I am for that.

Between 1989 and 1997 we have done many things together: working, eating, running, traveling. Memories that we talked about together, not so long ago. Memories that now suddenly take a different role. The PROCOMET workshop in San Miniato. That restaurant in Haifa where we had dinner each night during our working visit to the Technion. Those memories now turn into memories of Rob himself.

Dennis Dams
Bell Labs

I am deeply grateful for having had the privilege to work with Rob for the past five years. Rob was always one of the first people I consulted whenever I had questions on a scientific topic. He was one of the very first people who recognized the importance and significance of the GSTE work, and even encouraged me on many occasions to write a book on the subject. I remember that he carefully read the first draft of the theory paper, and walked into my cubicle one day with the copy full of his valuable comments ... and a coffee stain. He apologized for the mess, but I thanked him and told him jokingly that I would like to keep it as it would become a great souvenir when he'd be a very famous computer scientist some day. I have kept the copy and I will keep it forever as a memory of Rob.

Recently, Rob and I worked closely on mentoring the protocol verification project by Professor Lin in China. He was very enthusiastic and upbeat about it. I still remember the words he said when we discussed how to steer this project. With his unforgettable smile and an emphatic "No", he said "Jin, we should define the problem very clearly, but we cannot constrain how he is going to solve it". How could one argue with that?

Jin Yang
Intel

Rob, and Marly almost equally long, I have known since their post-university days at the University of Utrecht. We were coached, quite independently and at different places, by Willem-Paul de Roever. Initially with him, Rob and I at a later stage spent a period of about 10 years at Eindhoven Technical University, Rob as leader and later initiator of projects with European Community funding.

Workwise, that meant interaction on a day-to-day basis, where Rob's scientific perceptiveness, and his perseverance and integrity were a constant pleasure and inspiration. Spotting a new idea like a vague shadow of a fish, but then also catching it - making it precise and finding the appropriate formalism to describe it; accomplished artist and craftsman both. To honour Rob's love for books: Il Miglior Fabbro - certainly. Socially it meant isolated, few, pinpointed, but intense and happy interactions. Like triggered when our children were born (there still lives a toy-frog in our bathtub donated by Rob and Marly, though politely named after Rob), or when the mood was right, tired-out after work on a project assignment at the Weizmann Institute of Science in Israel.

This mix felt absolutely right. To me Rob was, apart from the driven, talented scientist, an unusually complete human being, in his tastes as well as in his priorities. Leaving a cinema late at night I once saw Rob and Marly cycling around rather hollow-eyed: having worked together on some paper, they had forgotten about time and food. But at that moment work was far away; their being together radiated happiness and was a joy to look at.

A furtive try to phrase what Rob means to me:

A good thing about a friend is that you understand and value him; the better thing is that he understands and values you - Rob did. The book "Pallieter" of which he once gave me a fine old copy (hunting down one, after noticing that I admired his) will always be on my shelf, as Rob will remain in my life.

Ruurd Kuiper
Technical University Eindhoven

Rob and Marly, Eindhoven, 1990

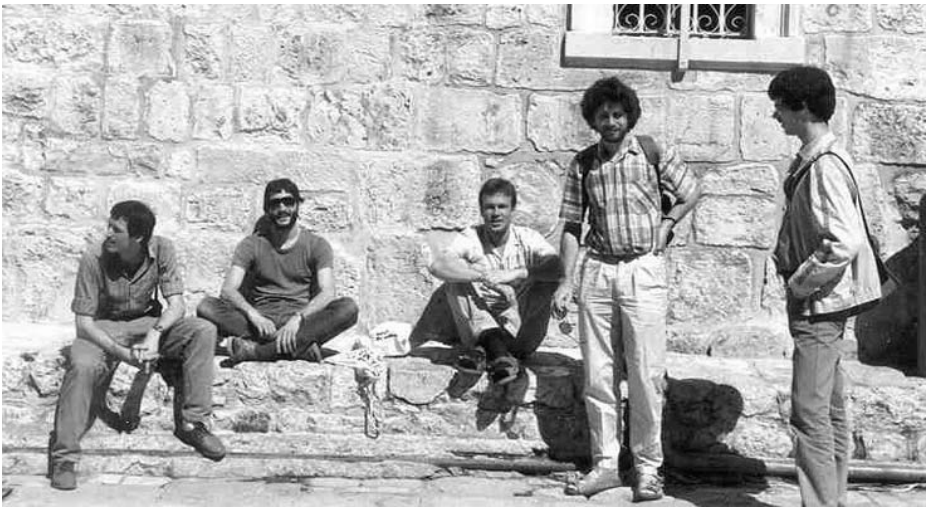


Fig. 4. Jozef Hooman, Rob, Ruurd Kuiper, Kees Huizing, and Ron Koymans, Israel.

Rob was my co-promotor and we worked together at the Eindhoven University. He was also my daily supervisor during my Ph.D. work and I learned so many things from him. About English writing, for instance. Maybe also a few bad habits, but what I remember most is his attitude to science. He combined strong opinions with a very open mind, which is exceptional, I think. Sometimes, when I had made a nice exposition of some point of view, he could ask with his eyes wide open in what seemed true surprise: "Why?". This single word made me rethink my whole position and suddenly every assumption I made seemed questionable. Although he had a very broad knowledge of computer science, he managed to approach the research field every time in a fresh and original way.

His original and sharp thinking also lightened up our coffee break discussions and now that he died I realise that I still miss him, even after these years. And not only for his conversation, but also for his independent mind, whether it be science, politics, or conventions and manners.

Kees Huizing
Technical University Eindhoven

I have written two papers with Rob. He was a challenging collaborator. He took nothing for granted and challenged everything. Working with him was like a furious tennis match. At times, you could not tell whether it was collaborative or competitive, it was so intense. When the end result finally emerged, it was always of superb quality, having been polished by clashing intellects. Rob's uncompromising quest for scientific truth forced his collaborators to bring forth their best ideas. His untimely departure leaves us with a gaping hole.

Moshe Vardi
Rice University

When I first met Rob in 1987 he was a student at TU-Eindhoven. All of Willem-Paul's group were impressive, but Rob stood out somehow. His brilliance and intensity were so plainly visible. He wasn't like a student. Instead, he already had the maturity and judgement of an excellent junior faculty member.

I followed Rob's career with interest. He brought much needed clarity and illumination to everything he worked on. I myself was doubly impressed with Intel's seriousness regarding formal verification when I learned they had hired Rob.

Rob was both a gentlemen and a scientist. I myself, his friends and colleagues, and the community will miss him enormously.

Allen Emerson
University of Texas

I was shocked to hear the sad news about Rob. Rob was a well respected member of our community – an international community of researchers working on specification and verification. I met him frequently at workshops and conferences.

Rob's intelligence was obvious, and he contributed a great deal to the progress of our field. But what impressed me most was his good sense and openness to new ideas. He was free of preconceived notions of how things should be done, and he was always willing to listen to what others had to say.

Leslie Lamport
Microsoft

Rob was very kind and welcoming to me whenever I visited Intel. Although we never worked on a project together, I had a tremendous respect for his scientific knowledge and often popped by his desk to consult him on tricky technical points. Rob was a generous man and always willing to share his time.

Rob also had a wonderful sense of humor and was a simulating and enjoyable lunch-time companion. I remember well how his face would light up with a smile, as he would agree with a gently ironical "of course" to some absurdity or other we were discussing.

In his work, I think what characterised Rob was his tremendous dedication to really getting to the bottom of difficult scientific questions. He wasn't content just to have an approximate understanding of a problem and its solution. He wanted to really *know* – and as a first-class computer scientist he had the ability to get there too.

Tom Melham
University of Oxford

Rob, Utrecht, 17 October, 2002

Rob had a remarkable mind. He had a deep understanding of computer science theory, particularly automata theory. He explained many concepts to me and to others on my team. Yet, he was never condescending. He enjoyed sharing his knowledge, and he didn't give up on an explanation until it was clear.

I knew that Rob had many interests outside of work, but sadly I didn't learn the full scope of those interests until after his death. I have come to understand that Rob was a true "Renaissance Man". He had a library with thousands of books and CDs, and the home he shared with Marly is filled with art and sculpture from around the world. I'm amazed at the breadth and depth of his interests and knowledge.

Rob had a subtle, wry sense of humor. He was often amused by my (usually futile) attempts to effect change despite the bureaucracy inherent in a large company like Intel. After one of my skirmishes with the powers-that-be, Rob presented me with a special award–something that he had invented just for the occasion. It was the inaugural Strategic CAD Labs "Don Quixote" award: a small porcelain Dutch windmill. He told me that tilting at that windmill would often be more productive than "tilting" at bureaucracy.

Robert Jones
Intel

Long before I met him, I actually tried to ignore Rob. Everywhere I went, people were raving about how smart he was, what a great guy he was, etc. At some level, I guess I was jealous, and I arrogantly said to myself, "If this Rob Gerth guy is such a hot shot in verification, why haven't I met him yet?" I should have heeded my own advice: you can't tell the difference between someone who is two steps ahead of you and someone who is two steps behind you. When someone is a bit ahead of you, you can understand what they're doing and why, and you acknowledge that they're ahead; similarly, when someone is a bit behind you, you understand exactly what they're missing and how they are behind you. But when someone is too far ahead, what they say and do is incomprehensible, and thereby becomes indistinguishable from gibberish. For me and Rob, it took me a lot of hard work and tough thinking before I reached the point that I could start to appreciate his comments and insights, and work he had done long before. Let me give a few concrete examples:

- I've long considered the verification of multiprocessor memory systems to be one of my specialties. This was how I got started in verification, and I've done quite a bit of work in this area. Only in the past few years, though, have I started looking at verifying the conformance of the system to a formal memory model, rather than verifying simple ad hoc assertions. As I moved into this "next step" in verification, I found that Rob had been there already. In fact, he had already edited an entire journal special issue on this topic!
- I have always ignored work on strange automata variations as interesting, but irrelevant theory. When I visited Intel SCL to work on the eminently practical GSTE, Rob remarked, "Oh, it's basically forall-automata. I'm more interested in alternating automata." After weeks of concentrating on GSTE, I eventually came to realize, "Oh yeah, I guess it *is* a forall-automaton." And I've started seeing alternating automata (the generalization of forall) popping up everywhere in my own research and that of others.
- The other comment Rob made to me about GSTE was that it was "using a graph to structure fixed-point computations". At the time, I understood his comment at a syntactic level: GSTE does indeed use a graph, and various fixed-point computations are done in accordance to the graph. Only after a year and a half am I starting to see that there was more to his comment, that there is a general principle, that there might be promising ways to generalize and exploit that general principle. I still don't think I can see what he saw, but I'm starting to catch a glimpse of his ideas.

Given how far ahead Rob's thinking and research has been, I take solace in knowing that he will continue to teach me things for years, perhaps decades, to come. I expect that, with enough hard work and hard thinking, I'll yet again rediscover something of Rob's, and I'll be able to see him, with that it's-really-very-obvious-if-you-think-about-it look on his face, showing me the way forward when I'm finally ready to follow him.

Alan Hu
University of British Columbia