The Humble Humorous Researcher

A Tribute to Michel Sintzoff

Axel van Lamsweerde[1]

Many of us lost a close colleague on November 28, 2010. More than that: we lost a friend. One that we used to meet regularly, all over the world, always listening to us, making interesting comments and suggestions on our work, joking on every occasion and making us discover how much fun our business is.

Rather than reviewing his work in detail, this note puts more focus on Michel's rich personality, with the hope that it will bring back fond memories among those who were lucky enough to share some good times with him. The style is intended to be personal and informal. Michel would have hated anything different, to be sure.

Born in 1938 in Brussels, Michel completed his master's degree in Mathematics at the Université catholique de Louvain (UCL) in 1962. After two years of civil service as a maths teacher in Katanga (Congo), he entered the MBLE Research Laboratory in Brussels in 1964 (MBLE stands for "**M**anufacture **B**elge de **L**ampes et matériel **E**lectrique"). This was a branch of Philips Research Labs specifically dedicated to background research in applied mathematics and computing science. After 18 years of research in programming languages, formal semantics, program analysis and concurrency at MBLE, he joined the newly founded department of Computing Science at UCL in 1982 with new interests in diverse areas such as proof systems, control theory and dynamical systems. Michel contributed to the PhD work of dozens of people in Belgium and France, as supervisor or contributor, without ever having been interested in getting a PhD himself. He received a Doctorate Honoris Causa from the Université Joseph Fourier in Grenoble (France) and was a member of the Informatics section of the Academy of Europe. Michel was an Emeritus Professor at UCL since 2003. The two of us were face to face in the same office at MBLE Research for 10 years (1970-1980), and had adjacent rooms at UCL for another decade (1993-2003).

Many consider Michel's paper *Calculating Properties of Programs by Valuations on Specific Models* (Proc. ACM Conference on Proving Assertions about Programs, 1972) as the precursor paper on abstract interpretation**.** This paper and others he wrote around that time on program verification and type discovery are heavily cited in the paper generally considered to have

[1] Warmest thanks to Gérard Huet, Jeremy Gibbons, Cliff Jones, Yves Ledru, Krzysztof Apt and Jeanne, André, Marie, Ivan, Catherine and Paul Sintzoff for their support and input material to this note. Thanks also to Joanne Allison, Dominique Gallez, Carlo Ghezzi, Lambert Meertens, Bernhard Mōller, Pierre-Claude Scholl and Bas van Vlijmen for their comments.

To appear in *Science of Computer Programming* (Elsevier) and *Formal Aspects of Computing* (Springer-Verlag).

opened the field (Cousot and Cousot, 1977). The technique described in his 1972 paper was simultaneously implemented in Paul Branquart's optimizing compiler for Algol68, confirming how powerful it was.

At that time Michel had a strong interest in the precise definition of language semantics; he was co-editor of the Revised Report on Algol68. To distract him from this Herculean task, I used to challenge him with my own research problems. This resulted in seven joyful years of joint work on invariant generation and formal derivation of concurrent programs. I noticed that Michel kept citing this work up to his very last papers (e.g., *Mathematics of Program Construction 2008*). Recent interactions suggested that he was still interested in some joint work around this, and I regret now not having picked up the gauntlet.

With a rich harvest of European grants and PhD students, Michel expanded his research horizons as he moved to the academic world. While staying fairly active in type theory and program verification, he developed techniques for formalizing and composing design decisions in a modular way; for specifying, refining and composing proof structures; and for synthesizing control information for multiple types of systems, notably relational programs, reactive systems and dynamical systems. His earlier 20 years outside the academic world inoculated him against the *publish-or-perish* syndrome. Michel published relatively few papers, but these were consistently innovative and thought-provoking. Quoting Cliff Jones, *I personally rate him one of the most imaginative researchers I've known. He had exotic ideas and worked them out carefully, a rare combination.*

Michel was also instrumental in collective projects such as *Raisonner pour programmer* (Dunod, 1986), a book on program design strategies and tactics which at that time had no counterpart of such quality, in my opinion. The book's author, Anna Gram, was a joke name for a largely French working group whose other members were J.C. Boussard, J.P. Finance, C. Gresse, P. Jacquet, G. Perrin, A. Quéré, P.C. Scholl, L. Trilling and J. Voiron.

From the very beginning of his research career Michel had two obsessions: *technical precision* (so much needed in software engineering research) and *formal construction of correct artefacts* (as opposed to verification of possibly poorly designed ones). Fortunately enough, quite a few of us were infected by these.

Michel's main research tools were paper and pencil. He was often joking about our business by quoting Edison's phrase *1% inspiration, 99% perspiration.* This was measurable by the number of paper sheets full of formal developments that was accumulating in his trash weekly. His family found hundreds of handwritten pages meticulously numbered YYYY-MM-nn (year-month-pageNumber). The last page is numbered 2010-11-1a, thus written the day before he went to the hospital.

Michel also used to spend a significant portion of his time reading papers and books in many different areas, including areas not connected to computing science. He kept saying: *it's good to learn from others to refrain from reinventing the wheel*. His scholarship proved invaluable to those working with him. How many times did we not get relevant references from him that we would never have read otherwise? How often did we not get directions and technical suggestions based on his erudition?

Michel was deeply involved in multiple services to the community. He launched the *Science of Computer Programming* journal in 1981 and was its editor-in-chief until 1999. I remember lively discussions about the scope of the journal, the initial editorial board and his questioning of the journal's name set up by North Holland; he thought that *Science* was a somewhat arrogant claim and *Computer* was inadequate (as convincingly argued by Edsger Dijkstra, who wrote a word of welcome in the first issue). Krzysztof Apt reported that Michel sent occasionally insightful notes in which he was giving the editors useful suggestions about the desired directions *science of programming* as a field should aim at. I'd like to cite Michel's own words in his editorial for that first issue, as I believe them still to be relevant 30 years later:

> SCP is intended to further the publication of systematic and formulated knowledge on program design. This allows for specific case studies as well as more general techniques on program development, and implies the use of the classical scientific method, to foster economy in brainwork and repeatability of the effects of proposed laws. This intended role should be clearly useful, being at the heart of what can be called the first intellectual industry in our history: it should assist far-reaching disciplines like programming methodology, software engineering, and design technology, which themselves underlie notational systems, support systems, and production systems. […] A forum based on the written word could facilitate communication within and between generations, and across unkind boundaries, as exemplified by the library of Alexandria.

Michel put a lot of energy in the SCP project. The impact of his dedication over 18 years is now measurable in terms of the impressive number of influential papers that appeared in the journal during his editorship.

Michel quite often served in review panels for European, national, and university research programmes – not to speak of countless thesis committees. As many of us experienced, he cultivated the art of kindly but firmly raising issues that were right to the point. There is a French saying *act as a donkey in order to get the bran* – something like *play the fool to catch the wise*. Michel was exactly like this. He used to naively ask the right questions and could quickly put his finger on shaky aspects or limitations without ever being destructive or unfair. When you told him enthusiastically what you thought to be a great idea, he invariably replied: *on se calme, on se calme* (*calm down, calm down*), giving you good reasons why your idea had to be further worked out. Gérard Huet reported a typical anecdote about this. He was presenting a demo of his Coq proof assistant during a review of a European project for which Michel was a reviewer. While Gérard was showing complex, elegant proofs Michel asked: *can you please prove* $(a + b)^2 = a^2 + 2ab + b^2$ *for me?*. He could find exactly the right counterexample. This problem was not at the time solvable in Coq other than giving by hand all the equational steps involved in proving this identity. It is only a few years later that this sort of algebraic identity could be solved trivially by the reflection tactic applied to the theory of rings – as a result of Michel's constructive criticism.

Michel has been a pillar of diverse working groups over many years, including IFIP Working Group 2.1 (Algorithmic Languages and Calculi) since 1968 and Working Group 2.3 (Programming Methodology) since 1976. With typical hospitality he hosted quite a few meetings in delightful places in Southern Belgium. He reluctantly agreed to be chairman of WG2.3 from 2003-2006. His characteristic interest in everything made him perfect in the role. As Jeremy Gibbons said on behalf of WG2.1 members:

> Michel was the longest-serving and most faithful active member of WG2.1, and we have all felt his influence over the last 42 years. He had a rare combination of excellent insight in technical matters, a determination always to understand, and a knack for asking penetrating questions. Even when pointing out flaws or omissions in our work, he always did so in a very constructive way. It is thanks to Michel that the ALGOL 68 Revised Report was so much more readable than the original Report: the structured presentation of the Semantics, replacing the goto style of the original, is due to him, as is the use of predicates in the syntax, and their use to bring the semantic context conditions into the syntax.
>
> We all have fond memories of Michel from the many WG2.1 meetings we have shared. He was unfailingly kind and welcoming, especially to younger academics, and was without arrogance or pomposity. We always knew that we had arrived at the right place when we heard his joyous laughter emanating from the general vicinity of the bar – often in response to his own jokes, but they were worth it. Dave Wile recalls the meeting in Ameland, Holland, the week of 9/11, when he was giving the banquet speech: *He always loved to tease and instigate trouble. We sat together at the banquet and as I began my speech, he saw that my notes were on my Palm Pilot. During the speech, he quietly changed the page so I had no idea where I was or what I was talking about. I certainly hope he retained that impishness right to the end.* Michel was loved and respected by us all, and he will be sorely missed.

The impressive number of similar reactions within a few hours after circulation of the sad news revealed how many personal relationships Michel succeeded in creating within the community. Remarkably, the most common thought emerging from many of these was about how Michel influenced careers in some way or another. In my case, for example, he decisively stepped in twice: at the very beginning, to offer me a desk in his MBLE office so that we had a better chance to work together, and 20 years later, to urge me to apply for a UCL position close to him. He had a systematic policy of stimulating young researchers. I remember a flight from Pittsburgh where he proudly admitted this to be his main professional achievement.

This brings us to Michel's great humility – explaining the title of this note, with a wink to Edsger Dijkstra's Turing Award paper *The Humble Programmer* that Michel loved. I never heard him speak highly of himself. His last wish I am aware of was to avoid any speech at his funeral, fearing that *people would say all sorts of things that are not true*. In fact he told me exactly the same when he retired, discouraging us from organizing a Festschrift on that occasion. We never heard of Michel being a member of the Academy of Europe, as another example. After he received his Doctorate Honoris Causa from Grenoble, he kept joking that this was not a Doctorate Laboris Causa.

Humility came with an instinctive aversion against any authority argument. I remember when I had to present our first joint paper at a conference attended by all the big names (the ACM-IEEE International Conference on Reliable Software, 1975). I was terrified by this. To make me feel confident, he explained how he managed this problem at an earlier conference when he was in that same situation. As he noticed *the* big name surrounded by an admiring court, he jumped into the circle and told the big name: *I know you are the Tsar of computing science but you don't scare me* (and he started laughing). This together with some technical verse he had prepared worked remarkably well. That story helped me a lot on multiple occasions.

Michel's distinctive laughter became a legend. It has never been that clear to me what part of it was controlled and what part was not. Most often it referred to hilarious situations he had caused and in which he was deeply involved. Typically, after creating the initial state, he iterated on expiration-inspiration cycles under uncertain termination conditions; every inspiration triggered a siren, increasingly resonant from one iteration to the other. After a few iterations, the noisy conversations among hundreds of persons at a conference banquet stopped and everyone was laughing by contagion. Many of us have plenty of anecdotes to report. My first exposure to this laughter was on my very first day of work. I was introduced to everyone at MBLE, room after room. When I entered Michel's office, he asked me what I planned to work on. I tried to be as smart as possible and declaimed my lesson. After having listened to me very seriously, he said: *very interesting, but are you sure that Philips will pay you for that?* – then that laughter when he saw my face. A much bigger, longer one occurred a few years later as Michel's wife and children came to the office on a Friday night to catch him. The trigger was his young boy saying: *Oh Dad, that's not like at my school, it's too easy for you to cheat and copy each other's ideas!* Friday afternoons were particularly enjoyable, from stories to stories we used to tell each other, intertwined with laughter, to the point that often half the lab ended up in our room to share our hilarity. Over the years I have interpreted Michel's laughter as a recommendation for never taking ourselves or our business too seriously.

Michel had other extra-professional skills. I remember him once showing me how he could concurrently write English with his right hand and Russian, his parents' language, with his left hand. Speaking of his hands, many will remember the admirable dexterity with which he was able to serve French fries with two spoons in a single hand – a skill he acquired during student jobs in Belgian restaurants. His family reported that he practiced regularly this art on Sunday lunches, the process being sometimes abruptly suspended by the need to write down a fresh idea he just got.

I used to meet Michel regularly since he retired. He kept working on things he was interested in; his last paper, co-authored with Roland Glück and Bernhard Möller, appeared in *AMAST2010*. He kept saying: *it's so great to be emeritus, you feel like being on perpetual*

sabbatical. As Krzysztof Apt wrote to him: *I understand that you are doing fine, even though you retired,* he replied: *replace "even though" by "because"*. When he left to the hospital, Michel was still working on a new paper (with many references to work from the seventies); the printed draft, full of red markings, is dated October 30.

Michel used to say: *we only have one single life* to deter people from considering too many research directions and encourage them to choose the right ones. On a recent discussion about the need, in my area, to consider bounded *Achieve* properties rather than liveness ones, Michel reminded me that he was for his part only interested in so-called *S-properties*, that is, properties that are achievable within Sintzoff's lifetime. We would never have expected S-properties to have such a short time frame.

As Brian Randell said when he was told of Michel's death, ... *but he was always so full of life*. Beyond Michel's contributions to our field, I'm sure that his kindness, subtlety and deep sense of humour will remain in the heart of many of us for a long, long time.

Axel van Lamsweerde

ICTEAM Institute, Dept. Computing Science
Université catholique de Louvain
B-1348 Louvain-la-Neuve (Belgium)

avl@info.ucl.ac.be