From the perspective of 25 years in the industry, the chief engineer of DEC's PDP-11 and cofounder of Encore Computer Corp. points out the winners, the losers and the classics.

A walk through The Computer Museum with GORDON BELL

The Computer Museum occupies a spacious converted warehouse on Boston's waterfront, facing a preserved wooden schooner and a series of glass and steel skyscrapers. The blending of old and new in the cityscape serves as a perfect backdrop to the museum, which contains both the relics of a machine age gone by and examples of technologies still under development.

The Computer Museum houses the world's largest collection of computer industry artifacts. C. Gordon Bell helped found The Computer Museum at Digital Equipment Corp., prompted by his deep involvement in the computer industry and a fear that all the interesting artifacts would be destroyed.

Bell earned his B.S. and M.S. degrees from MIT in the 1950s and worked as a DEC engineer from 1960 to 1966, witnessing an "exponential growth" in computer installations. During this time, Bell forecast the impact of home computers and saw his chance to make it happen. Preferring the challenge of larger machines, he instead engineered the PDP-5, -8 and -11 and helped set the standard for interactive computing.

Bell then went to Carnegie-Mellon University from 1966 to 1972, which in his words was "perfect timing — the beginning of the integrated circuit generation when things were moving slowly" in industry. He returned to DEC in 1972 to build the first VAX and bring in a new generation of minicomputers.

After serving four years on the museum's board of directors, Bell has now retired to become a permanent trustee. The museum now flourishes under the directorship of his wife, Gwen, and Bell has begun a new project. In July 1983, Bell founded Encore Computer Corp. with fellow minicomputer giants Kenneth Fisher of Prime Computer, Inc. and Henry Burkhardt of Data General Corp. Encore seeks to challenge the industry with yet another generation of powerful small computers.

Bell gave Associate Features Editor Amy Sommerfeld a guided tour of the museum, giving his own comments on the exhibits along the way.

In a way, The Computer Museum is just like a computer. We had a prototype to test whether it was a good idea and what the clientele would be. Only DEC employees and customers visited the museum when it first opened.

The Museum started up at the DEC facility in Marlboro [Mass.] in September 1979. It was totally DEC-sponsored, not public, although three-quarters of the artifacts were made by other companies. A lot of time was spent debugging what to show about the machines and what to say about them, namely: What's the achievement? Why is it here?

Then we solicited "customers," and in June '82 went public with a board of directors. We solicited members and became, in effect, a production model. The second production model is The Computer Museum here at Museum Wharf.

The Hollerith Tabulator, used in the 1890 US census

> *One hundred years from now I want people to come here and say, "Gosh, I'm glad they saved all that stuff." By then they'll understand that information processing is one of the fundamentals of society.*

The goal of the museum as I saw it was to collect the first object of a given class, the last object of a given class and then the important ones — the classics. The fun is trying to find out: When is something going to be classic? When is something going to be the first one? I always tried to err on the side of collecting more — ones that I thought were really going to be important.

At the entrance to The Computer Museum stands Whirlwind, an experimental computer started in 1945 at MIT that eventually yielded the first core memory. Only one model of this 16-bit computer was ever produced; it operated from 1950 to 1959.

Whirlwind was the first real-time and control machine. It's here in part because it was the origin of the machines that came out of the New England region. It's a classic mini — as big as a house — and it has lots of firsts, including parallelism and real-time, interactive I/O.

Whirlwind was a controversial project because the machine took longer than they thought it was going to take to build, and they spent quite a lot of money doing it. But once it was up and people started using it, then everyone began to see the benefits of having a fast machine like this and what it could do compared with the traditional [John] von Neumann-style calculating machines of the time.

MIT conceived Whirlwind as a simulator for aircraft stability. That was one of the reasons it ended up with a short word length. Machines that were being built around this time tended to have 36- to 40-bit word lengths, according to von Neumann's guidelines. Whirlwind's engineers built a 16-bit computer because that was all the precision they needed. All the other machines were serial and slow, while this one was parallel and very fast.

One feature of experimental machines is that you never know exactly what you're going to get out of them. The MIT/Forrester patent for core memory came out of this project. The standard Williams tube memory in use at the time was so unreliable that the Whirlwind designers said, "We've got to have a new memory." Core memory was first tested on the Memory Test Computer [MTC], which [DEC President] Ken Olsen engineered. The MTC ran for about a month. The memory operated so well that the engineers just took it right out and put it on Whirlwind.

Around the corner sit several large pieces of equipment that together make up the U.S. Air Force's AN/FS Q-7, developed by Jay Forrester and Robert Everett of MIT's Lincoln Laboratory. Installed in 1958 and decommissioned in 1983, the 32-bit Q-7 ran longer than any other computer, and was the first to serve 100 simultaneous users.

Whirlwind also ended up being the prototype for the Semi-Automatic Ground Environmental [Sage] air defense system computer, called Whirlwind 2. Later, IBM built it under the name AN/FS Q-7. MIT helped design the architecture and the circuits, and then IBM built these massive vacuum tube machines. This was a 32-bit computer, designed to do everything Whirlwind could do and more.

It was a lovely machine because it had two 16-bit words that could be operated on in parallel. Each pair used 55,000 vacuum tubes and took 150,000W of power. The machine you see here in the museum was decommissioned only two years ago, in February 1983, and still ran at a phenomenal 99.95% uptime because of careful design and an absolutely controlled environment.

Notice the way it's built — a constant stream of air blows on each tube. Every tube is running at the same temperature. In addition, the users did something called "marginal checking," which meant they varied the voltages up and down to detect whether a tube was going to fail. By the time this machine was built, its designers really understood how to build very high-reliability computers.

On a museum field trip, we saw the AN/FS Q-7 before it was decommissioned. People operated the computer from this console of lights and switches. Today you can't see what's happening on a computer, but in the early days of computers there was a light on every bit.

You flipped switches to compute data, and you could see everything that was happening inside the machine. If the machine stopped or you wanted to run it slowly, clock by clock, you could see the whole state exactly.

I have programmed in machine language, bit by bit. In fact, until you get that first level of software on machines, you have to operate all machines bit by bit.

Core Memory Unit 2 from the U.S. Air Force's AN/FS Q-7

Core Memory Unit 2 from the AN/FS Q-7 stands 6 feet tall. It was considered a very fast memory: Any word in the Q-7's core memory could be accessed in 6 msec.

This refrigerator-size cabinet holds half of an arithmetic unit. This memory was one of IBM's contributions to the project. It stored only 4K by 32 bits — 131,000 bits, or one half of today's 256K-byte chip, which as you know is a very small fraction of the size. Later, they had a 64K-byte version, but this was really quite a small memory. That's why they needed all the drum units, which were used to swap programs with core. To show you the scale, each of these large drum units equals roughly one small floppy — about 256K bytes.

Getting rid of all the poor memories and switching over to core was a major transition. It occurred in the late '50s, even though the core was first operational in '53. It took that long to get core into other machines.

Cores hit the market simultaneously with transistor circuits, and that occurred almost precisely in 1960 — the beginning of the second generation of computing.

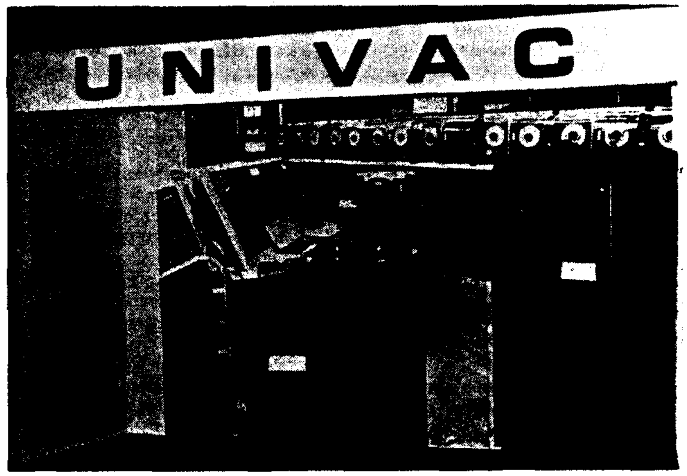
The year 1960 was a wonderful year, when a tremendous number of classic machines came out. Many were transistorized, and they all had core memories. That year was the beginning of serious computing. Reliable machines, relatively inexpensive, fast machines and good memories.

That's what really made computing start to gr[illegible] exponentially.

For many Americans, computers in the 1950s were synonymous with Univac. "That marvelous electronic brain" was first introduced to the general public by CBS newscaster Walter Cronkite during the 1952 presidential elections.

This is the Univac 1 that the Eckert and Mauchly company built. It really was the first commercial computer. When I say "first," I have to be careful, particularly saying the "first commercial." There were a couple of computers already operating in England, such as the Leo computer, but it's very hard right now to pin down when those were actually shipped.

When you say "first" you're asking, "When was it that a customer had it in his site, actually using it?" You have to read all the fine print.



For many Americans, "Univac" was synonymous with "computer."

There were 46 of them, which at the time was massive volume! The price was about $930,000 initially, and it declined over time.

The way to really see machines is to see how they were used at that time. The films the museum preserves and shows are really important for just this reason. They show, for example, what key punching was really like or how Eniac was used. Here's the first film on programming, and the first AI film and one on the introduction of Fortran. We also have a film made for the museum just when the last IBM Stretch was taken out of service.

The museum has a videotape of Walter Cronkite talking about the first time Univac predicted the 1952 election results. During the election, there was concurrent reporting about the election and the computer's handling of it. I remember there was a very different attitude than you see today, when everyone says, "Computers have really fouled up elections. Computers shouldn't be allowed to predict results because that will influence the voters," and so on.

The response then was amazement, absolute amazement: "How can this thing know what's going to happen after only a few hours?" The film the museum has of Cronkite's announcing doesn't quite match the amazement of the moment.

This machine was literally telling us what was going to happen. In fact, it seemed so eerie that the networks were refusing to use the results at first. The computer made an early prediction, and the networks didn't even put it on the air until later on because they just didn't believe it.

Museum visitors can operate this card punch and automatic sorter.

Several exhibits show the evolution of card I/O technology, from the original semiautomatic sorters in solid oak cabinets through the standard automatic sorters still found in universities nationwide to the final models of the card era. A small pile of tiny 96-column cards remains from IBM's System/3. They never caught on, and IBM introduced the first floppy disk the following year.

I was fortunate enough not to deal with cards much. I did one year as a Fulbright scholar and used cards all year. I swore I would never punch another card.

Then I went to Carnegie-Mellon University in 1966 as a professor, and they had an IBM machine with cards. I decided to write a book instead of computing — there was no way I was going to put cards in a hopper again!

I was spoiled. I had just built the first time-sharing machine at DEC, so I really didn't believe in batch processing at all. All the DEC machines were interactive, and we believed in having people talk directly to computers.

> ”
>
> *Technology is a driving devil. It conspires, and if there's a concept half-there or a computer half-designed, technology will complete it.*

But the general level of user-friendliness was still quite low at that point. The Apollo Guidance Computer here was used in the first Apollo space vehicle in 1962. Unfortunately, somebody took a piece off it, so we had to cover the console with plexiglass.

Below it, a [Hewlett-Packard] 150 computer performs the same function as the Apollo. When people play with it now, they say, "Oh, this is awful. The human interface is terrible." We answer, "Yeah, that's the way it was!" They ask, "How did they ever really control the spacecraft?" With great difficulty!

Also while I was at Carnegie-Mellon in the early '70s, I went to a seminar on IBM's minicomputer. It was odd — they had a System/3, and on it was this card reader with these little, nonstandard cards IBM was introducing. And I thought, "Oh my God, don't they know? Cards are dead!"

What happens in every technology is somebody tries to make the ultimate version, and it's an absolute disaster. These cards are a perfect example. Just when it was clear that there was no use or need for cards, they introduced these new 96-column cards. If they weren't as big, the logic went, you could have a smaller card reader and it could be cheaper. That was all the little cards had to recommend them.

The trick in any technology is knowing when to get on the bandwagon, knowing when to push for change and then knowing when it's dead and time to get

That's getting on the right bandwagon. The other trick is not to get on any wrong wagons.

The Viatron 21 terminal is an example of getting on too early. Viatron introduced a CRT, a processor, a keyboard — a whole data entry device for $40 per month, which was absolutely unheard of. A $1,600 device in 1968! The museum has a copy of the advertisement that ran in *The Wall Street Journal.*

This was one of the famous fiascos in selling. The company went public and sold stock, and the stock prices went out of sight because of this revolutionary data entry device. The problem was it was too early. You couldn't build it using the MOS technology they had then. They sold thousands, but they couldn't deliver any! The technology was too immature.

IBM's nonstandard punch cards were quickly abandoned.

A glass case packed full of artifacts — components, posters, books and sketches — fills one wall of the museum.

The purpose of this exhibit is to mark a period, 1950 to 1959, and to show a range of firsts, from basic technology to applications. The exhibit shows a complete census of all the machines installed by 1950.

As time went by, you can see there was an exponential buildup of computer installations. About 10 machines were installed during '51. They were all prototype machines. Twice that number were installed in '52 and twice that number again in '53.

There's another theme that's important. A time period of approximately 12 or 13 years shows up over and over again in the development of computing. It shows that things really don't change that fast. For example, it took that long to get the transistor into computers in full scale.

In this museum case lies the patent for the first point-contact transistor, which was filed in June 1948. By 1960, all the machines were transistorized, but that was a full 12 years from the invention of the device. Twelve years of hard work and production so you could produce the transistor, so people understood them, so the circuits got done and so on. It just took that long.

In 1959, the Noyce patent was filed on a new way to build transistors — the planar process. That was the beginning of the integrated circuit, but they weren't really produced until '67 to '68 — sort of a half-cycle. On the other hand, IBM's first integrated circuit computers didn't appear until 1973. That's a full 14-year gap.

In 1960, as the exhibit shows, there was an incredible number of new machines introduced, marking the second generation: the Control Data Corp. 1604 and 160, the beginning of CDC; General Precision's new machine; Sperry Rand's solid-state machine, Univac; Philco's transistor machine that put the company at the forefront; IBM's workhorse, the 1401, plus the 7070 and 7090, a real classic; and the DEC PDP-1, the beginning of DEC.

These machines formed the basis for the next 10 years of computing.

That was also the time when I said, "We're not going to have any more modified, kludgy typewriters on our computers." The next machine I designed had a Teletype on it. The next one after that was when we started using the ASR33.

We were the first ones to adopt the ASR33, which turned out to be a major product in marketing minicomputers. For $750 you could include a keyboard, a printer, a paper tape reader and a paper tape punch. Basically, we'd scaled the I/O problem down to something trivial. That's how DEC was able to introduce the PDP-8 at the $18,000 level, because we didn't have to charge $5,000 for a paper tape reader and punch.

In the same case, artifacts from the Atlas project include only a single board and a magazine article about the breakthrough by engineers in England.

Another fascinating introduction during this early period was Atlas, designed at Manchester University. I saw it in '61 and the r eum has

The Computer Museum starts up a rare treasure hunt

The Computer Museum launches worldwide search this month for rare artifacts to beef up its collection of early personal computers. Catching the spirit of the industry, the museum on Boston's Congress Street has turned the artifacts drive into a competition.

Winners will be those who turn up production units with the lowest serial numbers. Another winning category is exotic machines: the first Chinese personal computer, for instance, or the first Indian Speak 'N Spell.

"It's really a hunt for buried treasure," says museum President Gwen Bell. "We're not sure what's still out there." Bell says she hopes the drive will alert the worldwide community so that no more important artifacts are lost or destroyed.

"We are looking for low serial numbers, for early personal computer programs, for all kinds of memorabilia," Bell explains.

March 31, 1986 is the deadline to submit artifacts, in a campaign that will be advertised in newspapers around the globe. Prizes are nonmonetary, possibly including an expense-paid visit to the museum.

The Computer Museum does not stop at commemorating the past; it also gives visitors a look into the future. The personal computer and graphics exhibits offer the chance to experiment with advancing technologies, new products and even experimental devices.

"In the commercial world, there aren't places to come and play with new computers or peripherals without getting a sales pitch," says Gwen Bell. The museum provides a nonthreatening and nonmonitored atmosphere. "Visitors can experiment with a new technology that may not have an obvious application," she adds. "By playing with the device, you begin to think in new ways and dream up uses that you wouldn't have thought of without the hands-on experience."

The museum will also host a 6,000-sq ft exhibit called the Computer Discovery Center, created by the Boston Computer Society. Set to open next summer, the center will demonstrate the more sophisticated capabilities of microcomputers.

"We wanted a kid's corner, but we also want to show off the high end for the adults," Bell says.

"We intend to secure the museum's future with an incredibly good collection," she continues. "Displaying the most significant artifacts gives the museum a three-dimensional aspect. Visitors expect to see things they couldn't see anywhere else."

ome artifacts from it. Atlas was the irst virtual memory machine, using aging.

Again, the 12-year time delay for major product introduction: Atlas ame out as a research machine in 61, but Manchester's first machine an in '49. It took them that long to ind that two-level store is what you vant as a programming environment. DEC started building computers in '60, and by '73, we had a good virtual memory on the PDP-10. We were building minis — or what became minis — in 1966, and the PDP-11 had a good virtual memory on it by '78, when VAX was introduced — 12 years again.

In the semiconductor arena, the first processor on a chip was done in '71, and there still isn't a really good virtual memory microprocessor. National Semiconductor Corp. had a good chip set (the 32000) by '83, but they're really just delivering it now.

The idea of paging was written up in about five papers. The whole need and motivation for it was totally described by 1962. Anyone who had any feeling at all for computers could look at the concept and say, "Oh yes, this is the way you have to structure memory."

I don't know why it's taken so long to realize the concept. One thing is, all these industries — mainframe, mini and micro — are somewhat independent of one another, and it's unclear whether they learn from each other.

In fact the PDP-6, which was the first time-sharing machine and came out in '66, was designed for Lisp, because Lisp came out in 1959-60. I was convinced Lisp was really going to take over, but it's taken 25 years to catch on. It's amazing how long it's taken to get to a point where people really see the virtue of it. Lisp really is the most elegant language.

Unix was first developed in reaction to [General Electric Co.'s] Multics, and it was implemented on a PDP-9 and then on a PDP-11. Then when we built the VAX, we got [AT&T Unix developer] Ken Thompson to do a Unix critique on the VAX. We asked, "Can you run Unix really well on it?"

People always perceived this war between DEC and Unix, and that isn't the case. I certainly tried to make sure VAX was the best Unix machine. AT&T was a super customer, and I believed that Unix was going to take on about the position it has taken on, simply because of the open architecture and portability aspects it has.

The glass case also contains handwritten notes from some of the industry's leading technologists.

The work for An Wang's core memory was done in '48 and '49. This is a shift register that he built for the Harvard Mark IV, which stored 64 bits of data, and those are his notes — beautiful notes.

I think there's a good story here. Wang himself is a scientist-engineer, and I really believe you have to have that kind of leadership to build technological companies. Wang Laboratories is an excellent example of a strong company with a technically oriented leader.

DEC is another good example. Ken Olsen] was involved in the MTC at the outset and went on to work on Whirlwind, then TX-0. Apollo [Computer, Inc.] was started by very strong technologists. CDC, Cray [Research, Inc.] and many others have very strong technological roots.

But the company that's the most amazing to me in every respect, of course, is IBM. To me, IBM is a two-culture company: the very strong group that runs engineering/manufacturing and the field organization that markets their machines. [Thomas J.] Watson [Sr.]'s incredible drive for excellence set the tone.

The interesting thing is that a marketing person runs the company. To me that's a real exception. It's very difficult for a nontechnologist to run a technology company, independent of whether it's computers or bioengineering or any other field. If the technology is moving at all fast, then management has to be able to make decisions based on what's going on in the technology.

Apple [Computer, Inc.], for example, I consider more of a marketing phenomenon. With the exception of the Macintosh, I don't really regard Apple as a technology company, because the Apple I and Apple II weren't so much technological innovations. The first personal computer is right here at the museum, and it's not an Apple. A lot of companies had built small machines at the time.

Personal computers of all shapes and sizes crowd the PC Gallery. Whereas many regard the personal computer as a relative newcomer, some of these machines have the look of old-guard computers.

In the PC Gallery we have one of the Lincs [Laboratory Instrument Computers] that came out in '64 and which I think of as the first personal computer or scientific workstation. It had a personal filing system, keyboard and interactive display, and it was transportable. It cost about $40,000. Linc marked the beginning of a line of computers that included the Linc-8 and PDP-12 for personal, scientific and interactive computing. There are still Lincs in use.

Linc has all the attributes of a personal computer. It's for one person, it's interactive, you can go automatically from program definition to execution without any intermediate paper tape or cards or anything like that. But the main thing is it was used by one individual.

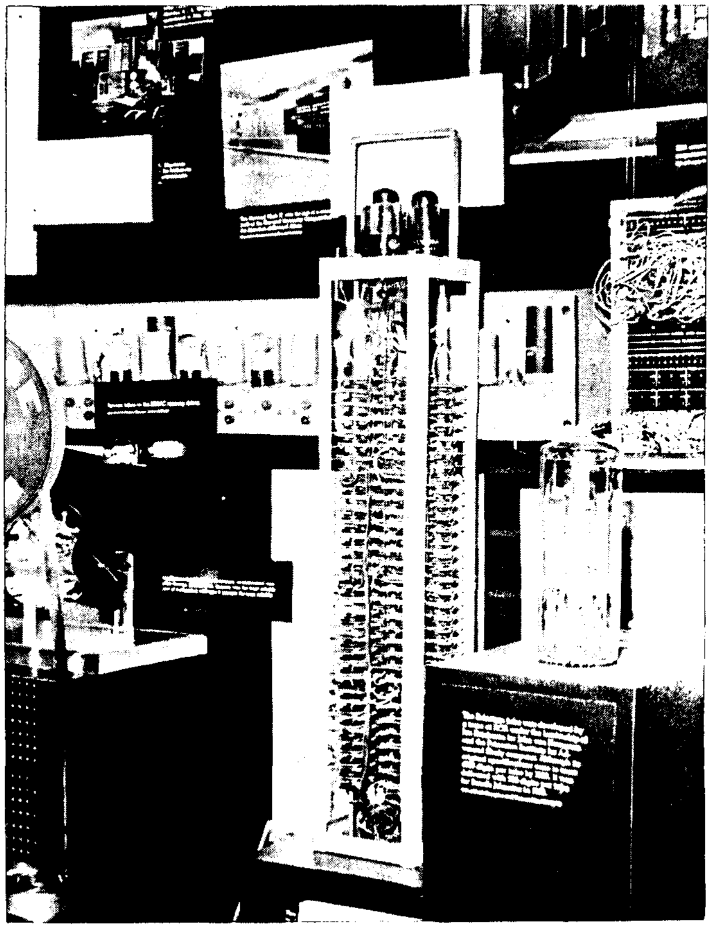
I think the issue of defining a personal computer is really one of scale. How much are you going to pay for a computer for one person? And what does it do?

The purpose of this exhibit is to display things you can't see in stores or in schools. It includes the first personal computers, like Linc and Altair, Apollo's first workstations and other artifacts. All the machines should have their skins off, their insides exposed. Computerland's Bill Millard, who is on the museum's board, has given a grant to collect and to enhance the exhibit. The main thing is to have a definitive, scholarly collection.

From the outset, personal computers were driven by memory technology. In 1975, a 4K memory chip was introduced, and the Altair was built using first a 1K and then the 4K chip. In 1978, the 16K chip was quickly incorporated into the Apple II. In 1981, the IBM Personal Computer came out using the new 64K chip, and then in '84, the 256K chip begot the Personal Computer AT and the Macintosh.

Furthermore, I don't believe anyone really invented the personal computer. "Invention" is too strong a word for it. A lot of things are called inventions when, actually, they were inevitable. I believe technology is a driving devil. It conspires, and if there's a concept halfthere or a computer half-designed, technology will complete it.

In retrospect, for example, I don't look at the microprocessor as an invention. It was something we were



Artifacts such as An Wang's core memory fill two glass cases, marking the periods 1950 to '59 and 1960 to '69.

A personal computer gallery displays all shapes and sizes.

all trying to do for a number of years. One day the technology reached a point where it could be done. In this case, it was a conspiracy between a good chip and adequate memory.

Apple happened to be the first to put that combination into a machine. I don't want to discredit them totally and say, "Oh, they were just a bunch of assemblers." They did a very nice job. The Wozniak disk controller was a very neat little piece of logic. But it was the 6502 processor, the 16K-byte memory chip and that disk controller that conspired, along with the idea of open architecture, in the first Apple computer.

Apple did a very neat job in pulling the pieces together and packaging the computer. If you read all of [Steve] Jobs' accounts, he really worked on the packaging. The key decisions were the user interface and good bit-map graphics. But I can show you all that same work at a laboratory at Xerox Corp.'s [Palo Alto Research Center] four or five years earlier. It was just waiting to happen.

I'm strictly an evolutionist. Get an idea and keep working on it. In the computer industry, we're not idea-limited now, it's just a question of pulling the ideas together. The machines that we can build now with the new technology are fantastic. We're anticipating machines that will execute 100 million to 1K million instructions per second.

The Computer Museum honors Seymour Cray with his own exhibit, titled "A man and his machines." Museum curators name Cray the "undisputed leader in the design of the most powerful computers."

Cray has built the world's fastest computers for 20 years. That's absolutely amazing! He has also produced an incredible string of ideas and basic technology. The reason he has been able to stems from his breadth, starting with the basic physics of the devices, of cooling, of wiring and computation . . . on into knowing how to build a compiler and operating system.

If you look at Cray and what he's done, you end up with a lesson on how to stay out of organizations. People get sucked into them. Cray stayed out of large organizations: first at CDC, by getting out of Minneapolis and going to Chippewa Falls, Wis. It was far enough away that people weren't coming to visit him all the time. He couldn't go to meetings.

He could never have built the 6600 in Minneapolis, I'm convinced. And then as Cray Laboratories grew, he must have seen the same thing happening again and said, "Gee, I've got lots of organizational responsibility, and the way to handle that is to split myself off again."

Organizations, no matter how tenuously connected, all start sucking up your time, and basically people don't have enough time for both computers and organizations.

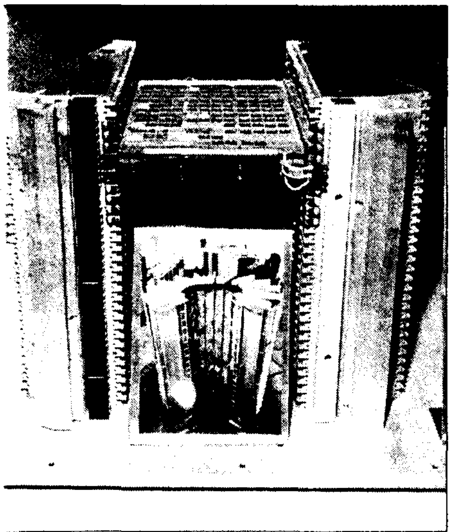
In this case, if you look at the

> **"**
>
> *A small crew is a prerequisite for designing really good machines. That's the nature of great computers — it isn't always the people with the most resources who succeed.*

Cray-CDC split from CDC's standpoint, the tragedy was letting him go, not being able to give him the environment he needed. But maybe it was inevitable. There's a discomfort that settles in with certain individuals in large companies. You suddenly see that it's really you who are supporting the company.

The Cray exhibit is dominated by the hulking remains of the first CDC 6600. Introduced in 1963, the 6600 was a product of Cray's Chippewa Falls lab and ran three times faster than IBM's Stretch.

CDC's 6600 No. 1 — a Cray brainchild — is preserved here. When the 6600 was announced, I remember being just awestruck by it. I put it with Atlas as one of the greats. In the development of ideas and projects at that time, these two stood out from everything else.



section of the Cray 1 on display.

The 6600 represents special creativity in a umber of aspects: It executed many instructions imultaneously, and they were all interlocked. ray had the idea of separate I/O computers and, f course, his [reduced instruction set computer Risc)] architecture. For the 6600, they had volved the circuitry enormously. This was the astest machine running at the time, with a very espectable clock time even by today's standards — almost a 40-MHz clock. The 6600 was also the irst machine to employ Freon cooling.

I love [IBM President Thomas J.] Watson Jr.'s omment about the Cray 1 announcement in '63, osted here: "I understand that in the laboratory leveloping the 6600 there are only 34 people, ncluding the janitor. I fail to understand why we have lost our industry leadership position by letting someone else offer the world's most powerful computer." That says it all!

And in fact, having a small crew is another prerequisite, I think, for designing really good machines. That's the nature of building great computers — it isn't the people who have the most resources all the time who succeed. I think while you are designing and building, you can't deal with the complexity that a large organization implies. It's very hard to segment the work and design all the interfaces so the design comes out right with a large crew.

To me, that's always been a challenge when you're building something large — to develop an architecture that will let the work be partitioned separately and independently among a number of groups. You want to get as many good ideas into the design as possible, yet not thwart it by the complexity of having every project member have to talk to every other one.

That's why I really believe in a strict, levels-of-integration model for segmenting work. That is, we work in very well-defined layers, starting from silicon or semiconductor devices up to hardware systems, operating systems, languages and then generic applications. Each layer has to be very cleanly structured so team members can build and develop them independently.

In a videotape the museum shows of Cray at Livermore National Laboratories, he talks about the size of the design team. He says his ideal number is one. Then he goes up to about six, each of whom leads one level, and then a layer of workers. Then he goes up to the next step, getting to about 30 in a hierarchy. But there's got to be not more than three or four who really understand the whole. That's assuming you've got one person who can lead the project and understand everything from semiconductors to applications.

The museum's collection includes production Model 17 of the PDP-8, introduced in 1964 at $18,000. The PDP-8 was designed by Edson de Castro (now president of DG) and engineered by Gordon Bell as an outgrowth of the PDP-5.

The PDP-8 was the first minicomputer. The reason it can be called a minicomputer is that it was built small enough to fit in a cabinet, and therefore it became a component to other systems. Furthermore, it was fast and easily mass-produced. The PDP-5, its predecessor, came out about two years earlier. It was the forerunner of the PDP-8, but I don't classify the PDP-5 as a minicomputer simply because no one integrated it with other systems.

The PDP-8 was implemented in a number of other technologies. By 1978, it was on a single chip that Intersil [Systems, Inc.] built. In fact, the number of sales of the PDP-8 has been higher in the last three or four years than at any other time because it's inside a word processor — the Decmate. So this one machine has lasted 20 years. Not bad!

The PDP-10 is also now about 20 years old. The PDP-11 was introduced in 1970, and later the VAX-11 was created to extend the PDP-11's range. There are still a lot of PDP-11s being sold, but VAX has really overtaken it as the main revenue source at DEC.

I personally made a decision in 1975 not to work on the first personal computer. I could see it was going to be quite a machine. But I went to work on the VAX instead, which to me was a much more fascinating engineering problem.

I wrote a memo in 1969 when I was at Carnegie-Mellon, urging a strong effort in home computers. It outlined the whole home computer industry essentially the way it has evolved. I have never been that interested in the smaller machines. Having worked on the PDP-8, I've always gravitated toward larger, more complex computers.

If I had decided the other way, I probably would have tried to build personal computers within DEC. In fact, DEC had a number of early personal computers, but the company just never marketed them.

At the time, the organizational model rejected the cost structure, the culture — everything about personal computers. The organization at DEC was aimed at a certain kind of machine, and anything that didn't fit that model didn't fly.

DEC actually did put a number of PDP-8s in people's homes. I had a time-sharing terminal in my home in 1966. Our children grew up programming on a terminal. For a long time it was a curiousity. Guests would ask, "What do you do with this thing? Why would you want one?"

The semiconductor gallery includes junked semiconductor boards, a 1970s rubylith mask and several important chips under a 50X microscope: a Mostek 4K-byte random-access memory, an IBM 64K-byte RAM, an NEC Corp. 256K-byte RAM and a DEC Microvax II processor chip.

I think the semiconductor industry people will throw everything away if we don't stop them. It's important to preserve their revolution because it's really been the basis for ours.

This exhibit gives a close-up of another lesson in technology. I feel the Microvax II is the best microprocessor chip or chip set made. It includes a one-chip VAX, a floating-point chip and memory management — it's a complete microprocessor. I came back to DEC partly to get the company involved in semiconductors. That was in 1972, and it took 13 years, but we succeeded.

Here's how: DEC produced what I call the first real computer before a semiconductor company produced a real computer by putting the VAX on a chip. I say the VAX is a real computer!

My criteria are the following: Does it have floating-point? Does it have a paging and memory management unit? And is the machine capable of being used to help design itself?

Now instead of being in competition with it, let's say I'm encouraging the semiconductor industry to build better microprocessors, because Encore is predicated on using lots of microprocessors to gain power. Our whole architecture rests on parallelism using a number of processors — a new computer structure called a "multi," for multiple microprocessor.

Our success depends in part on getting good components, so it's to my advantage to encourage the semiconductor guys to come through with good microprocessors. And that's beginning to happen.

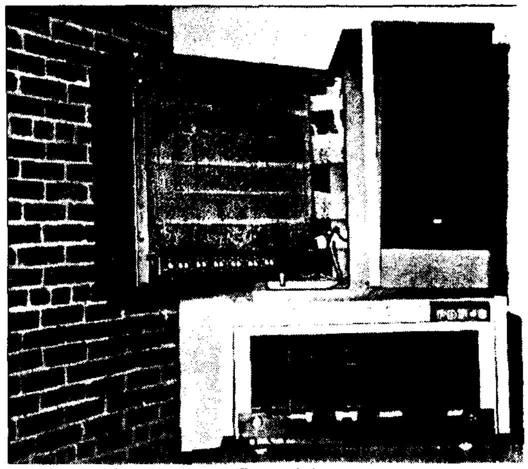
Despite Gordon Bell's official retirement from The Computer Museum board of directors, his interest in and proprietary sentiment for the museum remains lively. Even his necktie is imprinted with the museum's logo.

By 1990, my own personal goal for The Computer Museum is to collect every major artifact. We would need a bigger building, with enough space for storage and archiving. What I'd really like is to merge with the American Museum of Natural History and throw all the dinosaurs out. But I don't think it'll happen . . . it's not natural history!

Right now the goal is to increase people's understanding of the present and future, instead of focusing on history. The museum shows the incredible versatility of the computer. History by itself is too dry.

How do you measure a museum's success? You measure attendance and the attendants' response. But my own measure is the collection of artifacts, including the archiving of works and lectures by the pioneers.

One hundred or 200 years from now, I want people to come and say, "Gosh, I'm glad they saved all that stuff." By then they'll understand that information processing is one of the fundamentals of society.



DEC's PDP-8, the world's first minicomputer.

Encore's "multi" challenges VAX's price/performance

C. Gordon Bell helped found Encore Computer Corp. in July 1983 with the purpose of creating a new type of computer: the "multi." Two years into the project, he describes how the new product will fit into the ever-changing marketplace for powerful computer systems.

At Encore, we're predicated on making computers out of the current microprocessor technology. We will use microprocessors as the base-level component to build bigger systems. We call our computer the "multi," and I claim it's a new machine class.

If you look at computers in terms of technology generations (see chart below), integrated circuits caused the birth of minis; microprocessors and memories led to personal computers and workstations. Ultimately, they will also lead to multis. If I'm right, in five years everyone will build that way.

Over the last few years, a line of [transistor-transistor logic]-based machines has come out that show a 15% per year compound performance increase. [Emitter-coupled logic (ECL)]-based machines' performance is roughly two to three times higher than that. Of course, you can make bigger ECL machines, like Cray supercomputers and so on, but I'm speaking of maximum minicomputer or mainframe product lines.

The Microvax II has roughly the same performance as the VAX-11/780, on a MOS and Cmos technology line. These MOS-based machines show about a 40% to 50% per year increase in performance.

> *The multi provides substantially more computing power than any other machine in its price range. The architecture uses a common bus with several microprocessors. It's a very simple machine, built almost like a DEC PDP-11 with Unibus.*

But performance scales don't show the other dimension: cost. The Microvax II contains two chips, whereas the 11/780s might use 20 boards to achieve equal performance levels. So the Microvax obviously has higher performance per dollar by probably two orders of magnitude. The multi will do even better.

Machine classes come out of price ranges. The mini is what happens when you operate a machine yourself for a group. A mainframe is something that you get somebody else to operate, so it's a service. A supercomputer is often a regional machine, so it's a resource that either a few people have or that you operate for a region.

Mainframe costs cover a range from several million dollars down to $400,000. The mainframes' gradual price increase just reflects inflation. Prices haven't gone up, and there's no real push for them to go down. Companies aren't looking for lower prices when they buy new computers because they're already committed to a certain price range. They're looking for more performance.

The minicomputer came in at the $10,000 level in the 1970s. There were 100 mini companies in that period, of which seven survived. Then the personal computers and workstations were introduced at even lower costs, based on microprocessors. I'm predicting the multi will be a mini-class machine.

The multi provides substantially more computing power than any other machine in its price range. The architecture of the multi uses a common bus with several microprocessors, which connect with memory. I/O runs off the same bus.

It's a very simple machine, built almost like a DEC PDP-11 with Unibus. If you build a computer that way, you can put a great deal of processing power into a very small area and build it much more cheaply than mainframes or minis.

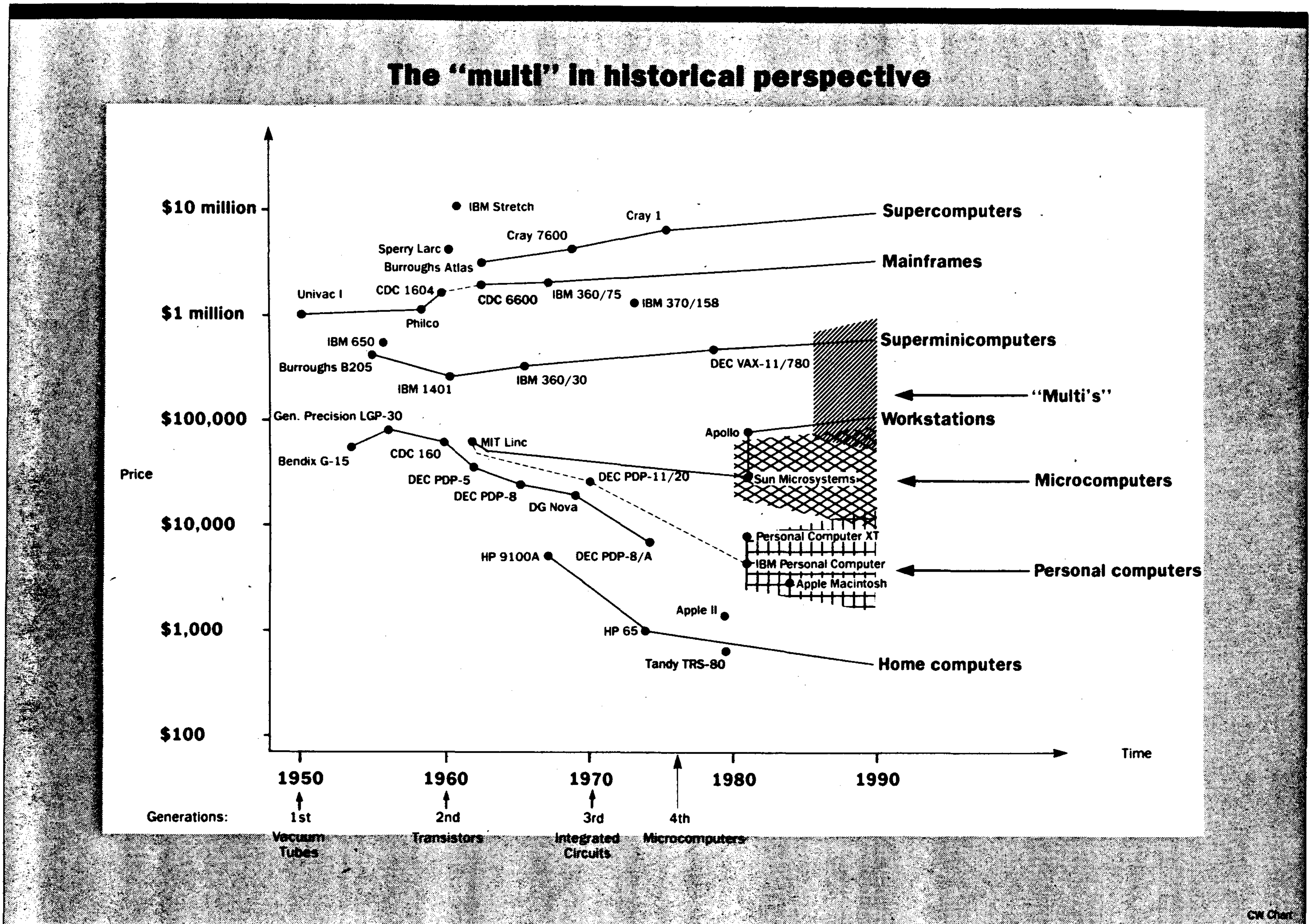
In fact, if you look at the resulting price/performance, Encore has a machine that starts at about $100,000 and goes up to about $400,000. Our first product offering is modular — you can add processors for additional speed and power. Performance goes up to 15 million instructions per second (Mips) in a straight line — sort of 1.5 Mips per card.

Compare that with the VAX family, which goes up to about 5 Mips with the 8600, at many times the cost. I believe we'll be able to deliver more power in one box. Our own bus can sustain operations of 100 Mips.

One factor in our development was that every time you want to create a new computer architecture, you need a new organization. In retrospect, that isn't why I started a new company, but it's turned out to be helpful.

At DEC, I tried to say "we're going to predicate our future on multiprocessors," but I had only a minimal effect, even though I had personally been involved in eight multiprocessor computers over the last 20 years.

The question still ahead of us at Encore is, how do you program them? How will users get the maximum power out of a structure like that?



Encore positions the new "multi" computers above workstations and personal computers.