

## **A Critique of IBM's Computer Science Research**

**Jim Gray, IBM Research, San Jose  
September 24, 1980**

I am resigning my position at IBM Research because it is seventy-five minutes from my home and I am a little tired of commuting. The commute time grows as San Jose's sprawl envelops IBM. IBM Research is planning a move deeper into exurbia in the near future. I'm not willing to Move to exurbia.

System R is completing the technology transfer phase and so this is a convenient time for both me and for IBM.

I had intended to go quietly, but my manager asked me to document my criticisms of IBM Research. These criticisms are not why I am leaving. I have had these criticisms for a very long time and have not left before. The reason I am leaving is the same one I had when I resigned from Yorktown research: I don't share IBM's passion for exurban life. I realize that most of my colleagues do like exurbia.

IBM has a very low turnover, the lowest in the industry. People who work here like it. If it were not for the commute I would stay here and continue my attempts to change the things I don't like about IBM. However, I have thirty years of work ahead of me. It seems I should move to a company located in a more urbane (if not urban) setting.

### **WHAT IS RESEARCH?**

Perhaps I should begin with a very personal statement: I aspire to be a scholar of computer science. All fields of scholarship, from religion to medicine emphasize three aspects: meditation, teaching and service. Meditation (called research by scientists) is the official part of research. But, teaching (writing papers, explaining your ideas, and transferring technology) and service (making computer systems and helping people use them) are also major aspects of the scholarly process. They keep the scholar in touch with reality.

Computer science is an empirical and multi-disciplinary field. The aspect of it that I work on, computer systems, requires lots of good people, time and equipment to produce anything of interest. Projects of five or ten people working for five or ten years seem to be about the right scale. More modest projects are unable to attack significant problems. More ambitious projects have unclear goals and have management problems.

Systems research projects attempt to fill the gap between what is and what is possible. Today, this involves making computers easier to use by trading performance for function.

### **WHAT IS IBM SAN JOSE RESEARCH LIKE?**

Computers and software are the things that computer systems are made of. The computer field is changing very rapidly. Research in systems must anticipate the hardware and software ten years in the future. It is essential that researchers have modern equipment to base their work on. Otherwise, the gap between what is and what is possible is even wider. If one has ten-year-old equipment, one fills the gap that existed ten years ago.

This memo is being typed on a terminal no longer made by IBM (the 3277) into a computer no longer made by IBM (the 370/168). This equipment was designed in the late sixties and came on the market in 1971. We are scrambling to get a new 370 (the 3033). When we get that processor, it will be obsolete.

IBM products are not very modern. Both Amdahl and Fujitsu make faster, more reliable and cheaper 370's. DEC makes much nicer small machines (the VAX has 32-bit addressing). There is much excitement inside IBM about research projects based on Motorola, Intel and Zilog chips. In some sense, working for IBM is a liability these days; there is so little neat hardware available internally. Even after the internal discount, IBM products are more expensive and not as nice as things available from other vendors. If you want to use Motorola chips, why not work for Motorola or Zenith? Where are the 801 chips?

We might do what most research labs do: build our own special hardware. In the past, we have not done that. The Rainbow display is the only hardware we have built in the last nine years. The Rainbow developers gave up developing graphics devices when it became clear that there was no way for them to get their ideas to the marketplace (Rainbow was not in IBM's product plans).

Somehow, hardware people have not found IBM Research a hospitable place to work. Xerox PARC in the mean time has built four different processors (MAXC, ALTO, DORADO, Do), several local networks, and a large variety of terminals, printers, and network interfaces. The Bell Labs Computer Science group has built two local networks (Spydernet, Datakit), a speech synthesizer, and two chess machines (the second will be a world-class player).

## MEDITATION

In 1974, confronted with the realities of archaic hardware, we picked a project that would not require new hardware (e.g., terminals, networks, or VLSI). Rather we chose to focus on using CPU+MEMORY+DISK. It was our projection in 1974 that these three technologies were on a track and that CPUs would get cheaper and faster, memories and disks would get much larger (but not faster). We also projected that the cost of programming would rise. These trends have not changed in the last eight years and will probably not change in the next eight. We then set out to make the accessing of data on disks more automatic, thereby increasing programmer productivity. This is the database problem. Our solution was System R. I won't bore you with the details but we had a hell of a time using the computers and tools we had available at the time. The facilities were really crummy. At the time, we were getting our MIPS from GPD.

System R was a success because it was a reasonable compromise with archaic hardware and limited computing facilities at our disposal. A similar project at Berkeley (Ingres) failed because it had even worse hardware (small address space PDP-11, small disk, simple operating system)

Members of the System R group are now going in three different directions: (1) distributed database, (2) database machines, and (3) text processing. Each of these new projects requires modern hardware (networks, special processors and fancy terminals and printers). Each of the projects is doomed unless the poor hardware base at San Jose is upgraded

I hope I have convinced you that the equipment at IBM San Jose Research limits the research topics we can address. Now for the software.

We have the languages: PL/I, PL/S, Pascal and assembler. Our experience with PL/I is bad (performance, ability to move to new systems and ability to transfer technology). I view Pascal as a PL/S look alike. I conclude that although PL/S is not supported by anyone it is the best language available to us.

All these languages were designed in the sixties. They lack many features, the most critical being an I/O library, type checking (at compile and link-edit time), type extension, exception handling, symbolic debugging, coroutines and tasking. The languages Mesa at Xerox and Ada from DoD cure many of these defects. However, although Ada has the force of the U.S. government behind it, no one in IBM seems even to take any interest.

It amazes me that NO programming language group exists in IBM (there is an optimizer group which has a PL/S-like language called PL.8.) No one in the Research division is working on making system programming easier for IBM or its customers. (There are application programming projects like QBE, SQL, Maple, ...) This despite the glowing reports from Mesa users on how much better Mesa is than any other language they have ever used. Paul McJones, who worked with us for a long time and left IBM for Xerox, assures me that Mesa would have made implementing System R much easier and would have caught many of the errors still lurking in the code. The people at Xerox have implemented BCPL and invented Mesa and Smalltalk languages in the last eight years as well as building nice hardware and applications.) The IBM computer science research effort, which is ten times larger, cannot match that record.

We run the IBM product text editors and text processors. They are adequate but nothing to brag about. The most interesting aspect of our text processing system, the equation processor that typesets equations, is copied from the

Bell Labs Unix group. I think it is fair to say that IBM Research has contributed nothing new to text processing in the eight years I have worked here. I believe this is because we have been hampered by poor terminals and poor printers.

The IBM electronic mail system is also adequate but nothing to brag about. The ARPANET community had a better facility earlier, and ours still loses mail at an unacceptable rate. (It lost all my mail last night.)

Summarizing, the hardware and software base at IBM San Jose Research is not very modern. In 1972, we had two terminals and a TSO/MVT system up eight hours per day (when it was up). Now we have the hardware and software we should have had then (the hardware we have now was on the market then). That is eight years progress in eight years. IBM San Jose needs to make ten years progress in the next two years just to catch up with the present. Otherwise, we will continue to solve the problems of the past.

One might ask the question, "Who's in charge here?". Why is a 700-person IBM Research group less productive than a 50 person AT&T research group or a 50 person Xerox research group? Not because we are stupid and they are smart (I went to school with those guys and they are just as stupid as you and me). I believe the answer is Bob Taylor (at Xerox) and Doug McIlroy (at Bell) who know how to hire and manage people. They hire good people, give them good facilities and make them do good work.

So far, I have talked about research. Now consider the other two aspects of scholarship: teaching and service.

We interact well with IBM employees here and in other divisions. We frequently travel to IBM locations, give seminars and learn about their problems. To a lesser extent we discuss things with IBM customers, although this is in part because being in IBM makes our relationship with customers very delicate.

IBM is inconveniently far from a center of learning (Cambridge, Stanford). This is a side effect of its passion for exurbia. However, we write papers, go to conferences and generally have good contacts with universities and other research labs. A major shortcoming is that we have very few students. Few students will come the distance to enjoy IBM's limited facilities.

Perhaps the most frustrating thing for me has been the technology transfer business. We have transferred System R to several IBM locations. However, our most successful transfer has been to Relational Systems, a company which sells a System R look-alike call Oracle. Oracle entered the market this year. It is nicer than System R in many ways. Why is it that IBM, to whom we gave both the code and years of consulting, is five years behind Oracle which started in 1977 with only the System R syntax and examples?

To give another example, all our ideas about distributed database are being implemented by Tandem. They credit us with the design. IBM is not planning to use our ideas until the late eighties.

Wheeler, Garnett, Watson and I worked very hard to put writeable shared segment sharing into VM. The code is installed and works at several IBM sites and is key to making System R work. We gave the VM product group the code, the updates to all the VM manuals, an FPFs, regression tests, and all the other boilerplate. All they had to do was include the files in the PID tape. But because the product (VS/QUERY) which was the planned user of this code folded, all our work had no official 'support' and a decision not to announce it was made at the last moment. Relational Systems copied our design and offers it as a fix to VM.

One must be a philosopher to do development in IBM, philosopher in the Latin sense of "love of knowledge" and philosopher in sense of being egoless. As several people have told me: "If you want to see your ideas and code get out, get out of IBM." This is a problem shared by all large corporations; Xerox has not let any PARC product out and AT&T can not figure out what to do with UNIX (they mail you a tape if you mail them \$30,000).

## SUMMARY

The difficulties of technology transfer are inherent in IBM's size. IBM is not an ideal place to do advanced development for that reason. But, most of the things I cite are easily fixed. The irony is that most people have agreed with me ever since I joined IBM. Leonard Liu worked very hard to improve our facilities and organization.

Frank King worked no less hard. Dick Kelisky has worked very hard to get us better computing. They all agreed with my analysis five years ago and probably still agree. Now Abe Peled sees the computing environment and lack of good hardware as a major problem. Due to all this concern, we have made great progress. But, I measure it as eight years of progress. We are now where we should have been in 1972.

I have several specific suggestions.

- The Research Division should create and cultivate Systems hardware groups like the 801 project in many areas (terminals, networks, disks, processors, VLSI, . . .) We won't get modern hardware from anyone else.
- The Research Division should encourage the building of software systems for IBM's internal use. We won't get modern software from anyone else.
- Don't replace me with a person, replace me with a fully configured 4341 for the exclusive use of the R\* project.
- The Research Division is becoming the Advanced Development Division of IBM. This will eventually drown out non-product related research. We should do what Bell Labs has done and create a Research Division within the Research Division.
- The Research Division should look at the Computer Science groups at Bell Labs (50 people now), and Xerox PARC (50 people now) and try to understand why they are so much more effective than we are.