Greetings to Bertrand on the Occasion of his Sixtieth Birthday

Tony Hoare

Microsoft Research, Cambridge

I first met Bertrand in July 1978, at a lovely French Château in le Breau sans Nappe. It provided most luxurious accommodation in the lovely French countryside for two weeks for my family with three children. The food was also superb.

The Château was run by Électricité de France as an educational institution for residential courses. Bertrand was at that time employed by EDF, and organised a Summer School on Computing. It was superbly organised. I learnt there that it is always rewarding to accept an invitation to an event organised by Bertrand; and my latest experience was in Elba as a lecturer in a LASER Summer School a few years ago. Thank you again Bertrand.

At le Breau, Bertrand invited me to lecture on concurrency (Communicating Sequential Processes). Barry Boehm lectured on Software Engineering Economics, and Jean-Raymond Abrial (jointly with Bertrand) on specification in an early version of Z. Of course, since then they have all become famous; and like me, they have all remained active to this day. I have met them again many times in the last thirty years, and they have remained good friends to me and my family.

In 1983 Bertrand migrated to an academic position in California, and in 1985 he founded a Company in California, to implement and market his successful programming language Eiffel. I became editor of the 'Red and White' Prentice Hall International Series in Computer Science. I wanted to persuade Bertrand to write a book for the series. I had no difficulty in doing that. In fact in his reply he said he was already engaged in writing six books, and which one did I want? So my main task was to help him to select a topic on the theory of programming languages, and then to persuade him to stop writing the other five books.

I did not succeed. The agreed and scheduled date in the publisher's contract receded into the past, and Bertrand became ever more deeply apologetic. Finally, he offered a consolation for the delay, in the form of a manuscript for a completely different book, entitled Object-Oriented Software Construction. Would we like to publish that first? Certainly we would. I could see it was an excellent book, and Prentice Hall was surely consoled by the fact that it turned out to be the best-seller of the whole series. It was published in 1988, and was translated to seven languages (many of which, incidentally, Bertrand speaks well). It is still selling very well in its second edition (1997). The original theoretical book on programming (also very good) appeared two years later.

In 1999, I retired from my academic career and moved to Microsoft Research at Cambridge. In January 2001, I visited the Microsoft Headquarters in Redmond, to conduct a survey of the ways in which Microsoft program developers

S. Nanz (ed.), *The Future of Software Engineering*,
DOI 10.1007/978-3-642-15187-3,

were using assertion macros in their code. It was mostly as test oracles, to help discover errors. Bertrand had been invited to Redmond at the same time to talk about Eiffel, the first language which supported the use of assertions as contracts between the writers of separate modules of a large system. In the evening of Saturday 20 January we met at my hotel to reminisce about the past and speculate for the future.

My hope was that Microsoft developers would extend their use of assertions to help discover errors in testing, and would discover later that the assertions would help them to avoid the errors altogether. Bertrand's hope was that Microsoft would adopt the Eiffel philosophy of use of assertions as contracts. I am glad to say that Bertrand's hopes have already been realised. Contracts are available in the languages supported by Microsoft's Visual Studio. My hopes are still hopes: but they are being actively pursued inside Microsoft. There are over a hundred researchers in Microsoft Research world-wide, developing theories, constructing tools and conducting experiments in program verification. There are even more in the development division, actively transferring the technology and improving and adapting the tools for use by different teams.

Bertrand shares with me the long-term ideal that programs should be specified and designed in such a way that the correctness of the delivered code can be reliably checked by computer, using mathematical proof. We felt that this was a challenge in which academic research was needed to make significant progress. I and Jay Misra therefore suggested an IFIP Working Conference on Verified Software, Theories, Tools and Experiments, which Bertrand volunteered to organise – after which its success was assured! This was the start of a series of workshops and special interest groups convening at conferences held throughout the world. Two further full international Conferences have been dedicated to this topic, in Toronto in 2008 and in Edinburgh this year.

Bertrand's enormous contribution to this international endeavour has been derived from his unique experience of the Eiffel project. In pursuit of the long-term scientific ideal of correctness, and exploiting his superb engineering judgment, he has been the chief architect of the EiffelStudio program development environment. This has recruited a community of enthusiastic users, including large institutions and companies. The Eiffel language has continued to develop towards its ideals, and its community has been willing to act as early adopters in experimental use of the new features. This provides a body of experimental evidence which shows the advantage of a principled approach to software engineering. I have reason to be personally grateful for the many discussions that we have held on theory and language design. And we all have good reason to wish him a happy birthday, as well as a long continuation of his extraordinary range of interests and activities.

Yours,

Tony.

Author Index