Foreword

It is a distinct pleasure to have been asked to write a forward to Brian Cantwell Smith's remarkable book, *Computational Reflections*. I count myself one of Brian's oldest friends and colleagues, having known him since the late 1970s when we were both in our twenties, computer zealots working at Xerox Palo Alto Research Center (PARC).

This volume is ambitious but Brian's lead up to the work laid out in these pages supports his ambitions. He started undergraduate study at Oberlin College in Ohio in 1967, where his interests included both physics and religion. Leaving after only two years, he became employed as a programmer at the Division of Physics Laboratory of the National Research Council of Canada (Ottawa); his project involved both the Fermilab (Chicago) and the Lawrence Research Laboratory (Berkeley). At all three sites he "programmed like crazy" in machine language on PDP 9 and PDP 15 microcomputers, building systems for experimental control and data gathering.

After a few years, Brian moved to Cambridge, MA to take classes at the Massachusetts Institute of Technology (MIT), studying what was then known as Social Inquiry; in particular the politics of high technology. But Brian soon realized that the understanding of computing the social scientists were critiquing was not the understanding of programmers, the latter of which he described as, as "computation in the wild".

"What drove me out of Social Inquiry and back to [Computer Science] was needing to be back in the practice. That skill was not something that people on the outside understood. I had to go back into the heart of the beast, as it were."

To return to his "computation in the wild", he began taking classes geared toward applying to the Ph.D. program in Electrical Engineering and Computer Science.

There was one small hitch, however, to his continuing education; when the MIT administration discovered Brian had not completed an undergraduate degree, they determined he could not be registered for graduate study. However, he so impressed the newly-appointed head of the Artificial Intelligence Laboratory, Patrick Winston, that Winston gave Brian an informal oral exam in topics from the MIT undergraduate computer science curriculum and based on Brian's performance, awarded him the credits necessary for an MIT undergraduate degree. This cleared the way for Brian's admission to the MIT Computer Science graduate program. Brian's unusual depth in both the technical and synthetic were unsurprisingly recognized early; this remarkable intellectual combination has continued to this day.

At MIT, Brian met Terry Winograd, who in 1976 left MIT for Stanford. Terry invited Brian to spend the summer in the Understander Group at PARC, where Brian joined in the development of a Knowledge Representation Language (KRL), which came to embody some of the ideas that Brian later developed in his Masters and Ph.D. dissertations [refs].

These biographical details bring us to the time and place where he and I first met: PARC. It was really, from this moment that Brian began to build the expertise necessary for the profound but radical thesis of this book:

- Member of the Scientific Staff, Xerox PARC
- Director, Xerox PARC System Sciences Lab
- Adjunct Professor of Philosophy, Stanford University
- Founding member of Stanford University's Center for the Study of Language and Information
- Founding member and first president, Computer Professionals for Social Responsibility
- President of the Society for Philosophy and Psychology
- Professor of Cognitive Science, Computer Science, and Philosophy, Indiana University
- Kimberly J. Jenkins University Distinguished

Professor of Philosophy and New Technologies, Duke University
- Dean of the Faculty of Information, University of Toronto
- Professor of Information, Philosophy, Cognitive Science, and the History and Philosophy of Science and Technology, University of Toronto
- Senior Fellow, Massey College, University of Toronto
- Reid Hoffman Professor of Artificial Intelligence and the Human, University of Toronto

Working at PARC, "As an exercise in using KRL representational structures, Brian Smith tried to describe the KRL data structures themselves in KRL-0. A brief sketch was completed, and in doing it *we were made much more aware of the ways in which the language was inconsistent and irregular*. This initial sketch was the basis for much of the development in KRL-1."[1] (emphasis added)

In addition, Brian's input into the (never completed) KRL-1 meant that not only could some parts of a system's data be *about* other parts, but that these data would be more than just commentary. They would actually play a role in the system's operation. For Brian, working on KRL-1, this use of data was motivated by a desire to formulate aspects of knowledge representation – negation and disjunction – intensionally as knowledge about knowledge, rather than extensionally as primitives built into the vocabulary of the representation language itself.

Brian called this set of ideas idea *reflection*

There was an assumption at the heart of Brian's reflective architectures started in KRL-1 but which occupied just one section of one chapter in his proposed Ph.D.; its resolution has

[1]Bobrow and Winograd, 1978, "Experience with KRL-O: One Cycle of a Knowledge Representation Language", in *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, Morgan Kaufmann Publishers, Burlington, MA. Available online at https://www.ijcai.org/Proceedings/77-1/Papers/032.pdf.

been a lifetime of intellectual thought and work for Brian.

Looking back, it seems that any difficulties in the resolution acted rather like the grit in the oyster, stimulating Brian's wholesale reconsideration of the nature of computation, and Computer Science as currently practiced, which is, at its heart, what this book is about.

~~I daresay that from my vantage point as a career computer scientist, Brian,~~ has accurately diagnosed a problem that the whole field of computing has missed. Brian convincingly argues that this has led the field to be altogether stuck, ~~but unable to see that.~~ He believes that seeing the problem will unstick the field.

As Brian himself revealed to me in a recent conversation, "That this is important needs to be said." As computer scientists as well as philosophers of computing who care about the field, this claim deserves our attention.

A word to the wise: because the discussion is not only interdisciplinary but also relentlessly foundational, it cannot assume a particular starting place. So it is not generally an option to say that this or that term will be used *as in* one or another field. And this will especially affect the use of standard terms in computer science. The best ~~interpretive strategy~~ is probably to hold definitions in abeyance, and let the ensuing discussion and argument do the work of gradually fixing the meaning of the book's terminology.

The ideas put forward in this book are dense and at times, highly technical, but so very important. Brian has described it to me as written rather like a detective story, in which the same set of problems is explored repeatedly, getting closer each time to a complete and self-consistent picture. As he said, "…what I've written should be read more like novel than like a manual. What things mean will gradually take shape. Be patient".

As Brian says, be patient, dear reader. If you care about computer science, as a practitioner, theorist, or concerned citizen, this book matters for you. Its conclusions matter, even if parts of it ~~are~~ obtuse to you. So even if, as a programmer, you find it difficult to see why you should care if the theorists have

got it wrong, be patient. If you're a theorist, and you find Brian's critique at times irrelevant or founded in misunderstanding, be patient. If you're a citizen, and the technical details are off-putting, be patient.

If you *are* patient, and stay the course, when you get to the end you will be well rewarded by arguments elegantly made, important fissures in our understanding of computing spotlighted, and the way forward, clearer than before. You will surely have learned what a subtle and committed mind finds important. There is a vision presented in these pages; I know that I as a practitioner am asking myself what I can do to realize this vision. As citizens, we need to cheer from the sidelines, and keep asking these fundamental questions about computing, especially in our contemporary era of machine learning and AI. Our future depends on it.

Henry S. Thompson
Toronto and Edinburgh
November 2024