Text 6
An Imaginary Tour of a Biological Computer
(Why Computer Professionals and Molecular Biologists Should Start Collaborating)
Remarks of Seymour Cray to the Shannon Center for Advanced Studies, University of Virginia

Seymour R. Cray earned a Bachelor of Science degree in electrical engineering in 1950 from the University of Minnesota. In 1951 he earned a Master of Science degree in applied mathematics from the same institution.

From 1950 to 1957 Mr. Cray held several positions with Engineering Research Associates (ERA), St. Paul, Minnesota. At ERA he worked on the development of the ERA 1101 scientific computer for the U.S. government. Later, he had design responsibility for a major portion of the ERA 1103 the first commercially successful scientific computer. While with ERA, he worked with the gamut of computer technologies, from vacuum tubes and magnetic amplifiers to transistors.

Mr. Cray has spent his entire career designing large-scale computer equipment. He was one of the founders of Control Data Corporation (CDC) in 1957 and was responsible for the design of that company's most successful large-scale computers, the CDC 1604, 6600 and 7600 systems. He served as a director for CDC from 1957 to 1965 and was senior vice president at his departure in 1972.

In 1972 Cray founded Cray Research Inc. to design and build the world's highest performance general-purpose supercomputers. His CRA Y-l computer established a new standard in super computing upon its introduction in 1976, and his CRA Y-2 computer system, introduced in 1985 moved supercomputing forward yet again.

In July 1989, he started Cray Computer Corporation to continue to expand the frontiers of scientific and engineering supercomputing. He was able to incorporate gallium arsenide logic design and micro-miniature supercomputers. The CRA Y-4 achieved a clock speed of one nanosecond.

Mr. Cray is the inventor of a number of technologies that have been patented by the companies for which he has worked. Among the more significant are the CRA Y-l vector register Technology, the cooling technologies for The CRAY computer, the CDC 6600 Freon-cooling system, a magnetic amplifier for ERA the three-dimensional interconnected module assembly used in the CRAY-3 and the CP-A Y-5 and gallium arsenide logic design.

Can Computers Think? Not Yet!

I remember about 10 years ago there was a lot of talk about artificial intelligence and writing a program that would learn. Particularly in Japan there was a lot of enthusiasm. Now that 10 years that have gone by, I hear less and less about it. I'm sure there's progress. There are some signs that machines are doing things kind of close to thinking, but I don't think we can say that we have a machine that learns today.

I suspect many of you followed, as I did, the recent chess match between Garry Kasparov and the IBM machine, I found that quite interesting on several counts. First of all, machines have got better and better at playing chess, and they are beginning to approach the capabilities of good expert humans. And this machine, the IBM machine, was especially designed to do the absolute best that we thought could be done with the computer.

So we had this chess match between the IBM machine and a world chess champion. It was for six games. They followed the rules of human chess competition. The chess clock was turned off and on for the computer, and the first game the computer won. And Kasparov was very impressed. So he sat up that evening studying how did he lose that match, what was the strategy of the computer. And what was the computer doing that night? Well, it was turned off in the corner.

So you know what happened. The computer didn't win another game! Garry Kasparov won three and tied two. Thus computers don't think yet, at least not chess computers.

Pedaflop Computing

Not long ago I attended a workshop, and it was called enabling technologies for pedaflop computing. Now, some of you may not know what a pedaflop is, so let me explain that, assuming that some of you don't. Along about 1960 I remember, because I was involved, we invented the player piano sequence and made the floating point in our computer, versus bringing a subroutine to do it. And from that time on we could say how many flops docs your machine do. Floating Point Operations, flops? How many flops does your computer do? And so today when we look at personal computers we say how many megaflops do they do? How many millions floating point operations per second?

People that can afford big workstations can say how many gigaflops does your computer do? That's 1000 megaflops. Well, that's enough for most people. But, you know, there's always a government laboratory that wants something bigger. And so we have one today.

It's the Sandia National Laboratory in Albuquerque, and they wanted a teraflop machine. And so they ordered one from Intel, and it's being delivered sort of piecemeal now, and by the middle of next year it's supposed to be all done, and it's supposed to run at a teraflop. And it has 9900 processors. It. is a real monster. And, of course, all the other national laboratories are very jealous and they say, well, it costs too much, $40-some million, it won't work anyway, who needs one? But I think it's kind of nice that we have a teraflop machine because I guess we needed one. I'm not quite sure. Anyway, that's a teraflop. Now, I think you know what a pedaflop is. A pedaflop is 1000 teraflops, and we're nowhere near to getting a pedaflop machine. But agencies like to talk about it. So they were the sponsors of this workshop. I was the keynote speaker at this first pedaflop conference. Now, they are annual. You know, once you get started you can do it every year.

And so I talked about revolution. I talked about where we might go in the future to build a pedaflop machine. And I talked about things like can't we use biology? And everyone smiled and said nice things, but as I listened to the other talks, everyone talked evolution. And what the group thought, and this is a group of probably 30 technical people. They were all supposed to be top-notch in their various areas, they said if we just keep doing what we're doing, in 20 years we'll have pedaflop. And they had a documentation to prove it. They had a straight line on semilog paper.

Now, you know how that works. I mean, anything is a straight line on semilog paper. And so what they'd done is they put two points on the chart for the last 10 years of progress in computers, and they just extended it for 24 years, and sure enough it came to a pedaflop.

Scaling Computers Down

Well, I got to thinking about what that might mean. How did we make progress in the last 10 years? We made machines faster, and we made them smaller, and if we keep doing that for 24 years, what size is it? Well, now we're building half-micron circuit technology. We're soon going to be building quarter. Perhaps, some people are now. And if I've talked to people that are doing research, they are talking about 0.15 micron technology. Well, if we extrapolate for 20 more years it's going to be really tiny.

Well, how tiny? How big is the molecule? Inorganic molecules are like a nanometer, but biological molecules are tens of nanometers. Well, 0.1 micron is only 100 nanometers. So we don't have far to go until we get down to the dimensions of biological molecules.

Let's suppose that this chart is right, and in 20 years we'll build silicon that has details of that dimension. I think that we're going to find that we're coming up against a couple of basic physical things, like the uncertainty principle, that those things will be small enough so they won't behave the way macro things do, and I think we'll be coming face to face with the life force, which I view as a factor here. So I want to talk about those things.

Inside A Biological Computing Facility

What I think would be real interesting today is if we take a tour of a biological computing facility. Now, you have to use a little imagination on this tour. I'll be the tour guide. I want you all to imagine that you are computer engineers, and my job as a tour guide is to translate for you the biological names that we're viewing so you will understand them as computer engineers. Now, there's another thing. You have to imagine yourself as being quite small, like, you know, maybe 1 micron tall, because biological things are really tiny. So if you're following me, I want to look inside a biological cell and try to identify those computing things which we can relate to our computers today with the name translations. Let's start with an overview. And let's take a human cell, because that's what we're studying most these days. Specifically, we're going to look at a human cell from the standpoint of how it computes.

For the overview, when we look in the cell, the first thing we see is a big DRAM memory in the nucleus. It's called DNA. Then we look around the cell, and we see there are several thousand microprocessors. They are called mitochondria. And if we look further at how they work, they all share a common memory and they have two levels of cache. Now, you may not believe all this, but wait till we get into the details.

Let's look first at the big DRAM memory. Well, it's packaged in 48 bags. These are called chromosomes. Now, as we look at those we are a little puzzled because there are some little ones and some big ones and some middle-sized ones, and how did that happen?

Well, when you think about it, this computing facility started with a very small memory, and it's been upgraded a number of times, and you know when you go to the store you'd like to get the biggest DRAM parts, but you have to go with what's available. And that's what happened with the biological system. It had to go with what was available at the time it was upgraded.

If we look further into the big DRAM memory, we see that probably the packaging isn't important. Forty-eight banks probably aren't significant. We can view the whole memory as one string of bits, a one-dimensional memory. And biologists, 1 think, agree with that today. And so how big is it? Well, it's six gigabytes. Now, that's very big compared to a personal computer memory today. That's big compared to even most workstations today. So this is a really big DRAM memory.