Im OUTPUT-Gespräch: Prof. Dr. Frederick P. Brooks

A Humanistic Computer Pioneer

Interview: Louis A. Venetz

Professor F.P. Brooks realised the first family of program compatible computers – the IBM 360. He is a professor at the University of North Carolina at Chapel Hill, USA, where his principal research area is developing color three-dimensional graphics tools for examining molecular structures, and virtual worlds systems. Prof. Brooks has received a degree of PhD h.c. at the ETH Zurich because of his fundamental articles and three books about software techniques (another book is in process). In this interview the reader will find a summary of his work and his message to students, scientists, managers and other people who are interested.

OP: *Prof. Brooks, what kind of machine was the IBM 360?*
Brooks: The 360 was not a machine, it was a family of machines. With the original development in 1964 there were seven machines. One was developed in England, one in Germany, five were developed in different laboratories in the US. The basic 360 concept was that all seven of these machines should be program-compatible from a very small model to a very large supercomputer model, all able to execute the same software. We developed software for these machines, and through the years there have been many revisions and successor generations in the 360 family. Today the IBM mainframes and the OS/MVS are still linear descendants of the 360 family.
OP: *Are you still developing computers?*
Brooks: No, I am not, I have some colleagues in Chapel Hill who are. In our laboratory Prof. Henry Fuchs and John Poulton have developed new graphics machines.
OP: *What is the meaning of Computer Architecture?*
Brooks: It means the properties of the computer that determine what programs run and what answers they get. It is the logical appearance of the computer as seen by the programmer, and so it's quite independent of the technology out of which the machine is made. It's the logical structure.
OP: *Do single-chip microcomputers (32 bit or more) have a chance in the development of computer architectures?*
Brooks: At any given stage in the technology you can get a certain number of transistors on a chip. You can use multiple chips to get many functions or you can use a single chip and put a small amount of memory and processor all on one chip. For things such as washing machines or elevator control, these simple smaller machines will be quite sufficient. So mostly wider computers offer an economic advantage. Yes, they have a chance.
OP: *You say: «Von Neumann was mostly right». Why «mostly»?*
Brooks: The ideas that are proposed in the original paper by Burks, Goldstein and von Neumann are to a surprising degree the ones we still use today. After having explored many alternatives we have come back to a standard architecture. Many features are the same of those of the original 1946 proposal. Now I say «mostly» because machines today have new ideas that have come in the years since 1946. But the ones that he had were, with few exceptions, fundamentally right as far as they went.
OP: *Prof. Zuse called this architecture a bottleneck. Do you agree?*
Brooks: The bottleneck arises from the fact that one has a memory and arithmetic units. The bottleneck is the path for data flowing between the memory and the processing units. In other words the memory bandwidth is the bottleneck. Memory bandwidth is the fundamental parameter determining speed. The computer architect's task is to take as much advantage as one can of the memory bandwidth.
OP: *Or you don't want to have a bottleneck at all. Do we need parallel computers?*
Brooks: That is a way to go. If one has many memories and processors independently then the bandwidth between the two is much wider. Now the question is how do you orga-

I'm always amazed and marvel at the capabilities of the human brain.

Porträt:

Prof. Dr. Frederick P. Brooks

Von Prof. Dr. W. Gander, Vorsteher der Abteilung für Informatik III C, ETH Zürich

Professor Frederick Phillips Brooks wurde im November 1991 die Ehrendoktorwürde der ETH Zürich verliehen. Er erhielt diese Auszeichnung in Anerkennung seiner grundlegenden Beiträge zur Softwaretechnik und Rechnerarchitektur, insbesondere der Realisierung der ersten Familie von programmkompatiblen Rechenmaschinen. Prof. Brooks wurde 1931 in Durham, North Carolina, in den USA geboren. Er studierte zuerst Physik an der Duke University und doktorierte anschliessend im Jahre 1956 an der Harward University in Angewandter Mathematik und Computerwissenschaften. Von 1957 bis 1964 arbeitete Professor Brooks bei der IBM. Er war der Projektleiter (der «Soft- und Hardwarevater») des sehr bekannten und erfolgreichen Systems 360. Seit 1964 ist er Professor an der University of North Carolina, Chapel Hill. Er gründete dort das «Department of Computer Science». Prof. Brooks ist ein Computerpioneer: Grundlegende Ideen im Bereich der Computerarchitekturen, des Software-Engineering und der interaktiven Computergraphik stammen von ihm. Sein bekanntes Buch über Software-Engineering «The Mythical Man-Month» ist seit 20 Jahren ein Bestseller.

nise those many processors to solve one problem?

OP: *And how tasks will be distributed, do you have a solution for this problem?*

Brooks: Well, there are many different kinds of solutions depending upon the applications. But I don't have any special new solution for that problem.

OP: *What is a supervisory program?*

Brooks: In the second generation, the first transistor computers also included the important concepts of program interruption so that a supervisory program can take control form a run-away problem program, memory protection so that the supervisor program can protect itself from a runaway problem program, a time clock so that one can stop loops. The supervisory program keeps control of the I/O devices. These are the facilities that were invented at that time to control machines instead of a human operator.

OP: *Why do you say that today the emphasis is too much on CASE tools?*

Brooks: I think the hardest parts of the software problem are the designing of the conceptual structures of the programs themselves rather then expressing them in machine languages. And so more and more work on the tools that aid expression will not make as much difference as more and more techniques for inherently better design will make.

OP: *You say that good programmers are much, much better than average programmers and that we have to look for them. But how do you judge them: who says, «This is a good programmer?»*

Brooks: To the first question: I think Jesus gave the answer for that a long time ago. He said «By their fruit you shall know them». You can tell by the output. Now, as to proper training, I think in a curriculum it is very important to provide the fundamentals in mathematics, in physics, in computer science, data structures, algorithms, representation. Then I think most of the development of the great programmer has to be done in the industry where it's a question of career planning and mentoring just as it is for developing managers in industry. Some careful attention to rotation of assignments, to breadth of experience, and to career development, has to be given to the very best technical people. Most companies are very careful about management development; most companies are not very careful about development of their technical power.

Prof. Brooks: the famous hardware and software father, author of the book «The Mythical Man-Month»

The advances in architecture will principally be made in the way of paralleling computers.

OP: *You revisited the book «The Mythical Man-Month» after 15 years. What are the results?*

Brooks: I think time has shown most of the basic concepts in the *Mythical Man-Month* to be valid. Some important things have changed, however. Now we know about building software by incremental methods, building a very small but usable version, and getting it into the field to get user feedback. This is a concept that I didn't know at the time I wrote that book. That is the most important thing that has changed completely. Also there are now a lot more data on how to make software.

OP: *How far is the research in your molecular graphics?*

Brooks: In the years since I have been in Chapel Hill my principal research area has been developing color three-dimensional graphics tools for examining molecular structures. Now the inter-atomic distance in a molecule is about one 5000th the wave length of light. One couldn't begin to see a molecule. But by making mathematical models of the molecules and then showing these with computer graphics, one becomes very familiar with the structures. It is as if you would see what in fact cannot be seen. In one of my lectures I talked about different aspects of the molecular graphics research program. The exciting thing today is that one can move, for example a protein, while doing real-time computation of the forces between the atoms so that it distorts in the proper fashion. That's something we haven't been able to do before because we haven't had the computer speed.

OP: *Do we see the reaction between the atoms H and O?*

Brooks: No, one cannot do those computations in real time yet. The bond formation has to be done at the quantum chemistry level and the quantum chemistry calculations still require supercomputers. Even they are not up to real time speed now.

OP: *Is this also useful for teaching?*

Brooks: Yes, we can see i.e. a molecule docking where the principal forces are van der Waal's forces and hydrogen bonding. But typically we cannot in real time modelling make or break covalent bonds.

OP: *Is the research in virtual worlds systems similar to modular graphics?*

Brooks: Yes, what we try to do

in real time three-dimensional graphics research is through the use of a head mounted display very fast image generation, and a mechanism for tracking the position and orientation of the head to make it possible to create an imaginary world. The user can move around and see and hear things that are at the wrong scale such as molecules or solar systems, things from different times, or things from distant places. This has been used by the NASA people for allowing an astronaut inside the shuttle to see and feel what an instrument is doing as it works on a satellite outside the shuttle. So it can be used for teleoperation.

OP: *Is this useful for the SDI project?*

Brooks: No one has gotten anywhere near that far in the SDI project.

OP: *Is the atomic kernel fusion influencing computer science or the reverse?*

Brooks: Oh, certainly the reverse. Without the supercomputer calculations one cannot begin to design fusion containment vessels. The magnetohydrodynamics involved are just incredible. Without computers it would be impossible to think of doing it in fusion. Now I think fusion has no special implication for computers except as providing another application.

OP: *What kind of architecture will be in the next century? Will there be another revolution of computers?*

Brooks: There will surely be another revolution of other computers. I can't tell you what it will be. I think the advances in architecture will principally be made in the way of paralleling computers. The individual processor will look much the same architecturally regardless of the technology. But the way in which they are put together – in communication paths, message passing, memory sharing, in synchronisation – those will be invented.

OP: *Do neuronal networks have any chance?*

Brooks: Notice that the field of research people call neural networks, is a very loose approximation to what goes on in the brain. It's a highly abstracted mathematical model. It has the disadvantage that the computational labor expands combinatorially in the number of neurons if you are not very careful. And this combinatorial expansion means that we can model on the fastest computers only very simple neuron systems. That kind of research is important as a way of understanding the way systems can be organised, but I think the practical applications will be not the next computer revolution.

Molecular Graphics: We can see a molecule docking where the principal forces are van der Waal's forces and hydrogen bonding.

OP: *What you think about optical computers?*

Brooks: It's a very difficult technology. I think first we will see optical transducers used for getting information off and on silicon chips rather than used for doing any computation proper. Optical memories are fundamentally delay memories. Their logical behaviour is very much the same as old acoustic delay memories or magnetic drums. And people know how to make them. But making large economical optical memories is difficult.

OP: *What do you think about biochips?*

Brooks: I don't know very much about biochips. I think the biological mechanisms for information processing are characterised by being a slow technology, very slow compared to silicon technology, but self-growing and self-repairing. That makes it possible to have millions of elements in a little space. So I am always amazed and marvel at the capabilities of the human brain. We will not in our lifetime see computing machines that begin to have the capabilities that the human brain has.

OP: *Do you think that we have in the next ten years computers to which we can speak like to human beings?*

Brooks: Ten is a long time! Notice that the problem is that in normal human speech we don't have to be as precise as we have to be in talking with computers. Even if the computer recognised and did exactly what I told it, I might be telling it the wrong thing. Because in human conversation what is to be done is in fact refined before we tell the person to go do it. We say, «I would like for you to do so and so.» «What do you mean?» This is back and forth. We will have to learn to speak more precisely. Just recognition won't solve the problem.

OP: *Why did you become teacher?*

Brooks: Well I am a follower of Jesus Christ. One of the things Christians do is to try to understand what their life should be from His point of view. And in 1964 when this possibility arose, it became clear that I should make this change.

OP: *What topics are you teaching at Chapel Hill?*

Brooks: Well I have taught almost everything in the computer science curriculum. The things I teach principally today are computer architecture, computer graphics, and professional communications-speaking and writing for computer scientists. I am also a teacher from time to time for the software engineering laboratory.

OP: *How would you summarise your life?*

Brooks: Well it has been my privilege to live through five computer revolutions. It never gets dull and it is always exciting both to see the new rapid advance of the technology and the spread of applications. And then working with students is always enjoyable. At the university people, not machines or papers, are the most important products.

OP: *What message will you give to young people, students?*

Brooks: The first message would be that computers are fun. The second message would be they are not ultimately important; it is people that are ultimately important, not the toys we build or the things we do with them. Jesus said: «What does it profit a man if he gains the whole world and loses his own soul?» And what does it profit a man if he builds machines as big as the Tower of Babel, which can do all kinds of marvellous computations, if in his own soul he becomes a small, dying person. It's much more important to become a real person and to go on living. It is people who live forever and not our organisations or our works.

OP: *And to scientists?*

Brooks: First, the scientist is always under special temptation to not be very careful with the truth. So it is important that the scientist is very careful with the truth. That is a professional obligation. Second, scientists, like engineers and other technicians, are sometimes so absorbed in the sciences that the other aspects of life are lost. It's important to keep one's total human perspective, to look at whole spectrum of life including the inner life, and the spiritual life.

OP: *Prof. Brooks, I thank you very much for coming to this interview. It's not very easy to ask the right questions to such a famous hardware and software pioneer. But I think further questions certainly will be answered in your interesting, edifying books and articles. Thanks again.* **OP**