Supercomputers: Directions in Technology and Applications

Computer Science and Technology Board, National Research Council, and Academy Industry Program, National Academy of Sciences

ISBN: 0-309-58214-8, 112 pages, 6 x 9, (1989)

Supercomputers: Directions in Technology and Applications

Academy Industry Program
National Academy of Sciences/National Academy of Engineering/
Institute of Medicine
and the
Computer Science and Technology Board
Commission on Physical Sciences, Mathematics, and Resources
National Research Council

NATIONAL ACADEMY PRESS
Washington, D.C. 1989

NOTICE: This book is based on a symposium cosponsored by the Academy Industry Program (a joint project of the National Academy of Sciences, the National Academy of Engineering, and the Institute of Medicine) and the Computer Science and Technology Board of the National Research Council. It has been reviewed according to procedures approved by a Report Review Committee consisting of members of the two Academies and the Institute of Medicine.

The National Academy of Sciences is a private, nonprofit, self-perpetuating society of distinguished scholars engaged in scientific and engineering research, dedicated to the furtherance of science and technology and to their use for the general welfare. Upon the authority of the charter granted to it by the Congress in 1863, the Academy has a mandate that requires it to advise the federal government on scientific and technical matters. Dr. Frank Press is president of the National Academy of Sciences.

The National Academy of Engineering was established in 1964, under the charter of the National Academy of Sciences, as a parallel organization of outstanding engineers. It is autonomous in its administration and in the selection of its members, sharing with the National Academy of Sciences the responsibility for advising the federal government. The National Academy of Engineering also sponsors engineering programs aimed at meeting national needs, encourages education and research, and recognizes the superior achievements of engineers. Dr. Robert M. White is president of the National Academy of Engineering.

The Institute of Medicine was established in 1970 by the National Academy of Sciences to secure the services of eminent members of appropriate professions in the examination of policy matters pertaining to the health of the public. The Institute acts under the responsibility given to the National Academy of Sciences by its congressional charter to be an adviser to the federal government and, upon its own initiative, to identify issues of medical care, research, and education. Dr. Samuel O. Thier is president of the Institute of Medicine.

The National Research Council was organized by the National Academy of Sciences in 1916 to associate the broad community of science and technology with the Academy's purposes of furthering knowledge and advising the federal government. Functioning in accordance with general policies determined by the Academy, the Council has become the principal operating agency of both the National Academy of Sciences and the National Academy of Engineering in providing services to the government, the public, and the scientific and engineering communities. The Council is administered jointly by both Academies and the Institute of Medicine. Dr. Frank Press and Dr. Robert M. White are chairman and vice chairman, respectively, of the National Research Council.

Support for this project was provided by the Academy Industry Program and by the following organizations and agencies: Apple Computer, Inc., Control Data Corporation, Cray Research, Inc., the Defense Advanced Research Projects Agency (Grant No. N00014-87-J-1110), the Department of Energy (Contract No. DE-FG05-87ER25029), Digital Equipment Corporation, Hewlett Packard, IBM Corporation, the National Aeronautics and Space Administration (Grant No. CDA-860535), the National Science Foundation (Grant No. CDA-860535), and the Office of Naval Research (Grant No. N00014-87-J-1110).

Cover: Donna J. Cox, Scientist: Charles Evans, "Neutron Star Collision," National Center for Supercomputing Applications, 1986

Library of Congress Catalog Card Number 89-62945
International Standard Book Number 0-309-04088-4
Available from:
National Academy Press
2101 Constitution Avenue, N.W.
Washington, D.C. 20418

Printed in the United States of America
S014

First Printing, December 1989
Second Printing, June 1990

ACADEMY INDUSTRY PROGRAM

ALLAN R. HOFFMAN, Director
EDWARD ABRAHAMS, Senior Staff Officer
LOIS E. PERROLLE, Staff Officer
DEBORAH FAISON, Senior Program Assistant

COMPUTER SCIENCE AND TECHNOLOGY BOARD

JOSEPH F. TRAUB, Columbia University, *Chairman*
JOHN SEELY BROWN, Xerox PARC Corporation
MICHAEL L. DERTOUZOS, Massachusetts Institute of Technology
SAMUEL H. FULLER, Digital Equipment Corporation
JAMES FREEMAN GILBERT, University of California at San Diego
WILLIAM A. GODDARD III, California Institute of Technology
JOHN E. HOPCROFT, Cornell University
ROBERT E. KAHN, Corporation for National Research Initiatives
SIDNEY KARIN, San Diego Supercomputer Center
LEONARD KLEINROCK, University of California at Los Angeles
DAVID J. KUCK, University of Illinois at Urbana-Champaign
ROBERT LANGRIDGE, University of California at San Francisco
ROBERT W. LUCKY, AT&T Bell Laboratories
RAJ REDDY, Carnegie Mellon University
MARY SHAW, Carnegie Mellon University
WILLIAM J. SPENCER, Xerox Corporation
IVAN E. SUTHERLAND, Sutherland, Sproull & Associates
VICTOR VYSSOTSKY, Digital Equipment Corporation
SHMUEL WINOGRAD, IBM Corporation
IRVING WLADAWSKY-BERGER, IBM Corporation
MARJORY S. BLUMENTHAL, Executive Director
DAMIAN M. SACCOCIO, Staff Officer
MARGARET A. KNEMEYER, Staff Associate
DONNA F. ALLEN, Administrative Secretary
CATHERINE A. SPARKS, Secretary

PREFACE

Trends in supercomputing technologies and the use of supercomputers in three innovative U.S. corporations are discussed by leading experts and by industry representatives in these proceedings of a symposium on supercomputers held at the National Academy of Sciences complex on September 8 and 9, 1988. The presentations that compose this report have been revised and updated in the interval between the symposium and publication of this report. The symposium was the product of two groups, the Academy Industry Program and the National Research Council's Computer Science and Technology Board.

The Academy Industry Program was created in 1983 to open a dialogue between the National Research Council and industry leaders. The program has a two-part purpose: (1) to make National Research Council studies, which number about 300 each year, available to industry decision makers and (2) to learn from industry how this country should best address its long-term needs in science and technology. Sixty-nine companies are currently members of this expanding program. Industry, government, and academe provide the three legs of the U.S. science and technology base, and the Academy Industry Program helps ensure that industry's role in that triad is carefully considered within the National Research Council.

The Computer Science and Technology Board, created in 1986, has an ambitious agenda—one focusing on research needs and public policies to enhance U.S. production and use of new computer technologies. The board's membership, which is half corporate and half academic, reflects its belief in a partnership of the corporate and the academic sectors. In

addition, the board is an intentional mix of people who might identify themselves as computer scientists and engineers or who would list one of the sciences as their discipline and might also call themselves computational scientists. The board is also somewhat unusual in that a very large proportion of its support comes from the corporate sector. IBM Corporation, Digital Equipment Corporation, Hewlett Packard, Cray Research, Inc., Control Data Corporation, and Apple Computer, Inc. are all corporate sponsors.

The board's most important activity by far is to study items of national interest having something to do with computing, but there is one overarching theme, the competitiveness of the United States, that lies behind almost every one of the studies.

The board believes that it is not the manifest destiny of the United States to remain the leading computer country in the world. In fact, if we act complacent, it is assured that we will not be and that we deserve not to be. The reason that leadership in computing is so important is that computing is the enabling technology. If we lose computing, we lose much more. In an information society where much of the industry is in the service sector, it is extremely easy for companies to go abroad. If we think that we saw an outflow in the manufacturing sector, we must realize how easy it would be for companies that do not have a capital base in this country to move their companies abroad.

Among the board's recent projects are the following:

- *The National Challenge in Computer Science and Technology* (National Academy Press, Washington, D.C., 1988), addresses nature and nurture issues for the computer field. Many of its major recommendations are in line with the remarks made by Senator Albert Gore, Jr., in his keynote address for this symposium.
- A report, *Toward a National Research Network* (National Academy Press, Washington, D.C., 1988), written in response to a request from the Office of Science and Technology Policy for a review of a proposed national research network, which came out of what is sometimes called the Gore initiative;
- A study, requested by the State Department, of the technology that might affect U.S. policies on export control (*Global Trends in Computer Technology and Their Impact on Export Control*, National Academy Press, Washington, D.C., 1988);
- A review, requested by the National Aeronautics and Space Administration, of NASA's computer science research program;
- A colloquium, "Keeping the U.S. Computer Industry Competitive: Defining the Agenda," held in May 1989; and

- A survey of over 100 supercomputer users and developers that was prepared by the board to help guide its future assessments of high-performance computing.

These projects are just examples from the board's rich portfolio.

Supercomputers are prominent among the board's projects because science and technology advances make high-performance computing an increasingly essential element of the U.S. scientific and industrial bases. How and why this is so are the focus of this symposium report.

JOSEPH F. TRAUB, CHAIRMAN
COMPUTER SCIENCE AND TECHNOLOGY BOARD

CONTENTS

PART A

OPENING REMARKS

1

Welcome

Robert M. White
National Academy of Engineering

Ladies and gentlemen, on behalf of the National Academy of Sciences, the National Academy of Engineering, and the Institute of Medicine, it gives me great pleasure to welcome you to our symposium on supercomputers.

This symposium was organized by both our Academy Industry Program, which seeks to strengthen the interactions between industry and the National Research Council, and our Computer Science and Technology Board, which, on behalf of both academies and the National Research Council, is responsible for the oversight of developments in computer science and technology and for providing advice on computer activities to various groups in the federal government and to others.

Certainly the digital computer and its applications have now become ubiquitous. Each generation has its own supercomputer. These, the most powerful computers produced by our industry, have characteristically opened new avenues for exploration in industry, government, research, and engineering.

In welcoming you to this symposium, I would like to comment briefly on the application of supercomputers in three branches of geophysics that are among the most active users of supercomputers and have been since the advent of large-scale digital computers. These are weather, atmospheric and ocean studies, and seismic analysis.

I was privileged to go through the period of watching weather forecasting being transformed from an art to a science, beginning in about the mid-1950s. Today it is impossible to think about a weather forecast without

thinking about the application of supercomputers to that activity. Supercomputers, for purposes of weather forecasting, are now literally scattered throughout the world.

It is also important, considering the growing national and international concerns about the greenhouse problem, to recognize that the only way we have had to simulate and experiment with the consequences of increasing concentrations of infrared gases in the atmosphere has been by modeling atmospheric and oceanic systems with supercomputers. All our information, all our forecasts, and all our predictions about possible consequences of increasing amounts of greenhouse gases stem from the application of those mathematical models and their integration on supercomputers.

We have in this audience individuals who are deeply familiar with the applications of supercomputers in a third area, seismic exploration.

These are areas—affecting industry, government, and research—that have been totally and utterly transformed by the supercomputer and the various generations of the supercomputer, and these are fields that still remain limited by the present capacity of supercomputers.

We can use almost whatever capacity can be developed and provided for us to make better forecasts, understand the climate better, and model stratigraphy in the earth—so please, keep at it. But supercomputers have transformed not only these fields but also many, many other fields, as this symposium's participants will affirm. It is the ability of large computers to simulate large and complex systems—whether they be physical, chemical, social, or economic systems—that makes them so central to social and economic progress and to progress in our understanding of nature.

A concern shared by Senator Albert Gore, Jr., and the other participants in this symposium is the challenge the U.S. computer industry faces from abroad. This is a serious and important challenge. It is a contest where we dare not come in second. We hope that this symposium will communicate at least in part what is at stake.

2

Supercomputers: Vital Tool for the Nation's Future

The Honorable Albert Gore, Jr.
U.S. Senate

Introduction of Senator Albert Gore, Jr., by Frank Press, chairman, National Research Council: I want to welcome everyone. This is an impressive turnout from both academia and literally the most important companies in America interested in the opportunities available for supercomputers. All of us are fully aware of the exploding use of these machines—the locomotives of the information age. They can slow time in showing the excited states of atoms and ions and chemical reactions or the behavior of quarks. They can quicken time by reshaping the earth's surface in minutes. They can help depict the unobservables—the propagation of cracks in a material under a load of stress, or electrons circulating around a neutron star. But that is science, and only part of the story.

There is a bewildering and growing array of applications in the development of technology and in the creation of new products and industrial processes, and that is the part of the story that this symposium addresses.

In introducing Senator Albert Gore, Jr., I say with confidence that few in Congress understand better the essential role of advancing technology in this nation's future than does Senator Gore, and he has expressed that understanding by legislative leadership, by an informed and vigorous critique of federal science and technology policy, and most especially by his efforts to ensure that this nation maintains and exploits its forefront position in supercomputers.

As a member of the Senate Commerce Committee, he has held numerous hearings and introduced important legislation related to the development and implementation of supercomputers in the United States.

Recently, Senator Gore held a hearing on supercomputer networks. Introducing that hearing, Senator Gore said, "Ensuring America's world leadership in advanced computer technology may well be the most important economic and technological challenge of the twenty-first century." Now he will tell us more about that.

Senator Gore: Thank you very much, Frank. I am delighted to be here. This is a crucial meeting, and I hope that those who might have come here with some questions in their minds about where the nation will go, and where their companies or their institutions will go, will see this conference as a beginning point for the creation of a new national consensus about where America can go to take advantage of the revolution in supercomputing, with all it means for the future of our nation.

Of course a lot of those decisions will be made on Capitol Hill, and just as the development of hardware always outpaces the development of software, so the development of software almost always outpaces the development of public policy. And true to form, we have been very slow to react.

We are going to face many great challenges in the years to come—from finding shelter for 2 million homeless men, women, and children in this country to giving the next generation of Americans the best schools on earth. But I firmly believe that there may well be no more significant economic and technological challenge than pushing forward to ensure America's leadership in advanced computer technology.

We have come to a turning point as a nation. That is said so frequently it almost has become a cliche, but it really is incredible to live in the present time. We have discovered the ability to destroy human life. We have found the blueprint of life itself. We have invented artificial intelligence. We are creating global environmental problems that sound like the plots of bad science fiction novels, and in all of these fields our so-called common sense is challenged—because common sense is an accumulation of distilled experience, and we have moved beyond historic experience.

What is happening is not only new, it is not only unprecedented, but it is also, in many cases, unimagined and is so different from what has happened in civilization up until now that we are jarred and knocked off balance and require some pause to collect ourselves and realize that this is indeed a turning point. Those of us alive today must answer a fundamental question that underlies the nuclear arms race, the greenhouse effect, the epidemics, and the starvation in the world. The underlying question is, Are we as human beings capable of rising to this unprecedented challenge?

It is a question that I believe and hope will be answered affirmatively. In a sense we are confronted in field after field not just with a crisis or

a problem, but with what Yogi Berra once described in his inimitable way when he said, "What we have here is an insurmountable opportunity."

As we look at the possible solutions for so many of the problems that we face, we can see the emerging developments in supercomputer technology as a fabulous opportunity that must not remain insurmountable.

I think that the people attending this symposium therefore have a remarkable opportunity to guide this country's future. The supercomputer is not just another useful invention. I do not know who came up with the analogy first, but I have used it many times: The supercomputer is to the Information Revolution what the steam engine was to the Industrial Revolution.

We are ahead, we like to tell ourselves. We manufacture 72 percent of the supercomputers in the world. But the benefits of supercomputing do not come from the creation of the machines; they come from the use of the machines. And we are not using the machines. The companies that could be using them to open new frontiers of science and technology and competition do not have the people who can sit down and use supercomputers. Partly as a result, the companies are not buying supercomputers. And the people who do have supercomputers are not able to communicate with each other very effectively.

So we do not imagine the new uses that are the most important ones, the ones that we do not understand yet. Make no mistake about it, this is a completely new field of scientific inquiry. Just as we have the two established methods of creating knowledge, inductive reasoning and deductive reasoning, so now we have computing, a totally new avenue to knowledge. We must learn to understand it and use it.

Over the past year, I campaigned in every region of this country and saw the foundations of solid economic progress. But when I went to Larry Smarr's National Center For Supercomputing Applications in Urbana, Illinois, and talked with others of you in this field, I really was inspired by the potential for what can take place in America.

But none of that will take place unless we solve the problems that make it an insurmountable opportunity. Frank Press referred to my hearings last month. Those hearings were part of a continuing involvement that has led me to the conviction that high-performance computing should be a top priority for the nation.

We must launch an immediate assault on five fronts:

1. We must create a national fiber optic network with high capacity for linking supercomputing centers throughout the United States.
2. We must stop chiseling at the margins where funding for supercomputing centers is concerned. We must take advantage of the investment that has already been made—which is, after all, such a minor expenditure

in the scheme of things—to allow this nation to take advantage of what is already in place.

3. We must put in place a special initiative to address the bottlenecks in computer software development, where an extreme shortage of specialized software for important applications is impeding our progress.
4. We must give the highest priority to educating and training young people, graduate students, and postgraduate students so that we will have the people who can help us participate in this revolution.
5. Finally, we must provide adequate funding for research and development in these related areas.

Six years ago, when I introduced the idea of a national fiber optic network, I spoke with people from Corning Glass Works who were, understandably, enthusiastic supporters of the concept. But I could not find widespread support for the idea.

I remember as a 10-year-old child sitting in Senate hearings where my father, who was a senator at that time, introduced legislation creating the Interstate Highway System. I remember listening as the problems were confronted and discussed, and then I remember not too many years later seeing the bulldozers move the earth and seeing the drive from our farm in Carthage, Tennessee, to Washington, D.C. cut from 17 hours to 9 hours, making it a 1-day trip instead of a 2-day trip. And I remember watching the truck traffic and commerce expand exponentially. The effects on the country have been so dramatic that they have never really been cataloged or even studied intensively. It is like the effect of the telephone system on the country—it is so pervasive it is difficult to study.

I began with that model and then changed it significantly as I began to explore high-performance computing. When we think and talk in the United States particularly in my profession of politics—about infrastructure, we often mean highways, bridges, sewer lines, and water lines, and we need those things. But we are kidding ourselves if we think that that kind of infrastructure is the key to competing with other countries in the future. Trying to compete on that basis is like rebuilding World War I military hardware in preparation for World War II.

Infrastructure is attractive to the political system because it is a function of government. It provides a role for government to play that liberals and conservatives can both accept. The benefits, generally speaking, are available to all. All boats are lifted, to use John Kennedy's analogy to the rising tide. But in thinking about infrastructure, we have to expand our imaginations and realize that the infrastructure we most need, if our objective is to increase our nation's capacity to compete and to pursue knowledge, is going to be different from the kind of infrastructure on which we have concentrated.

Can we rely on the market system to give us that kind of infrastructure? For the Interstate Highway System, we could not, yet private industry was the principal beneficiary after the government's investment was made. It was difficult to project the benefits that would generate user fees from trucks and cars paying gasoline taxes on interstate highways that did not exist, but the government made the investment and did not add a penny to the national debt, because the commerce generated by the investment was so vast as to generate user fees that have created a huge surplus in the trust fund that was established.

We must commit an act of faith once again and invest in new infrastructure that we know—*we know*—will create benefits to private industry so vast as to generate user fees or some other form of compensation to more than cover the relatively small investment that would be required to create this infrastructure.

Many people attending this symposium know a great deal about this network. For those who are hearing about it for the first time, let me briefly sketch the problem. Private industry, whether it is the communications industry or any other industry requiring fiber optic cable, does not yet need the kind of capacity that supercomputers need. The numbers sound big to me as a nonscientist, but 50 million bits per second is considered huge as a capacity, and the communications companies in most cases are not driving the networks much beyond that capacity. It does not make economic sense for them to do so.

Supercomputers can usefully take advantage of 1 billion bits per second, or 3 billion bits per second, or 10 billion bits per second. Two years ago, with help from the House Commerce Committee, I authored and steered to passage legislation that authorized and kicked off a wide-ranging study of high-performance computing networks by the Office of Science and Technology Policy. There are already more than 100 major networks in the country, including the NSFNet, but coordination among them is limited.

The study found, as I expected, that these superhighways for information are now more like left-turn lanes at rush hour. They have low capacity. They are overloaded. They are unable to keep pace with demand. This study also warned that, while the United States continues to develop the best supercomputer technologies, we have been less than successful in applying them to address our needs. Once again we are in danger of inventing a technology only to watch other nations apply the technology.

As John Young has said, Silicon Valley is not very different from Detroit if the latest trade figures for electronics are examined. The Japanese are not that far behind, because although we invented electronic devices and created them first, we have not been using them.

John Connolly at the University of Kentucky's Center for Computational Sciences said at the hearing that computer users will be able to send high-density applications such as high-quality pictures and graphics through supercomputer networks, but that demand for capacity far exceeds supply. He said that the nation may soon find itself in a "graphic jam."

A few years ago I noticed that the Japanese entity for targeting key technologies for accelerated funding and attention had produced a list of 10 or 12 top-priority projects. One of them was a 10-gigabaud—a 10-billion-bits-per-second—fiber optic network. What are *we* doing? We need to build on the advantages that we have in this field before it is too late.

It is getting to the point that it is almost too late. But it is not yet too late. That is the good news. We have the capacity to move quickly. But I want to argue to the National Academy of Sciences, to the commercial entities represented in this symposium, and to scientists and researchers from fields other than computer science that the creation of this national high-volume fiber optic network, a superhighway for information linking supercomputing centers, ought to be the *number-one science priority for the United States of America.*

It is not just a computer science project. We all know from studying the history of science about the intimate link between communications capacity and scientific advance. Why did scientific discovery explode after the invention of the printing press? Why do so many important advances occur almost simultaneously at different points around the world? It is, of course, because communication makes it possible to assemble all of the pieces of a mosaic that then becomes apparent to many people at the same time.

Now the communications technology of tomorrow, the computing capacity inherent in supercomputers, is virtually impossible for us to use because we cannot link supercomputer centers. Think for a moment with me what it would be like if clusters of researchers in universities and commercial enterprises all over America could share the capacity of those machines and the infrastructure existing at supercomputing centers, and then communicate on a regular basis with their counterparts all over the United States. The word *synergy* is inadequate to describe the advances that would occur.

Imagine for a moment what it would be like if state governments competing one with another built interchanges to connect to this network, helping to create clusters of small businesses entering the information industry and able to download their products for distribution to a hungry market connected to the network at other points throughout the United States.

There are few things we could do that would contribute more to this nation's ability to compete effectively in the future. Many questions have

to be answered before the project I propose can become a reality. The questions are being attacked right now, but the commitment must be there. I believe that those who are part of this field should turn their attention to this network as a high priority.

I believe that access to the incredible capabilities of supercomputers on a broad scale would of itself pull more bright young people into the field and into programs where they could acquire the skills that would enable them to then go into industry and teach their employers how to revolutionize the particular businesses in which they were working.

Another vital concern that we must address is the bottleneck in software development. I have introduced legislation that is focused particularly on educational software and again is designed to address a particular area that market incentives are not solving. There are specialized applications for software that have a relatively small market at this point, and the money to pay for the software is really not there, but the payoff from that software is incredibly large.

We ought to understand that, we ought to fill that gap, and we ought to create incentives where none now exist to produce that software. I have called for the establishment of a public-private corporation that will evaluate particularly high-priority projects and then provide seed money that has to be leveraged by private investment with a large multiple to the public investment. But the private investment will be pulled in by the imprimatur or seal of approval that comes from the selection of a particular software project that is greatly needed and has high priority. I think the basic idea has worked in the past, and I think that it can work in the future.

The inattention to education is an old story. I mentioned John Young earlier; he and his fabulous commission have received virtually no attention. But their report (*Picking Up the Pace: The Commercial Challenge to American Innovation*, Council on Competitiveness, Washington, D.C., 1988) is epochal, and it emphasizes education—pre-18 education, education in computer science and technology, and education in the traditional basic sciences. We must face up to the problems in the educational system.

We must also address the other problem that the Young Commission focused on. That is, how do we fill the gap between research and applications? Senator Fritz Hollings, chairman of the Commerce Committee, has come up with some imaginative proposals, as have others. I intend to spend much of the next 2 years focusing on that particular problem, and I welcome input from many here who have thought a great deal about it.

And then finally, we have to have adequate funding for the supercomputer centers and for research and development across the board.

I believe we need leadership. I believe we need vision. I believe we need commitment. Because we know what can be done. We know what the payoff can be. In a high-speed, high-stakes competition with the Japanese,

mild words of encouragement are simply not good enough. You know that, and I want you to feel one day soon that your government also knows that.

Each of you in your field is doing America a favor by pointing the way to the future. The Japanese have proved what a nation can accomplish with powerful ideas and determination, and I believe it is America's turn to do the same. After all, we were the ones who showed them how, and it is up to us to renew the American spirit. I believe we are up to the task.

3

Introduction

Larry L. Smarr
National Center for Supercomputing Applications
University of Illinois

We are gathered together at a moment in which the country is going to have to make some critical decisions about its future, and many of the people attending will help make those decisions.

I think the question we are all asking is, Why supercomputers? We certainly understand that the media like supercomputers. We read about supercomputing. We hear about it. And yet many of you here from corporations are wondering, Do I need to get involved in supercomputing? And if so, how? What are some of the reasons?

It is not enough that supercomputers represent some of the most exciting technology today.

Senator Gore in his keynote speech really put his finger on it. It is a technology the use of which by American industry may very well determine the future of the U.S. economy in the global economy. The constant references to the supercomputer as the steam engine of the information age or the machine tool of the 1990s are shorthand attempts to capture that idea. And yet probably not more than 15 percent of the Fortune 500 companies own supercomputers, so supercomputers are not seen as being fundamental to all research, development, and manufacturing today.

But if you are watching the trends, you will discover major changes happening in corporations embracing this technology. We have here today representatives from three of those trend-setting corporations—Abbott Laboratories, Eastman Kodak Company, and Apple Computer, Inc. Listen to why they are using supercomputers and listen to the struggle they are

going through internally to get their people, their scientists, and their engineers to use these machines. What I have found—and I have probably talked to several hundred industries in the last 2 years—is that in many ways the resistance is coming primarily from the scientists and engineers themselves.

As I have reflected on this phenomenon, I think that much of the difficulty goes back to the key role that universities play in their relationship to industry, as well as to the critical role that federal funding for science plays in determining the pace of scientific progress and the future of this country. Between 1970 and 1985 there was a period that I have referred to as the "supercomputer famine in American universities."

During this period federal funds were cut off for placing advanced computing equipment in our universities. For these 15 years, university students and professors were not doing their research on supercomputers. For 15 years industry hired students from universities who did not bring those skills and attitudes into industry that would create a demand for supercomputing. Now our country has placed up to very high levels in industry a whole generation of scientists, engineers, and managers who have never used, seen, or cared about a supercomputer.

From this point of view, it is not difficult to understand why there has been resistance to the use of supercomputers in industry and why America has missed its opportunity to take advantage of these machines and place them at the base of the American economy.

Fortunately this situation is changing extremely rapidly because of the foresight of the National Science Foundation (NSF) and the Congress. They undertook an initiative in 1984 to set up a national program for providing supercomputing access to American universities. This consisted of creating national supercomputer centers and beginning to build what Senator Gore referred to as the "superhighways of the information age." This proposed national network would hook together the supercomputer centers with the research universities in the country, thereby coupling the personal computers or workstations on the investigators' desks with the remote supercomputers.

That program now funds five supercomputer centers. Doyle Knight, a member of the steering committee for this meeting, is the director of the John von Neumann Center in Princeton, and I am the director of the National Center for Supercomputing Applications (NCSA) at the University of Illinois; Sid Karin, the director of the San Diego Supercomputer Center, is also participating in this symposium. The other two centers are the Pittsburgh Supercomputer Center and the Cornell Supercomputer Facility. All of the American supercomputer manufacturers (Cray Research, Inc., Control Data Corporation's ETA Systems (disbanded in April 1989), and IBM Corporation) have been represented in those centers. Three years

ago, one had to leave the United States and go to Europe to get access to American-built supercomputers to do basic research. Now we have created a national infrastructure that, between the five centers, serves roughly 6000 scientists at 200 universities in the United States. Each scientist is allocated time through a peer review of proposed research to assure the quality of the projects. That is a rather miraculous discontinuous change by American time scales. What we are going to see now is that the graduates of these 200 universities will come to industries wanting to know where their supercomputers are to do their work.

This will be similar to what we saw when engineers, who previously had used drafting tables, wanted to know where their CAD/CAM workstations were when they were hired. In a short period of time the whole notion of engineering CAD/CAM changed in America, and both productivity and the complexity of design increased.

Through the NSF supercomputing centers, the federal government is providing universities with the kind of education and training necessary to bring new blood into the national pool of individuals trained in advanced computing. Unfortunately this does not directly help the vast current research community within industry that is not going to go through that process. What we need to do is to create additional structures to expose key industry people to the same opportunities, so that they develop the enthusiasm for advanced computing that we see among the bright young graduate students and professors in American universities.

Each of the five NSF centers is pursuing industrial participation at their centers in different ways. One model will be discussed in this symposium when Cliff Perry talks of Kodak's participation in our center. Over 60 researchers from Kodak have come to the NCSA Interdisciplinary Research Center in the last 2 years to convert codes that run on ordinary computers without visualization to ones that work in a modern distributed environment of supercomputers networked to mainframes and workstations. They can work with some of the world's experts in visualization technologies to create visual interfaces to the massive numeric data fields they compute.

The most important principle that I have learned as director of a center is that if this country is really going to meet this crisis and take advantage of this opportunity, *teamwork* is going to be the key idea. This means both the structural teamwork between the federal government, industry, and universities and the kind of teamwork we see in our Interdisciplinary Research Center between individual scientists, artists, computer scientists, and computer professionals working together as small teams to take on problems of enormous complexity. Teamwork is America's strong suit. It is what will pull us through this.

A very good example of the results of teamwork can be seen in the art exhibit presented at this symposium by Donna Cox, an assistant professor

of art and design at the University of Illinois and an adjunct professor at NCSA. She is one of the most innovative computer artists today. She creates her art by taking the numeric output of supercomputers, in areas of science, engineering, and mathematics, and then working with what she terms a "Renaissance team"—an artist, a scientist, and a computer scientist—to create beautiful visualizations. Scientists are able to see their results through this teamwork in ways that they, as individual scientists, would never have known how to do, and the visual result ends up as pure art that is being shown in galleries all over the United States, and now internationally.

I think you could hear in Senator Gore's voice, as he gave his keynote address, a sense of urgency. I believe that he feels as many of us feel. I just came back from 12 days in Japan. If I felt an urgency before, I certainly feel a great deal more urgency now. This is very serious business. I think we have possibly a 1- or 2-year window as a country to take advantage of some of the lead we have in distributed computing, visualization, and our long tradition of using supercomputers in national laboratories.

But it will not happen by the normal process of diffusion on the time scale that we usually use in this country. It must be something more than that. I think that making that extra effort is what Senator Gore was challenging us to do.

The first session of this symposium, "The Changing Landscape of Supercomputer Technology," is a tutorial given by two of the leading experts on the technology of supercomputers.

First, Jack Worlton will talk about the various kinds of architectures one finds in the supercomputer arena, how we have reached the point where we are today, and something about the computer industry itself. Jack is a lifetime laboratory fellow of Los Alamos National Laboratory. He has spent 30 years in the laboratory in a number of key positions. He is now president of Worlton and Associates, and he consults and lectures worldwide. He is probably the most sought-after speaker in the world today for teaching people about the technology of supercomputers. He is also, by the way, one of the world's experts on the actual details of U.S. and Japanese competition in this area.

He will be followed by Steve Chen, who studied for his Ph.D. from the University of Illinois with David Kuck and represents one of the brilliant people who have come out of that long tradition at Illinois since World War II in architectures and software engineering for supercomputers. He went, after being at Floating Point Systems, to Cray Research, Inc. He was chief designer of the X-MP, which has been one of the best-selling supercomputers to date. He then became senior vice president at Cray Research, Inc. Recently, he moved on to become the president and chief

executive officer of Supercomputer Systems, Inc. Steve will be talking about the technologies that we should be tracking in the next 5 years and that will make supercomputers even more super in the years to come.

For the symposium's second session, "Existing Applications of Supercomputers in Industry," we have selected three different corporations that are using supercomputers in rather different fashions to maintain and increase their competitive position in the world's marketplace. Of the three, Apple Computer, Inc. is the one that actually owns its own supercomputer. Eastman Kodak Company has access to supercomputing through the NCSA and is doing interesting work onsite at Kodak with different kinds of machines, and Abbott Laboratories is in the process of deciding what to do about supercomputing.

Our first speaker in the second session is Beverly Eccles, a group leader in computational chemistry at Abbott Laboratories who is on the project team evaluating supercomputers for Abbott. She obtained her Ph.D. in theoretical chemistry from the University of California at Irvine, and she has been at Abbott for 2 years. Previously she was at the Beckman Research Institute of the City of Hope, where she did image analysis.

Cliff Perry, who represents Eastman Kodak Company in the second session, obtained his Ph.D. from Purdue University and then became a member of the faculty at the University of Minnesota. For the last 20 years he has been with Kodak in a variety of very interesting positions and was until recently the director of Kodak's Computational Science Laboratory. Kodak is probably one of the few corporations in America that has a computational science laboratory. Now he is the director of the Information and Computing Technologies Division, which oversees Kodak's Computational Science Laboratory.

Larry Tesler, whose degree is from Stanford University, will discuss supercomputer use at Apple Computer, Inc. He was a member of the legendary Xerox Palo Alto Research Center group, that troop of very exceptional people who generated many of the modern ideas about workstation-human interfaces and hardware construction. He joined Apple in 1980 and is currently vice president for advanced technologies.

PART B

THE CHANGING LANDSCAPE OF SUPERCOMPUTER TECHNOLOGY

4

Existing Conditions

Jack Worlton
Los Alamos National Laboratory
and
Worlton and Associates

Characteristics of High-Performance Computers

Types of Computers

The wide variety of computers used in computational science and engineering can be illustrated by plotting the performance and cost of all of the computers available from the computing industry. The resulting band of products, as illustrated in Figure 4.1, ranges over microcomputers, minicomputers, superminicomputers, minisupercomputers, high-end mainframes, and supercomputers.

There is a band rather than just a line because there is a range of performance that is available for a given cost and a range of costs for a given performance. In general, the lower edge of the band is represented by the older products, and the upper edge of the band is represented by the newer products. That is, newer products have a higher performance for a given cost and a lower cost for a given performance than do older products.

One of the tongue-in-cheek definitions of supercomputers is that a supercomputer is any computer that costs $10 million. In fact, current supercomputer prices range from about $1 million to $20 million, but in terms of constant dollars, the $10 million average is a useful rule of thumb over about 3 decades. For example, the IBM 704 in the mid-1950s cost $2 million to $3 million and operated at about 10,000 operations per second. However, if these 1950s dollars are converted to 1980s dollars,

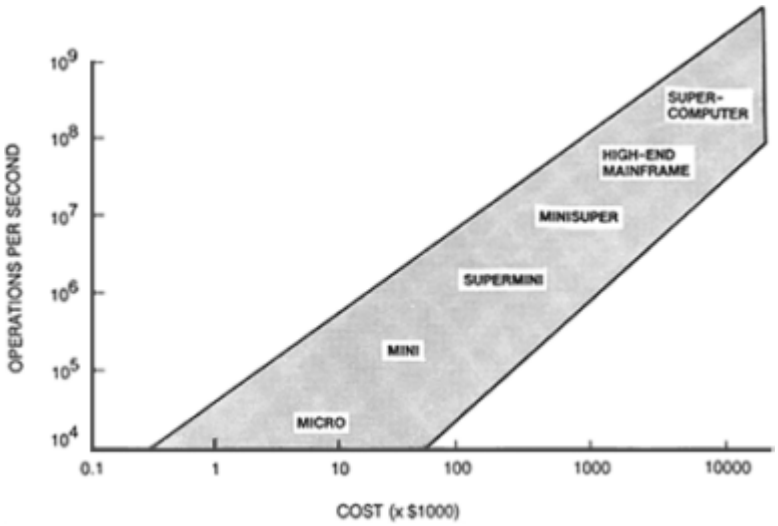


Figure 4.1
Performance and cost of available types of computers. (Reprinted, by permission, from Worlton and Associates.)

the cost of the IBM 704 is approximately $10 million. One way to view this development is that over the past 30 years, for a constant cost the performance of supercomputers has increased by a factor of 10,000; or, put another way, for a constant performance the cost has decreased by a factor of 10,000. This is equivalent to an annually compounded rate of improvement in the ratio of performance to cost of about 36 percent per year, a factor of 2 every 2.25 years, or a factor of 10 every 7.5 years.

Taxonomy of High-Performance Computers

If we examine in greater detail the high-performance computers, we distinguish at the first level three categories of computers: general purpose, special purpose, and research (Table 4.1).

General Purpose

The general-purpose high-performance computers include the supercomputers, high-end mainframes, and minisupercomputers. The supercomputers (such as the Cray Y-MP and ETA[10]) are distinguished by their relatively higher execution rates (sustained rates of about 10^8 to 10^9 operations per second), their larger memory capacities (up to several hundred

TABLE 4.1 Taxonomy of High-Performance Computers

Category	Instances
General purpose	Supercomputers High-end mainframes Minisupercomputers
Special purpose	Single-purpose Bit processors Massively parallel Systolic
Research	National Industry University

million 64-bit words), their high-speed input-output systems, and their high prices. The prices of the mainframes (such as the IBM 3090, Amdahl 5890, and Control Data Corporation 990) are about the same as those of the supercomputers, ranging from about $1 million to $20 million, but their performance is typically about a factor of 5 below that of supercomputers; also, their memory capacities are typically lower and their input-output systems are slower. The minisupercomputers (such as the Convex C-Series, the Alliant FX/8, and the SCS-40) have performances that are typically 5 to 10 times lower than those of supercomputers and prices in the range of $100,000 to $1 million, although some of the larger systems have prices in the $1 million to $2 million range.

Special Purpose

There is no clear line of distinction between general- and special-purpose computers; rather, the distinction is relative. That is, the general-purpose computers have relatively more applications that they can execute with high efficiency than do the special-purpose computers. It has been known for decades that a trade-off is available between execution rate and generality; that is, for a given component technology, special-purpose computers can be made that are faster than general-purpose computers. However, the cost of a special-purpose computer is in addition to, not instead of, the cost of a general-purpose computer, because users cannot afford to acquire a special-purpose computer for every application and therefore must always have general-purpose computers that are specialized for particular applications through application software. Thus the role of

special-purpose computers is to augment, not to replace, general-purpose computers.

The most limited purpose of design is found in single-purpose computers such as logic emulators. While limited in the range of problems they can solve, these are highly effective when the workload is highly specialized. Bit processors are designed for applications in which the data can be expressed by a single bit, such as image processing. Massively parallel computers attempt to achieve high performance by using a very large number of slow processors. Finally, the systolic computers implement an algorithm by pumping data through a series of identical functional units; for example, a systolic computer that implemented a matrix multiply would have an array of identical multiply-add processors that communicate their partial results to one another.

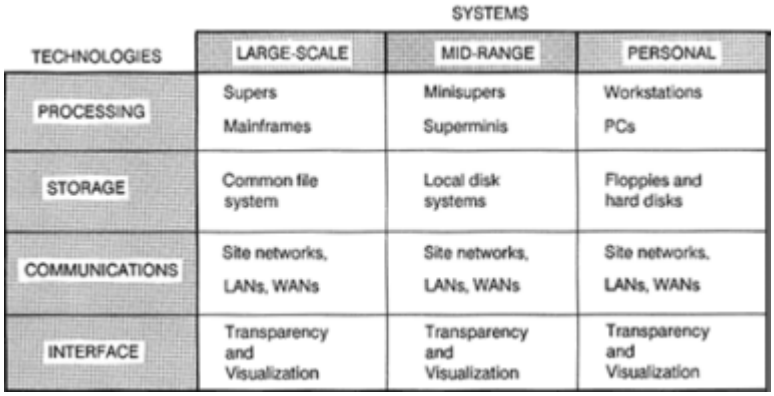
It should be observed that the generality of many of the special-purpose processors is being broadened by both hardware and software techniques in the newer designs.

Research

Research computers are designed to investigate how to make better computers rather than for applications in science and engineering. Some of these are supported at the national level, the best known being Japan's Super-Speed Computer Project, which combines the efforts of Japan's six largest computer companies to design a supercomputer that will operate at 1010 floating-point operations per second and will be ready for demonstration by March 1990. This project is funded at the $100 million level by the Ministry of International Made and Industry in Japan. There are similar projects in Europe, such as the Alvey project in Great Britain, the Marisis project in France, and the Suprenum project in West Germany. In the American scheme of things, these types of projects are conducted in universities; examples include the Cedar project at the University of Illinois, the Ultra project at New York University, and the Hypercube project at the California Institute of Technology. Finally, industrial firms are also building research computers, including the RP3, the GF11, and the TF-1 projects at IBM.

THE SCIENTIFIC COMPUTING ENVIRONMENT—A MATRIX

It is useful to think of the scientific computing environment in terms of a matrix in which we match the different types of computers against the generic information technologies, of which there are four: (1) processing, in which we transform an input into an output according to an algorithm; (2) storage, in which we transfer information over time; (3) communications, in which we transfer information over space; and (4) the human interface, in



TECHNOLOGIES	SYSTEMS: LARGE-SCALE	MID-RANGE	PERSONAL
PROCESSING	Supers Mainframes	Minisupers Superminis	Workstations PCs
STORAGE	Common file system	Local disk systems	Floppies and hard disks
COMMUNICATIONS	Site networks, LANs, WANs	Site networks, LANs, WANs	Site networks, LANs, WANs
INTERFACE	Transparency and Visualization	Transparency and Visualization	Transparency and Visualization

Figure 4.2
The scientific computing environment. Note: LAN, local-area network; WAN, wide-area network. (Reprinted, by permission, from Worlton and Associates.)

which we present information to and accept information from the user. We now classify the types of computers by the level of sharing among the users: (1) the personal resources that are not shared (personal computers with prices less than $10,000 and workstations with prices between $10,000 and $100,000); (2) mid-range computers that are typically shared by a few tens of users (minisupers and superminis with prices in the range of $100,000 to $1 million); and (3) the large-scale systems that are typically shared by a few hundred users (supercomputers and high-end mainframes with prices in excess of $1 million). By matching the generic information technologies against these categories of computers, we thereby create a 12-way taxonomy that defines the scientific computing environment, as illustrated in Figure 4.2. Some major trends and characteristics of the technologies are described below.

Trends in the Generic Information Technologies

Processing

For all three types of computers, the processing power for a given cost is increasing, and the cost of a given level of processing power is decreasing. There are no fundamental, technological, or economical limits that will prevent these trends from continuing over at least the next decade. These trends may slow down somewhat, but they will continue.

Storage

There is no single storage technology that meets all requirements for fast access time and low cost, so a hierarchy of technologies is used, including semiconductors for the main storage, magnetic disks for secondary storage, and magnetic tapes for archival storage. Progress has been most rapid in the main memory, where access times are now just a few tens of nanoseconds (a nanosecond is 10^{-9} s). The access time to magnetic disks is typically a few tens of milliseconds (a millisecond is 10^{-3} s, the time being constrained by the physical rotation of the disks. Thus there is a 6-orders-of-magnitude gap between the access times of main memory and those of disks. This causes unacceptable delays when storage requirements exceed the capacity of main memory, so that a new level in the storage hierarchy has been created, often called the solid-state disk (SSD), with access times of tens of microseconds (a microsecond is 10^{-6} s). The third level in the storage system, the archival level of storage, has been and remains a problem, because magnetic tape is not an ideal medium for very large archives: it has low volumetric efficiency, its shelf life is limited, and it can be erased. Optical storage technology is being developed in both disk and tape formats, but so far there are no recording standards for optical media, and most users are unwilling to record their archives on a nonstandard medium; thus magnetic tape will continue to be used for archival storage by most scientific computing centers until recording standards are developed for optical disks and tapes. An exception will be found in specialized applications where the advantages of optical media exceed their disadvantages.

Communications

Communication both within and among computers and their users continues to grow in importance. Three major trends are evident: (1) the rate of information transmission is increasing, (2) the cost per bit of information transmitted is decreasing, and (3) the connectivity—the number of users who have access to data communications ports—is increasing.

The point about connectivity is a matter of urgent management concern. In 1981 Los Alamos National Laboratory conducted a strategic planning exercise that led to the conclusion that data communications should be provided to all employees as a utility comparable to heat, light, power, and telephones. This policy was called the Data Communication Utility. At that time the laboratory had only about 1000 communications ports for a staff of about 7000, and new ports were being installed at the rate of only 200 per year, so that providing the full staff with data communications would have required about 30 years. However, the pace of port installation has been increased to 800 to 1000 per year and there are now between

5000 and 6000 ports for the staff, so that the Data Communication Utility will soon be a reality at Los Alamos. The Data Communication Utility is a policy that should be considered by all organizations.

The networks to which data communications ports provide access include local-area networks (LANs) that span a building, site networks that span a whole site, and wide-area networks (WANs) that span continents. These networks permit transfer of information among computing resources and among an ever-growing set of scientists and engineers who communicate with each other electronically.

Interface

The function of the human interface in information technology is twofold: (1) to provide *transparent* access to the information technologies, and (2) to provide *visualization* tools that lead to the insights that are the very purpose of computational science and engineering.

It is a fundamental principle of technology development that all technologies strive toward the condition of transparency, that is, the ability to achieve the function of the technology without specifying how the function is achieved. For example, the driver of an automobile is concerned primarily with just four devices, the steering wheel, the gear shift, the accelerator, and the brake pedal, but not with the thousands of parts these interface devices control. Similarly, the computer user should be concerned only with the nature of the problem being solved and not with specifying how each step in the solution is to be accomplished. Many computer systems are deficient not because they lack the computational power the users require, but because they are not transparent—they require the user to specify in great detail how the computation should proceed. Future systems will increasingly allow users to specify what is to be done, not how. For example, parallel-processing computers are being developed today, but these will not be mature until parallel execution is transparent to the user.

Visualization tools are now recognized as being as important as processing, storage, and communications in computational science and engineering. Richard Hamming's dictum that the purpose of computing is insight, not numbers, is the rationale for visualization. The visualization technologies include graphics terminals, microfilm, video tape, system software, and, more recently, the interdisciplinary collaboration between computer scientists and artists to present information in powerful and easily perceived forms.

Modes of Operation

Most science and engineering computer centers offer their users this whole environment, and the users then choose from among the resources

those that best meet their needs. For example, some users prefer to use a workstation for development of programs, visualization of results, and execution of small-scale problems; when problems grow beyond the ability of the workstation to provide an acceptable response time, the problems are then submitted to a supercomputer. Other users find the local control and ease of use of mid-range computers desirable as an intermediate step between workstations and supercomputers. An important characteristic of the computing environment should be that users have a uniform interface across all three types of computers, so that they can move applications among the types of computers without significant conversion effort. This is the main reason for the growing popularity of the Unix operating system: Unix is available on all three generic types of computing systems and hence can provide a relatively seamless interface among them.

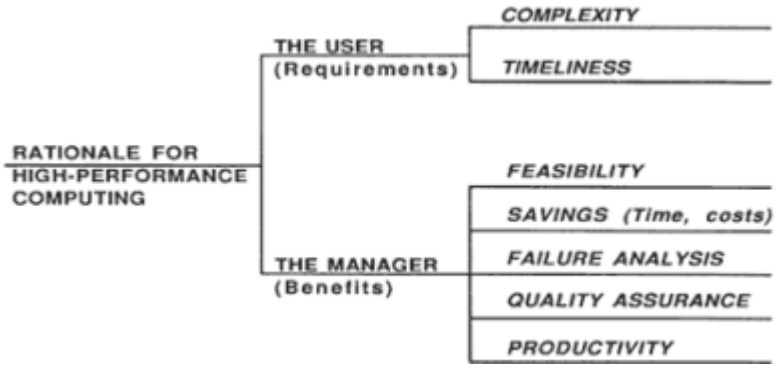


Figure 4.3
The rationale for high-performance computing. (Reprinted, by permission, from Worlton and Associates.)

RATIONALE FOR USE OF SUPERCOMPUTERS

Why do scientists and engineers use powerful but expensive supercomputers instead of less powerful but less expensive mid-range and personal computers? And why do they use supercomputers in addition to mid-range and personal computers? This discussion has to do with the rationale for supercomputing, which can be viewed from two perspectives: that of the user and that of the manager, as illustrated in Figure 4.3.

Requirements for Supercomputing: The User Perspective

From the point of view of the user, supercomputers are required to solve highly complex problems in a timely manner. If the computing environment does not have complex problems to solve, then supercomputers are not required; also, if there are complex problems to be solved but there is no time urgency for their solution, then again, supercomputers are not required. However, if both complexity and timeliness are important to an organization, then it is a management error not to provide the most powerful computing resources available, and these are by definition the supercomputers.

We can analyze computing requirements of all kinds by using a simple but powerful dimensional analysis in which we decompose execution rate into two explanatory variables, complexity and response time:

$$\text{Execution rate [operations per problem]} = \frac{\text{Complexity [operations per problem]}}{\text{Response time [seconds per problem]}}.$$

Complexity has units of operations per problem, where *operations* refers to the ordinary things computers do—add, subtract, multiply, and so on. *Response time* refers to the time required to solve a problem, in units of seconds per problem. Because this model is dimensionally correct, we can use it as an algebraic equation in which any two of the variables uniquely determine the third. For example, we can specify the complexity of the problems we want to solve and the response time required, and this determines the execution rate of the computer necessary to meet these requirements. Alternatively, for a given computer we can specify either problem complexity or response time, but not both.

Given this model we can see that there will be requirements for high execution rates if the complexity of the problem is large, or if the response time required is small, or both. It is the need to solve problems with ever-larger complexities and ever-shorter response times that is driving the unrelenting demands for higher execution rates in supercomputers. Historians tell us it is inherent in the nature of science, technology, and engineering that they grow in complexity. For example, a tour through the Smithsonian's Air and Space Museum provides a perspective not only on the history of aerospace but also on complexity. The Wright brothers' flyer was a relatively simple device compared to the DC-3, the DC-3 was not as complex as jet aircraft, and jet aircraft are not as complex as aerospace vehicles. We can quantify this growing complexity: in the decade from 1910 to 1920 it required only on the order of 10,000 engineering hours to

design large aircraft, but in the decade of the 1970s this metric had grown to about 10,000,000 engineering hours.

Similarly, shorter response times are becoming more critical because of intense international competition. In the September 1988 issue of *IEEE Spectrum*, an executive of a semiconductor firm is quoted as saying, "The limit for the time it takes to design an integrated circuit is a year. Any longer and it will already be out of date when it is introduced" (p. 33). Success in the university, in industry, and in government is often determined not only by the results produced but also by the time scale on which the results are produced.

We can quantify this relationship with a nomograph, that is, a diagram that shows a correct relationship among the variables for any straight line drawn across it, as illustrated in Figure 4.4. The scale of response times decreases as we go higher in the diagram, the scale of complexity increases as we go higher, and the sustained execution rate links the two other scales. Suppose we want to solve a problem that has a complexity of 10^9 operations, and we desire a 15-min response time, which is about 1000 s. The required execution rate is then $10^9/10^3 = 10^6$ operations per second. However, if we required the solution on an interactive time scale, for example 10 s, then the required execution rate would be 10^8, operations per second—100 times faster. Alternatively, if the 15-min response time were satisfactory but the complexity of the problem were 10^{12} operations, then the required execution rate would be 10^9 operations per second. This last example is taken from the requirements at National Aeronautics and Space Administration Ames for the Numerical Aerodynamic Simulator. Figure 4.4 shows that it is the combination of high complexity and short response times that forces the use of high-performance computers.

The largest problems that are being solved on current supercomputers have a complexity between 10^{13} and 10^{14} total operations. However, critical problems, such as long-range climate studies, that have complexities that are orders of magnitude more complex await the development of more powerful supercomputers.

The Dimensions of Arithmetic Complexity

We can analyze complexity in more detail by continuing our dimensional analysis; we decompose complexity into a 4-factor formula:

Complexity (operations per problem) = $A \cdot V \cdot T \cdot G$

where

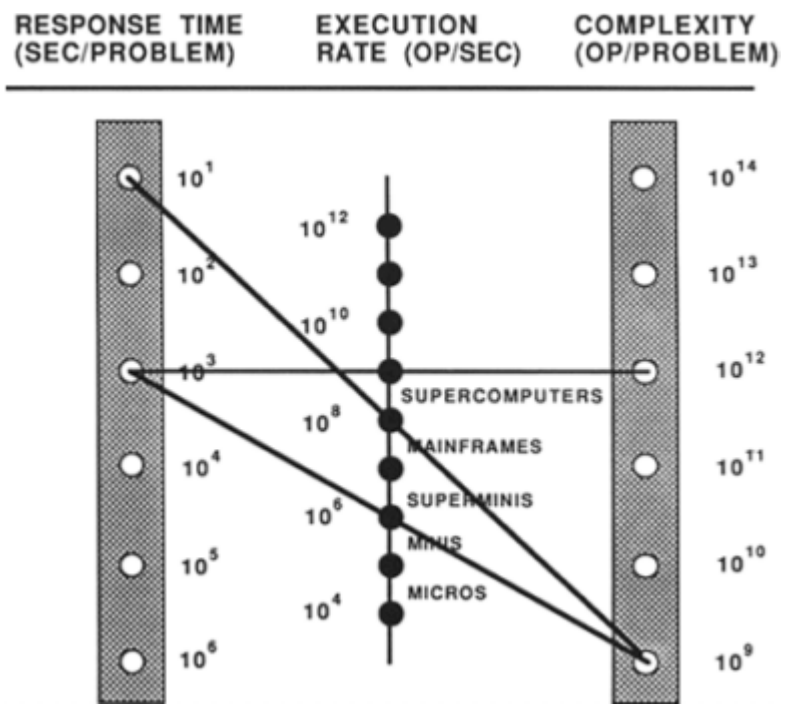


Figure 4.4
Nomograph of computational science and engineering. (Reprinted, by permission, from Worlton and Associates.)

G = geometry [points per time step]
T = time [time steps per problem]
V = variables [number of variables computed per point]
A = algorithm [operations per variable].

Geometrical complexity comes from the number of dimensions and the resolution (number of points) within each dimension, *dimension* meaning not only the space dimensions but also any variable that must be systematically varied over some range. For example, if we are tracking particles that do different things depending on their energy and angle of flight, then energy and angularity are treated as additional dimensions. If the problem is time dependent, then calculations must be performed at all of the geometrical points for each time step. For each point in space-time, a certain number of variables are computed, and for each variable, a certain number

of operations must be performed in its computation. Thus total complexity is the product of all of these 4-factors.

In practice, one of the difficult decisions in computational science and engineering is the trade-off among these variables. There is a practical limit on total complexity, so that as we increase any one of these factors we must decrease another. For example, as we increase geometrical complexity, we may have to use simpler physics; or if we put in better physics, we may have to decrease the number of dimensions or the resolution.

An Example of Complexity

The following example illustrates the structure of a typical supercomputer calculation:

Spatial resolution $G = 10^4$ mesh points (100×100)
Time resolution $T = 4000$ time steps
Variables $V = 100$ variables per point
Algorithm $A = 30$ operations per variable per time step per point

The total complexity is the product of these factors, equal to 1.2×10^{11} total operations. The problem was executed on a Cray-1 supercomputer at Los Alamos National Laboratory in 1983 at an execution rate of about 20 million operations per second, so the response time was $1.2 \times 10^{11}/2.0 \times 10^7 = 6000$ s = 1.67 h.

The Scale of Response Times

There is a broad range of requirements for response times, from only a few seconds up to about 100 h. Problems exceeding 100 h are usually deemed to be intractable, both because they delay progress on the part of the user for so long and because they consume so much of the computing resource that they deny access to other users. A problem that requires 100 h of total execution time cannot be executed nonstop but must be executed at a rate of a few hours per night, so that the real-time requirement for completion is weeks to months. The scale of response times can be thought of as follows:

- *Interactive*. Some response times must be only a few seconds, and ideally the delay should be imperceptible to the user for such simple tasks as entering a line of instruction during code development.
- *Preproduction and postproduction*. Before a long calculation is executed, it is often submitted for a short run to determine if the input is correct, to avoid wasting time. Also, after a longer production run is made, it is often necessary to analyze the results in calculations that are typically

minutes to tens of minutes long and that are run during the daytime. Small-scale production calculations are also of this size.

- *Production*. Problems that require hours of run time are run at night during the so-called production period. Thus any problem of this size can yield only one result per day to the user.
- *Benchmark*. At rare intervals, problems of higher complexity than the production runs are executed to check on the sensitivity of production runs to higher resolution or better science. The run times of benchmark calculations are typically in the range of 10 to 100 h.
- *Intractable*. Calculations requiring over 100 h are so costly in both the expenditure of resources and in delays in real time that they are rarely, if ever, performed.

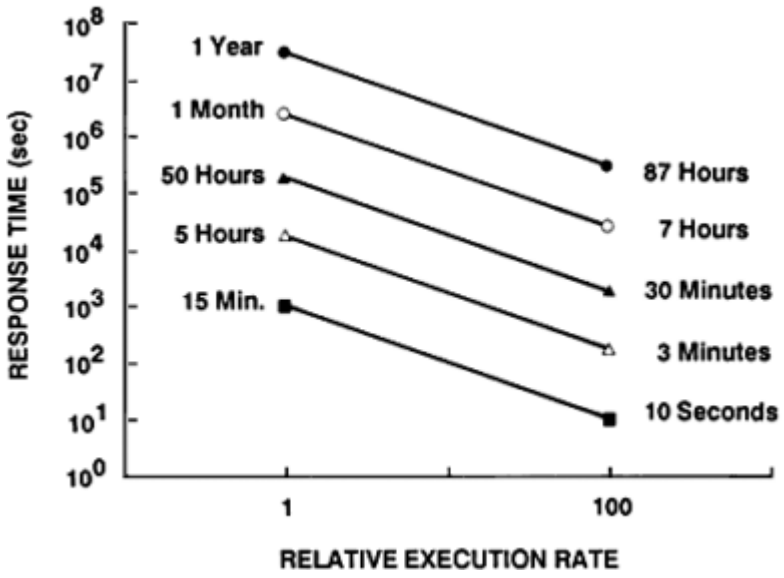


Figure 4.5
Scale of response times. Note: Solid square, interactive; open triangle, preproduction and postproduction; solid triangle, production; open circle, benchmark; and closed circle, intractable. (Reprinted, by permission, from Worlton and Associates.)

To understand the importance of more powerful computers, consider the effect of executing the above scale of problems on a computer that is 100 times faster than the original computer, as illustrated in Figure 4.5.

For ease of reference, we refer to the original computer as 1X and to the faster computer as 100X. Problems that would require a full year of calculation on a 1X, and hence would be unthinkable, could be executed

as benchmark calculations on the 100X in 87 h. Intractable problems that would require nonstop runs of 1 month (720 h) on the 1X could be done as routine production jobs on the 100X in about 7 h. The 50-h benchmark calculations on the 1X could be completed in just 30 min on the 100X. Five-hour production runs on the 1X could be done in only 3 min on the 100X, and 15-min calculations on the 1X could be completed in just 10 s on the 100X. This comparison is relevant to the relative performance of mid-range computers and supercomputers and to the relative performance of current and future supercomputers.

Benefits of Supercomputing: The Managerial Perspective

The above discussion of requirements is of interest primarily to users who have to solve complex problems in a timely manner, and this is, of course, an important part of why supercomputers are used. However, the issues of problem complexity and solution time are only of peripheral interest to managers. Managers are interested in the benefits to the organization of making the substantial investment in these computational resources. Users sometimes make the mistake of trying to explain to their managers why they need supercomputers in terms of problem complexity and response time, when what the manager wants to know is, What can we do with supercomputers that we couldn't do without them? What savings in time and money can we achieve? How will their use affect the productivity of the organization? Five benefits are briefly outlined as follows:

- *Feasibility*. The ability to overcome the limitations of experimental and theoretical methods through computational science and engineering makes it possible to solve problems that are intractable by conventional methods alone.
- *Savings*. By using computational science and engineering to guide experimentation, costly and time-consuming experiments can be focused on the most productive areas, thereby economizing on manpower, time, and budgets.
- *Failure analysis*. By identifying failure modes early in a project in the world of information rather than later in the production or operational phase, the consequences of failures can often be avoided. The space shuttle *Challenger* is a tragic example, but failures of all kinds need to be avoided to minimize delays, wasted resources, and embarrassment to an organization.
- *Quality assurance*. If the maximum set of options possible within the constraints of the time of a project are explored, optimum quality results can be produced. In the modern world of intense international competition, high quality is essential for survival.

- *Productivity*. By increasing output and reducing time and cost, the supercomputer increases the productivity of an organization.

TRENDS IN SUPERCOMPUTER TECHNOLOGY

This analysis of trends in supercomputer technology focuses primarily on system hardware, and within that topic, on trends in the technologies of processing and storage. The equally important topics of system and applications software, and the technologies of communications and the human interface within system hardware, are beyond the scope of this analysis.

Trend in Execution Rate at Los Alamos National Laboratory

High-Performance Computing

Los Alamos National Laboratory has been, and continues to be, one of the world's leading organizations in the application of high-performance computers to science and engineering, and it is instructive to analyze the history of computing at Los Alamos for insights into how high-performance computer technology has evolved. This history is illustrated in Figure 4.6.

When the laboratory was first established in 1943, there were no electronic computers, so punched-card accounting machines were used in early R&D efforts; these operated at about one operation per second. As electronic computers became available, the most powerful of these were installed at Los Alamos; Figure 4.6 illustrates the approximate sustained execution rate of the fastest of these computers (in units of operations per second normalized to the CDC 7600 for administrative purposes). The fastest computer currently installed at Los Alamos is the Cray Y-MP, which has a sustained execution rate on the order of 10^9 operations per second. Thus there has been an increase of some 8 to 9 orders of magnitude in the execution rate of computers at Los Alamos in the past 45 years. Whether future improvements in the execution rate of supercomputers will match that of the past decades is a matter of deep concern to computational scientists and engineers.

Serial versus Parallel Processors

All of the computers installed at Los Alamos through the Cray-1, the first of which was installed at Los Alamos in 1976, were serial processors—that is, they were designed to execute only a single stream of instructions in sequential mode. It is clear from the shape of the trend line in Figure 4.6 that there was a gradual slowing of the development of these serial

designs, but in 1982 a new trend line began with the installation of the first of the parallel-processor supercomputers, the Cray X-MP/2 with two vector processors, and later models in that line with four and eight vector processors. Future prospects for faster supercomputers will be based not only on improvements in component technology and the architecture of single processors, but also on the increasing number of processors used in supercomputers. Whereas the currently available supercomputers use up to 8 processors, supercomputers developed during the period 1990 to 1995 will use from 16 to 64 processors. The dramatic increase in execution rate for the projected points in Figure 4.6 is expected to come from both the increase in the number of processors and the development of higher-performance logic circuits (see below, "Trend in High-Speed Logic Technologies").

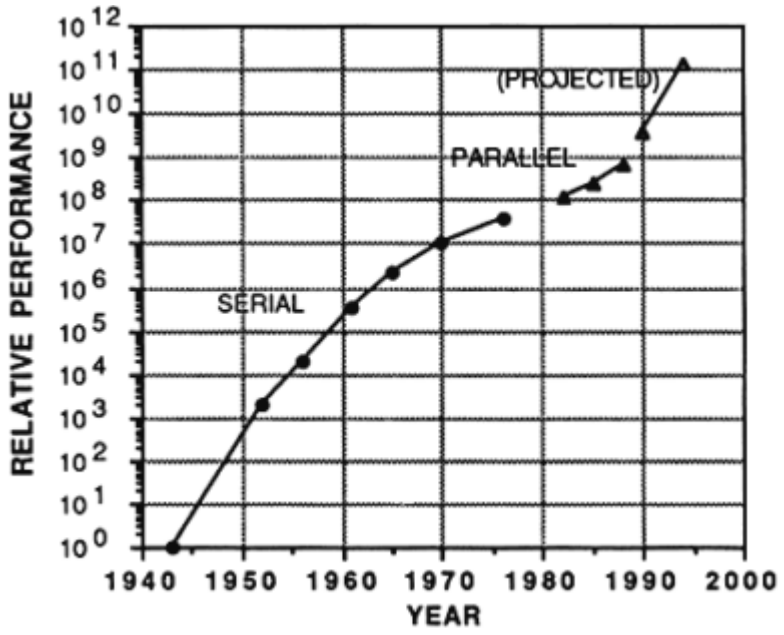


Figure 4.6
History and projection of execution rate at Los Alamos National Laboratory. Note: solid circle, serial; solid triangle, parallel; open triangle, projected. (Reprinted, by permission, from Worlton and Associates.)

Dimensional Analysis of Execution Rate

We can systematically explore the prospects for designing faster supercomputers by using a dimensional analysis of execution rate. We decompose the units of execution rate (operations per second) into three explanatory variables: (1) the cycle time, in units of seconds per cycle; (2) design efficiency, in units of results per cycle per processing element (PE); and (3) the degree of parallelism, in units of the number of Pes.

$$\text{Execution rate [results per second]} = \frac{\text{Design efficiency [results per cycle per PE]}}{\text{Cycle time [time per cycle]}} \times \text{Parallelism [number of PEs]}$$

There are three methods of increasing overall execution rate: (1) decrease the cycle time, (2) increase the single-processor efficiency, or (3) increase the number of processors. We will explore the trend for each of these variables.

Trend in Component Densities

Both speed and cost improvements in both logic and memory components depend on increases in the density of components per chip, and the general trend lines are shown in Figure 4.7.

During the period 1959 to 1972, the density of components per chip increased by about a factor of 2 every 2 years, a trend first identified by Gordon Moore of Intel and known as Moore's Law. It is useful to track these trends in terms of their quadrupling time, rather than their doubling time, because memory chip generations increase by a factor of 4. Thus, in this early period, the density of components per chip increased by a factor of 4 every 2 years. Since 1972 the density of components per chip has been quadrupling about every 3 years and is expected to slow down to a quadrupling every 4 years sometime in the early 1990s. This pattern of change is a "piecewise exponential," that is, a series of exponentials in which the successive exponents become gradually smaller, ultimately approaching a limit. This is a pattern commonly found in the evolution of electronics and high-performance computing, including semiconductor memory, magnetic disks, and the execution rates of supercomputers. Both dynamic and static RAM memories follow this pattern, with a quadrupling period of 3 years, whereas the quadrupling period of magnetic disk density is much longer, about 8 years.

Trend in Cycle Times

A scatter diagram of the cycle times of leading-edge supercomputers since the mid-1960s is shown in Figure 4.8 There has been a decrease in

supercomputer cycle time from 100 ns in the CDC 6600 in 1964, to 27.5 ns in the CDC 7600 in 1969, to 12.5 ns in the Cray-1 in 1976, to 4 ns in the Cray-2 in 1985, and there is a projected decrease to the 2- to 3- ns range in several computers scheduled for delivery in 1990. The recent trend in the leading edge of cycle-time development shows a decrease of about a factor of 2 in 4 to 5 years. This is a fairly slow rate of improvement compared to the improvements in earlier decades. For example, in the period 1955 to 1970, cycle times improved from about 12 μs to 27.5 ns, a factor of 436; however, in the next 15 years, cycle times improved from 27.5 ns to 4 ns, a factor of only about 7. This is a major reason for the slowdown in the growth of serial-processor execution rates. Those computers having cycle times that fall above the leading edge of this trend have attempted to use architectural features to compensate for their slower cycle times.

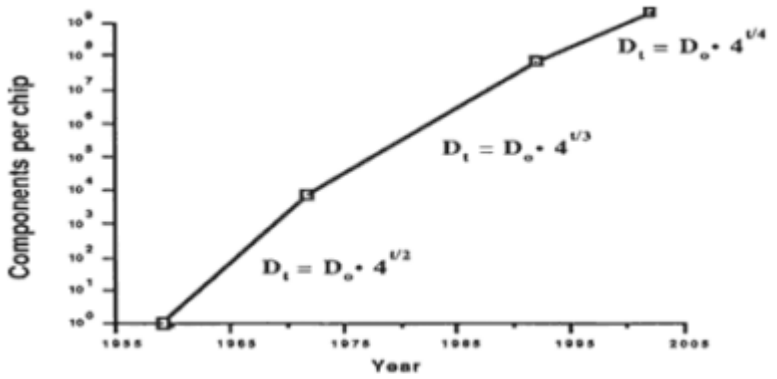


Figure 4.7
Trend in component density. (Adapted from James D. Meindl, 1987, Chips for advanced computing, Sci. Am. 256:78–88.)

Trend in High-Speed Logic Technologies

The prospects for faster logic technologies are illustrated in Figure 4.9, which shows the gate delay (in nanoseconds) and the power per gate (in milliwatts) for several high-speed logic technologies. Technologies to be preferred are those with both low gate delays and low power dissipation per gate. Not shown in Figure 4.9 are other important characteristics of logic technologies such as gate density and cost. The traditional logic technology used for high-performance computers has been bipolar emitter-coupled logic (ECL), which is the fastest of the silicon technologies and is used in most modern supercomputers. Two other logic technologies being used

in recent designs are complementary metal-oxide semiconductor (CMOS) and gallium arsenide (GaAs).

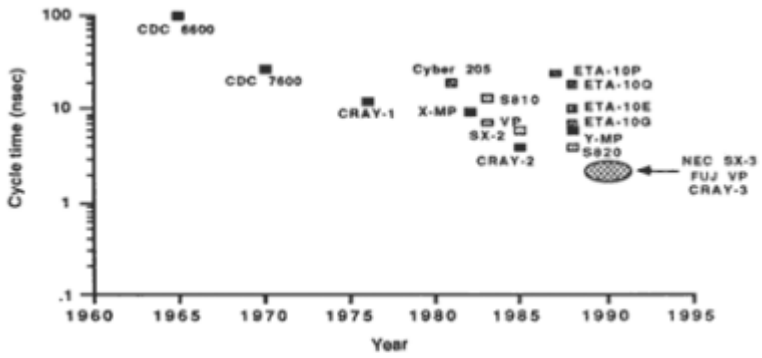


Figure 4.8
Trend in cycle times of supercomputers. (Reprinted, by permission, from Worlton and Associates.)

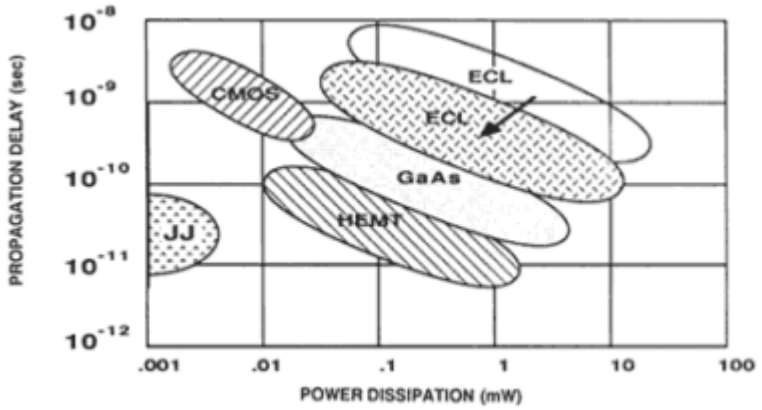


Figure 4.9
High-speed logic technologies. Note: CMOS, complementary metal-oxide semiconductor, ECL, bipolar emitter-coupled logic; GaAs, gallium arsenide; HEMT, high-electron mobility transfer, JJ, Josephson junction. (Courtesy Hiroshi Kashiwagi, Electro-Technical Laboratory, Japan.)

CMOS. The CMOS technology being used in the ETA[10] has two advantages relative to ECL: lower power dissipation and higher gate density. The lower power dissipation makes it possible to achieve high densities of gates per chip; the higher gate density makes it possible to incorporate more functions

per chip and therefore avoid transmission delays from chip to chip. CMOS has a major disadvantage as a logic technology for supercomputers: its gate delay is slower than that for ECL. However, its relatively slow speed can be partially compensated for by cooling it with liquid nitrogen to a nominal temperature of 77 K, which lowers the gate delay. The cycle times of the ETA[10] range from 24 ns down to 7 ns.

GaAs. Gallium arsenide has two advantages relative to ECL: it has a lower gate delay and lower power dissipation. The lower power dissipation implies a potential for increases in packing density without causing heat dissipation problems and therefore even further speed increases. This technology also has two disadvantages: higher cost and lower gate density per chip. The gate density of GaAs is projected to increase dramatically compared to the gate density of traditional logic technologies, as illustrated in Figure 4.10. If this projection should prove to be true or even approximately true, the gate density disadvantage of GaAs would no longer exist. The GaAs industry is maturing rapidly, and the cost per gate is projected to fall as increasing gate densities are achieved. Gallium arsenide is being used in the design of the Cray-3 for delivery in about 1990, with a projected cycle time of 2 ns.

HEMT. The high-electron mobility transistor (HEMT) technology is an aluminum-doped version of GaAs that is cooled to liquid nitrogen temperatures. It is more complex than GaAs and its development will take longer, probably into the mid-1990s. If it is developed successfully, it may offer cycle times even lower than those offered by GaAs.

Other Logic Technologies. Josephson junction technology is a superconducting technology that was investigated in the 1970s and early 1980s but then was dropped by most companies because of its complexity and also because its performance characteristics could be achieved with other technologies that were more tractable; however, Japanese companies continue to study this technology. Optical, molecular, and quantum-tunneling technologies are being studied for their potential for high-performance computers, but they seem to offer few prospects for the near term.

Summary. For the immediate future, it is expected that most supercomputers and other high-performance computers will use ECL and CMOS logic technology, but if the implementation of the Cray-3 in GaAs is successful, it could create a guidepost for others to follow. The leading-edge cycle time should be about 2 ns around 1990, and cycle times of 1 ns should appear in supercomputers by 1995.

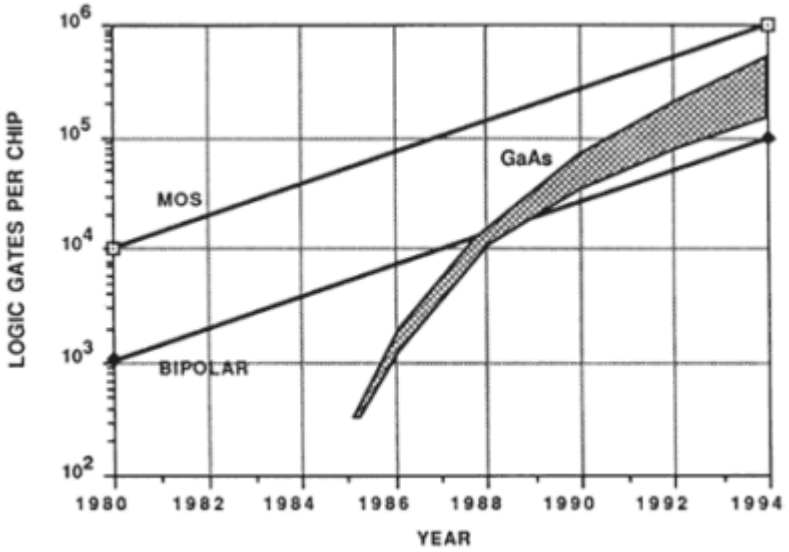


Figure 4.10
Projected trends in gate density of high-speed logic circuits. (Reprinted, by permission, from Bernard C. Cole, Gallium arsenide starts getting some respect, Electronics, June 1988, p. 41.)

Milestones in Processor Architecture

Architectural Efficiency

Efficiency in computer architecture is measured in units of results generated per clock cycle. Improvements in computer architecture that have resulted in higher efficiency are due largely to concurrency, that is, to doing more things at once. Some selected examples are illustrated in Figure 4.11.

In the mid-1950s in computers such as the IBM 704, instructions were executed in a sequential scalar mode; that is, they specified only one operation on one pair of operands, and the processing of instructions included a series of sequential steps: fetching the instruction, decoding it, forming the effective address, fetching the operand, and then executing the operation. Beginning in about 1960 in computers like the IBM STRETCH, an instruction lookahead provided the ability to fetch and process instruction N + 1 while executing instruction N. By the mid-1960s in computers such as the CDC 6600, multiple function units were included that allowed several executions, such as add and multiply, to be performed concurrently. By about 1970, the operations were speeded up by pipelining, that is, subdividing

the steps of the operations into substeps and overlapping these substeps for faster execution. The 1970s saw the development of vector processors in which the same operation was applied to many operands or operand pairs, rather than to just one pair as in the scalar designs. The major supercomputers today use vector processing to achieve high speed. The first generation of vector processors had memory-to-memory designs; that is, they fetched the vectors from memory and returned the results to memory. However, this caused long start-up times, and the second generation of vector processors followed the lead of the Cray-1 in using vector registers that allowed faster start-up times. Most vector designs today use the register-to-register design. The leading edge of supercomputer architecture today is found in designs that incorporate multiple vector processors, or parallel-vector designs.

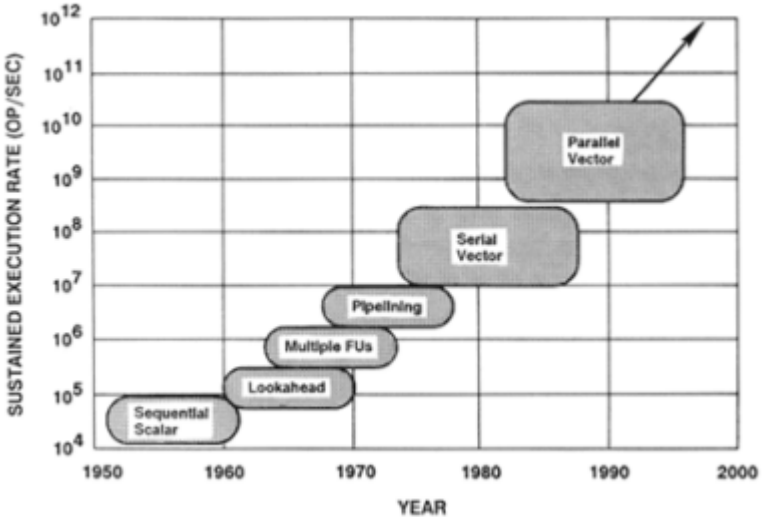


Figure 4.11
Some architectural contributions to higher performance. (Reprinted, by permission, from Worlton and Associates.)

Balance in Vector Processor Design

One of the key issues in vector processing is the balance between the scalar speed (doing one operation per instruction) and the vector speed (doing many repetitive operations per instruction). Figure 4.12 compares the merits of two designs, one with a relative performance of 10 in the scalar mode and 100 in the vector mode, and another with a relative

performance of 2.5 in the scalar mode and 400 in the vector mode. The overall performance of the system is a function of the fractions of the work done in each of the different modes. For practical applications, the fraction of vector work is in the range 0.3 to 0.7, so that even though the second design has a higher peak performance than does the first design, the first design, which is better balanced, is clearly to be preferred.

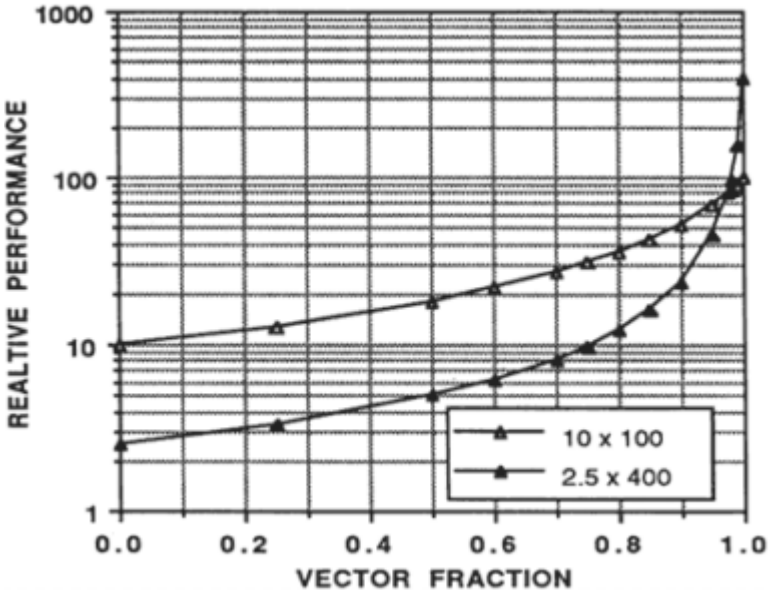


Figure 4.12
Balance in vector processor design. (Reprinted, by permission, from Worlton and Associates.)

Parallel Processing

In addition to decreasing cycle time and increasing design efficiency, a third approach to increasing computer speed is through the use of multiple processors, and there are many design issues for these so-called parallel computers: Should there be a few fast processors or many slow ones? Should the memory be shared, attached to each processor, or both? What kind of interconnect network should be used to provide communications among the processors, and among the processors and the memory? There are literally thousands of ways to design parallel computers, and at the moment there is wide disagreement in the industry about which are the best

choices. To analyze future prospects in these options, we turn to a taxonomy of computer architectures—an intellectual road map of possibilities.

Flynn's Taxonomy

Michael Flynn's taxonomy from the 1960s has been the most commonly used guide to generic types of architectures. It matches the control concurrency expressed in the number of instruction streams (one or many) against the execution concurrency expressed in the number of data streams that are being processed (one or many), to generate the four types of architectures: (1) single instruction, single data (SISD); (2) single instruction, multiple data (SIMD); (3) multiple instruction, single data (MISD); and (4) multiple instruction, multiple data (MIMD). This has been a valuable guide for over 20 years, but it is no longer an adequate guide because it is not comprehensive for all of the architectural design options being explored.

An Expanded Taxonomy of Architectures

In addition to the types of control concurrency—serial and parallel—included in the Flynn taxonomy, a third type is being used, called clustering. In a clustered design, clusters of multiple-instruction-stream processors are connected together with global control with access to global memory. Also, the types of instructions employed in the 1960s were limited to scalar instructions, and there are now several new options that need to be included in any classification scheme.

We can systematically explore the options for designing instruction types by considering a number pair (*a,b*), where *a* = the number of operations specified in the instruction and *b* = the number of operands (or operand pairs) specified. There are thus four options, as follows:

- The (1,1) option defines the scalar type of instruction, in which one operation is specified on one pair of operands.
- The (1,N) option defines the vector type of operation, in which one operation is specified, but the operation is applied to many operands or pairs of operands.
- The (M,1) option defines the systolic type of operation, in which many operations are performed on each operand that is fetched from memory.
- Finally, the (M,N) option defines the horizontal type of operation, in which many operations are specified on many operands for each instruction. The horizontal category is also referred to as very long instruction word (VLIW).

We can now create an expanded taxonomy of architectures that matches the 3 control categories against the 4 instruction categories to create a 12-way taxonomy, as illustrated in Figure 4.13.

CONTROL CONCURRENCY	INSTRUCTION TYPES			
	SCALAR (1,1)	*VECTOR* (1,N)	*SYSTOLIC* (M,1)	*VLIW* (M,N)
SERIAL				
PARALLEL				
CLUSTERED				

Figure 4.13
An expanded taxonomy of computer architectures. Notation (a,b): a, number of operations specified; b, number of operands (or operand pairs) specified. (Reprinted, by permission, from Worlton and Associates.)

Historically, through the CDC 7600 in about 1970, most supercomputers had serial-scalar designs; that is, they executed one stream of scalar instructions. First-generation vector processors had serial-vector designs; that is, they also executed a single stream of instructions, but the instructions specified vector operations. Machines of this generation were typified by the Cray-1 and the Cyber 205. In the 1980s, parallel-vector computers have been developed in which multiple vector processors are interconnected through a communication network. The Cray Y-MP and the CDC/ETA[10] are examples of computers in this category. The further development of vector designs is expected to be generalized to clustered-vector designs, with clusters of multiple vector processors interconnected with global networks. An example of this category is the Cedar project at the University of Illinois that uses the Alliant FX/8 as the cluster.

The parallel-scalar category represents those designs that use a large number of relatively slow processors to attain high performance, such as the NCUBE/ten and the Intel iPSC. These can also be designed with clustering, as found in the Myrias-2.

Systolic designs have been largely special-purpose computers that implement a particular algorithm such as a matrix multiply or a fast Fourier transform, and they can be developed in serial, parallel, or clustered form. The WARP system developed by Carnegie Mellon University is the leading edge of this type of design. The horizontal or VLIW designs attempt to execute multiple operations per clock period, and they too can be developed in serial, parallel, or clustered form. Examples of the VLIW design are found in the Cydra-5 and the Multiflow Trace computers.

This taxonomy can be extended by subdividing each of the categories to develop an even more detailed taxonomy.

Key Issues in Parallel Processing

The trend toward parallel computation is perhaps the major trend to watch in the next decade of computer architecture, and it is useful to identify some of the key issues that are being explored in the dozens of commercial and research parallel-processor projects. These issues can be thought of in three categories: system software, system hardware, and applications.

Parallel processing will affect system software in language design, compilers, operating systems, and libraries. For example, should we continue to use Fortran to specify parallel processing or should we abandon this old language and adopt something new? Fortran is being modified by adding new constructs that allow the specification of parallel processing, and no doubt this language will continue to be used long into the future because of the huge commitment to applications that exist for it. Compilers are being influenced by parallel processing because they need to identify opportunities for parallel execution in a transparent manner, that is, so the user need not be concerned with the details of how the parallelism is achieved but only with what is to be done. Operating systems are now more complex because they need to maintain control and provide services for many streams of instructions rather than just for one. Finally, system libraries are being adapted to use parallel processors.

Applications will be affected by parallel processing by the need to insert parallel control statements, to develop parallel algorithms, and to rethink the mathematical models on which the algorithms are based.

One of the key issues in the design of high-performance parallel computers is the trade-off between speed and parallelism. That is, in principle we could select the number of processors to be from as few as one or two to as many as thousands (or indeed millions), and we could select the speed of the individual processors to be as slow as a bit-serial microcomputer or as fast as a supercomputer, as illustrated in Figure 4.14.

However, a design with only a few slow processors would be too slow to be of any use as a high-performance computer, and a design with thousands of supercomputers in a single system would not be economically feasible, so that the practical area of trade-off lies in between. The evolution of parallel processing is following three guideposts.

First, the zone of "large-grain" parallelism is defined by the use of a relatively small number of fast processors, $1 < P < 100$. This form of parallel-processor design was pioneered by Cray Research, Inc. in 1982 with the introduction of the Cray X-MP product line, in which the individual

processors are vector supercomputers in their own right and the number of processors varies from the initial offering of 2 to the current offering of 8 in the Cray Y-MP/8. Other vendors have followed this design as a guidepost, including parallel-vector processors offered by Control Data Corporation, IBM Corporation, Convex, Alliant, the Japanese Super-Speed Computer Project, and widely rumored introductions of similar designs from the Japanese supercomputer vendors. The number of processors in this domain is being expanded from 8 processors to 16 in the near term, and to 32 and 64 in the generations in development for delivery in the early 1990s.

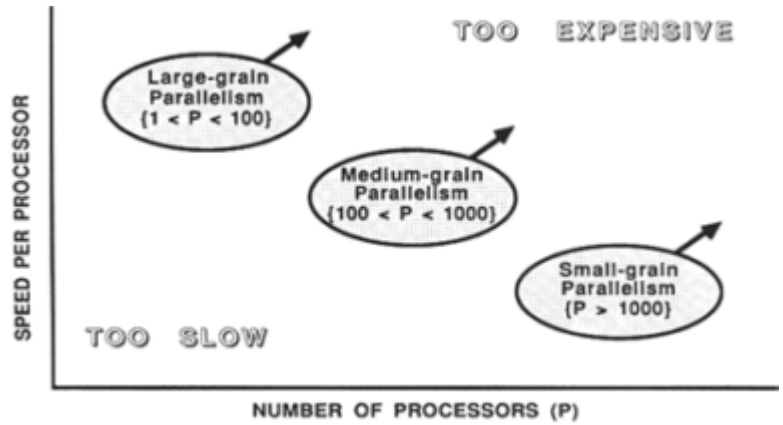


Figure 4.14
The trade-off between speed and parallelism. (Reprinted, by permission, from Worlton and Associates.)

Second, the zone of "medium-grain" parallelism is defined by the use of a larger number of slower processors, with the number of processors being in the range of $100 < P < 1000$. This form of parallelism is typified by the BBN Butterfly, which has 256 processors, and by the Myrias SPS-2, with 512 processors.

Third, the zone of "small-grain" parallelism is defined by the use of an even larger number of processors, $P > 1000$, each of which is even slower and is typically a bit-serial microprocessor. Here the number of processors varies from 4096 in the AMT-610 DAP, to 16,384 in the Goodyear MPP, to 65,536 in the Connection Machine.

A Taxonomy of Limits to Massive Parallelism

The term *massive parallelism* is loosely applied to any of a variety of computers in which the desired performance is achieved through a very

large number of relatively slow processors, with the threshold beyond 1000 processors often used to mean *massive*.

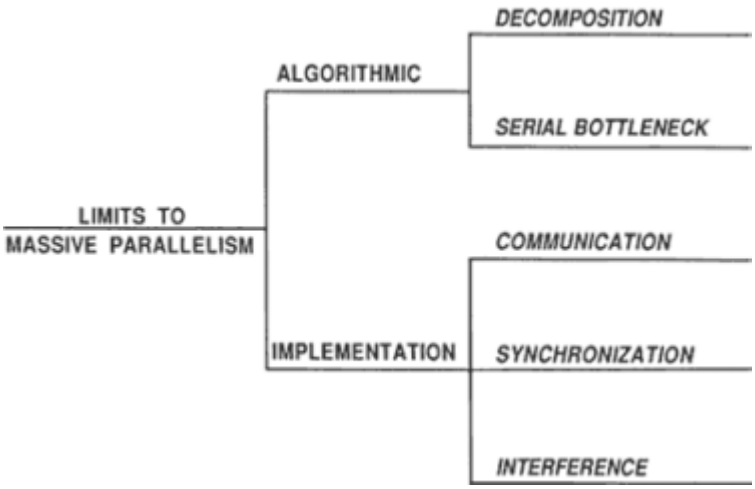


Figure 4.15
Taxonomy of limits to massive parallelism. (Reprinted, by permission, from Alex Yuen-Wai Kwok, Center for Supercomputer Research and Development Report No.679, August 1987.)

The ability of a parallel computer to perform efficiently as the number of processors is increased is referred to as scalability. A study of architectural scalability published at the Center for Supercomputer Research and Development at the University of Illinois presents a taxonomy of the limits to scalability, as summarized in Figure 4.15.

Limits of the first kind are due to algorithmic constraints. For a system with P processors to perform at full effectiveness would require that the number of available tasks equal or exceed the number of processors *throughout* the calculation. However, not all algorithms can be so decomposed, so that there is a decomposition limit to scalability. When the number of tasks available is just one, this is referred to as the serial bottleneck, which has very serious implications for the effectiveness of massive parallelism.

Limits of the second kind are due to the implementation details, including latencies caused by communications, synchronization, and interference. As tasks are decomposed to finer levels of detail, the communications requirements increase in proportion to the level of decomposition, and the amount of actual communication latency is dependent on the system design. A second implementation limit is caused by the need to integrate the results of the decomposed tasks through various synchronization techniques. At

the synchronization points, the number of available tasks decreases, so that the number of idle processors increases, thus limiting performance. A third implementation limit is caused by interference as the processors attempt to access shared resources; these resources may include shared memory, shared communication networks, or shared input-output systems.

Following Hockney's $n_{1/2}$ method, it can be shown mathematically that the fraction of parallelism that is required to achieve 50 percent efficiency for a parallel-processor system is given by

$$\pi_{1/2} = \frac{(P-2)}{(P-1)}$$

where P = the number of processors, and hence the fraction of serial work cannot exceed

$$1 - \pi_{1/2} = \frac{1}{(P-1)}.$$

Thus as P grows, the domain of applicability becomes inherently more limited, because $\pi_{1/2}$ approaches 1.0 and 1- $\pi_{1/2}$ approaches 0 very rapidly. This does not imply that it is impossible to use parallel processors effectively, but it does provide a guide to the domain of application, with a smaller number of very fast processors being more useful for general purposes than a larger number of slow processors.

CONCLUSIONS

It has been over 300 years since Galileo unlocked the door to the world of empirical science, using a telescope as a key, so to speak. In the year Galileo died (1642), Isaac Newton was born, and he invented a key we call the calculus that he used to unlock the door to the world of theoretical science. And it has been a scant 4 decades since von Neumann recognized that powerful electronic computers were the key to still another world of science, computational science.

In the opening lines of *Man of La Mancha*, the author (Dale Wasserman) invites the audience, "Come, enter into the world of my imagination." The supercomputer, too, bids us enter into the world of imagination. In the real world, we are constrained by limits of time, space, energy, matter, and costs, but in the world of the supercomputer, these constraints are merely bits of information that we can readily manipulate. In the world of the supercomputer, we can experiment with new designs for such things as aircraft, automobiles, accelerators, weapons, and chemicals; we can experiment with these designs until they either perform as we desire or we discover why they fail. By experimenting in the world of the supercomputer, we can avoid many of the costs, delays, and failures that would have

occurred if we had experimented with these designs in the real world. In this new world we can create previously unthinkable systems such as black holes, stand in space to watch them perform, and create videotapes of this adventure to take back to the real world.

The university, the industry, or the nation that would be a leader in the modern world of intense international competition must master the information technologies, the leading edge of which are the supercomputing technologies. It is imperative for our future success as a nation that we accept the invitation offered by the supercomputer—"Come, enter into the world of . . . imagination."

5

Toward the Future

Steve Chen
Supercomputer Systems, Inc.

If Jack Worlton is a lifetime fellow-user of supercomputers, I have become a long-time pursuer of a dream machine. I have chased this machine for more than 10 years. I still have not found the perfect machine to fulfill the users' needs. This has become very challenging but also very rewarding work.

My hope is that some day we can come up with a machine that is about 100 times faster than today's machines. This machine, as one of the fundamental tools, will be used by scientists and engineers in many different disciplines to study things they cannot do today.

I would like to share with you some of my thoughts on the future developments in supercomputing and their potential impact. I will speak only from a designer's point of view.

THE CURRENT STAGE IN SUPERCOMPUTING

Supercomputing has come a long way, when viewed from many angles: in speed, the central processing units (CPUs), memory size, input/output (I/O), peripherals, physical size, and software.

Speed

You have heard about machine clock rate coming down from 100 ns to 50 ns, then to 25 ns, 12.5 ns, 6 ns, and 4 ns. And each time the clock rate is reduced by half, the underlying component technology becomes more

complex. Furthermore, the requirements for data space increase. So the challenge we face in designing the machine gets worse.

Central Processing Units

The central processing unit (CPU) is the heart of the system. When we cannot get more speed out of a single CPU, we start combining more CPUs. But this is not an easy job either. We cannot just tie many boxes together and make the machine faster. My favorite analogy: to build a faster racing car, we have to decrease the car size and at the same time have more engines in the chassis. We cannot put in larger engines because the car would become big and clumsy. So for each generation, we have to invent a smaller engine that runs faster than the previous one and link together as many engines as possible, such that the car can run efficiently when all engine power is applied concurrently.

We have seen the number of CPUs increasing from 1 to 2, to 4, to 8 in a machine. But keep in mind that each CPU has to be faster than the previous generation. That makes the development work tough!

Memory Size

We start with 1 million words per CPU for data space. Next we see the words increasing to 4, then to 8, then to 16 megawords per CPU. The data space is increased to allow solving bigger problems as each generation's machines harness more and faster CPUs. We are trying to stay one step ahead of the application. Unfortunately, sometimes we have felt that we are fighting a losing game. The memory component designer can only give us a bigger memory chip with very little improvement in speed. Hence the data access time from memory becomes slower relative to data compute time. We must now figure out all kinds of tricks to compensate for the gap between the memory chip and the CPU speed.

Input/Output

Many years ago an input/output (I/O) channel could run about 1 megabyte per second. This was increased to 10 megabytes per second, and then to 100 megabytes per second, which soon will become a standard rate for anything usable. So the trend is clear. To solve bigger problems of the future, we cannot just add memory size and CPU power without significantly increasing the I/O transfer rate.

Peripherals

Peripherals are also a serious problem. Advances in storage technology are failing behind the CPU's improvement in terms of capacity and speed.

Ten years ago it was common to have disks with hundreds of megabytes and a 1-megabyte-per-second transfer rate. Today, we have gigabyte storage units with a 10-megabyte-per-second transfer rate. In the meantime, we still have to use a solid-state secondary memory device as a buffer to smooth out the speed difference between CPUs and peripherals.

Physical Size

Not too many people recognize the changes in the physical size of supercomputers. Many years ago the CDC 6600 filled about 500 square feet of floor space. The CRAY-XMP occupied roughly 100 square feet. The CPU module of the CRAY-YMP is suitcase-sized. Future products may shrink even further. But that does not mean that such a CPU is easy to design. We cannot just squeeze everything together. As each generation of machine comes down in size, the heat dissipation becomes harder to deal with. We can increase circuit density in the chip, but we cannot proportionally reduce the power per gate.

For example, a suitcase-sized supercomputer may dissipate a couple of thousand watts of power. We may be able to put it on the desktop, but we will have an instant meltdown in case of a cooling malfunction—it will go right through the table. We are dealing with a fantastic problem. It's no small design challenge to try to keep a supercomputer cool.

Software

No one paid attention to software initially. Most people were thinking about supercomputers as just pieces of hardware. The user was forced to figure out how to use it and then hand-code to optimize everything. Later on we had a little primitive compiler software. Then slowly, people started to recognize that this was not good enough anymore. Production-quality compiler software was developed for vector processing over the past 10 years. User expectations for software functionality and performance features continue to rise as more and more supercomputers become available and are widely used.

Systems

Let's view supercomputer development from a different perspective to appreciate how far we have come. When we look at the 10 year period from 1955 to 1965, we can see that the CDC 6600 was a dominant factor in the supercomputer arena, with 1 million to 10 million floating-point operations per second.

In the period 1965 to 1975, the CDC 7600, the TI ASC, the Burroughs BSP, and the Illiac IV were developed. They reached from 10 million to 100 million floating-point operations per second. The CDC 7600 was the major workhorse during this time period.

From 1975 to 1985, thanks to Seymour Cray, a new machine took the lead. Cray created the CRAY-1 architecture to take advantage of extensive pipeline vector processing. In addition, supercomputer systems became more reliable. The mean time between failures jumped from 10 hours to 100 hours and then to 1000 hours—a viable product for use in commercial industry. After the CRAY-XMP was introduced, applications expanded rapidly, from pure laboratory research to various commercial product areas.

During this time, more machines and manufacturers entered the market: the CRAY-2, the CDC Cyber-205, and also, from overseas, the Fujitsu, Hitachi, and NEC models. These machines generally reached from 100 million to 1 billion floating-point operations per second. Many more players have joined in because they see the importance of supercomputing, not only in the computer industry itself, but also in its wide effects on many key industry applications.

Personally, I have had the good fortune to work with two of the best designers in the world, Dave Kuck and Seymour Cray. I have learned a lot from them. Dave Kuck inspired me with the Illiac IV and with the follow-on Burroughs BSP project. These projects gave me a deeper inside view of the system and software areas. I was also pleased to be able to join Cray Research. Seymour Cray was a good model of the best designer in the hardware and packaging areas. Finally, I was lucky to have the opportunity to participate in designing the CRAY-XMP and the Y-MP, to try my first foot in the water.

THE NEXT STAGE IN SUPERCOMPUTING

What's in store in the next 10 years? Definitely more companies will enter the competition, but also some will fall out. The important thing is that speed will be widespread. In the highest-performance arena, instead of going 10 times faster, the range will increase to 100 times faster. We will see machines with 32 to 256 CPUs in production use. Machine speed will reach between 1 billion and 100 billion floating-point operations per second. This is based on the technology as far as we can see, barring any major breakthroughs.

Even this may not be fast enough. The Director of the National Center for Atmospheric Research, Bill Buzbee, once told me that the next generation of ocean problems may take about 100 to 1000 hours of current supercomputer time. I couldn't even comprehend the problem he was

describing. But the problem definitely cannot be solved today. We need to continue to push supercomputer technology forward in order to fulfill those requirements.

My personal goal in the future is to develop such a computational engine for scientists and engineers to open new frontiers in science and industry, similar to those made possible by the electron microscope and by steam- and gas-powered engines in earlier days.

I have discovered that developing such a machine is not an easy job anymore. No single person or single company can do it alone. We must depend on various technologies—component, software, and application—to advance in a balanced way. We need to take advantage of every technology we can get and stretch to move all these areas ahead.

Parallel Processing Environment

We are going into the arena of parallel processing, and it is just a matter of time before people will learn how to do it. I know it is painful. But we have moved from assembly language to Fortran. We took a long time to get there and now Fortran may never die. Now we must move from Fortran to parallel Fortran. It took about 10 years to grow from serial Fortran into vector Fortran, and now it may take another 10 years to go from vector to parallel Fortran. But if we don't start now, we may never be able to take advantage of the performance of future machines.

So we see where the train is going. Today, and in the near future, we will have in production from 1- to 16-processor, high-performance machines. But we also have seen experimental or developmental machines that have 32 to 256 processors or even 1000 processors. Right now such machines are in the research and development stage—the critical task is to study how to use them. Because each processor is quite slow, these machines are not used in production for general applications.

Our goal is to move gradually toward more and faster processors, while maintaining a consistent system architecture. This approach will ensure that no users will suffer a degradation of performance in running their existing production codes on the next-generation, more parallel machines when they become available. In the meantime, as users gain experience in developing more parallel application algorithms, they will be able to explore higher performance through the added number of processors. I believe this is a sensible approach to protect the users' software investment and, at the same time, induce the long-term development of parallel applications.

Next, let us focus on how the three key technology areas—component, system, and application—may proceed in developing a future high-performance supercomputer.

Component Technology Development

We will stretch the currently available component technology. We must combine improvements in many elements to enhance the design of the machine.

Device Speed

Device speeds have come down from 1 ns to 0.5 ns, and then to 250 ps and 125 ps. They may even come down to the 50-ps range. Complementary metal oxide semiconductor (CMOS), gallium arsenide (GaAs), and bipolar devices are all viable. Each has its own advantages and disadvantages.

Circuit Density

Depending on the device type, today's circuit density is approaching the 1 K-gate level for GaAs, the 10 K-gate level for bipolar, and the 100 K-gate level for CMOS. In the future, we may see even larger-scale integrated circuits. How usable are these big chips? Bigger doesn't always mean better. The advantage of these superchips depends on the trade-off of speed, power, circuit complexity, and overall system considerations.

Metal Interconnect

As circuit density increases, more transistors have to be connected in a relatively small and expensive silicon area. One way to keep the chip size down is to make the interconnect metal thinner, so that more signal lines can be placed next to each other. However, a thin metal line may degrade the signal speed and integrity. As a result, the electronic signal may travel more slowly between transistors, even though each transistor's intrinsic switching speed is very fast. And, in the worse case, the signal may not travel far enough before it disappears.

Furthermore, very thin metal may cause an electromigration problem in a high-speed (high-current) application. This is due to the loss of the electron-carrying property altogether inside the chip, leading to unreliable components. Hence we have to develop a better metal interconnect system within the integrated circuit to allow sufficient current-carrying capability (for speed), while maintaining smaller physical size (for density). The balancing act between speed and density is among the most demanding requirements facing our component designers in the future.

Substrate Material

The substrate material used to fabricate the printed circuit board is another critical factor. The traditional fiberglass-like material may not be

sufficient for future high-speed and high-density applications. The electrical property of the material may cause the signal to slow down and become noisy and lossy as speed increases. In addition, the mechanical and thermal properties of the material are also important in deciding the number of signal layers, the density of signal lines, and the compatibility between chip and substrate. We should continue to enhance current substrate materials and search for new ones to give us the maximum component packaging density required for a high-performance system.

Power Consumption

As I mentioned earlier, for a given technology, power per gate in a chip is not coming down as fast as we would like it to. We have seen improvements from 50 to 100 milliwatts per gate dropping to 10 to 20 milliwatts per gate (a factor of 5 reduction), then to 5 to 10 milliwatts per gate (only a factor of 2 reduction). This power-reduction trend appears to have flattened out. Hence, while we are increasing circuit density, the total power per chip is rising, causing difficult cooling problems at the component and system levels. This is a very critical area, and we need intensive cooperative research efforts with component manufacturers in the future.

Packaging

Many of the integrated circuits we are using are getting faster. Unfortunately, the performance gains at the component level are derated significantly because of the packaging loss all the way up to the system level. Multiple levels of interconnect media, such as printed circuit boards, chip attachments, connectors, backplane wires, and so on, all affect performance. As clock rate increases, component, module, and system packaging becomes a very critical issue for the total system design.

Testing and Measurement

The bigger the chip, the more pins there are to handle. Future chips might have 250 to 1000 pins. In addition, they will operate at high speeds and high power levels. As a result, the problem of testing chips becomes quite complex and expensive. The same is true for high-speed measurement equipment for circuit board and system checkout. Because a piece of test equipment may cost up to $5 million, the availability of cost-effective, high-performance test equipment has become a more visible concern.

Unfortunately, it is getting harder to find suppliers of advanced test and measurement equipment to satisfy the performance requirements. Companies in the United States keep dropping out of the market, and some equipment is only available from overseas. Without such equipment, one

may have the best design, but one cannot build, test, and ship the machine. So this is also a very important area to watch.

System Technology Development

Architecture Concepts

Once we have the best components, the next step is to put the system together in the slickest way. There are many ways we can do this. We hear about many different architectural concepts being explored: single versus multiple processors; system throughput versus processor speed; single-level versus multiple-level parallelism; loosely coupled versus tightly coupled system interconnects; monolithic versus distributed memory; and special-purpose versus general-purpose system design. If one looks underneath the design of future machines, it will have one or more of these architecture flavors. However, the most important thing is to design a balanced architecture and provide good software to support an application or marry applications. The user, in general, should be aware of but not be bothered with the complexity of system design.

Solution Time

As I have mentioned before, the issue now is not how fast one can design a machine to do A + B; the real issue is solution time. In earlier days, people compared different machines by counting how many millions of floating-point "add" or "multiply" operations could be done in a second (MFLOPS). That measurement is similar to the RPM (revolutions per minute) rate of the wheels of a racing car. The RPM rate is not an indicator of how much usable horsepower is available when driving on a real road. Similarly, the MFLOPS of supercomputers bear no relation to the performance obtainable on real user applications.

Later, when performance was measured by how fast a machine could compute "Livermore Loops," some people could not differentiate between a real supercomputing system and a "designer machine" targeted for Livermore Loops.

We should raise ourselves to a higher level. It took me about 5 years of preaching—I can tell you that's how long I've kept arguing the point—to convince users to find a new performance measurement yardstick. Fortunately, now they have gone up one notch to use LINPACK, a set of mathematical subroutines for solving linear algebra that is, in general, more usable than just the Livermore Loops rate or the peak MFLOPS rate.

Even so, the performance numbers on LINPACK are still only an indicator of the computation time for a small part of the total solution process. To be successful in future high-performance parallel processing

systems, we must strive for overall system performance and start to talk about solution time. And we need the users' help to define what we mean by solution time rather than computation time.

For example, three-dimensional seismic processing may involve reading more than 20,000 tapes of earth data before a machine begins to do A + B. The process starts by getting data into the machine with the 20,000 tapes and then generating the analysis and output to see exactly what is underneath the ground. The whole process may take 3 months of today's supercomputer time, during which only a few days may be spent on numeric-intensive computational tasks. We need to define this whole process so that we can measure "total time to get results." We want to make sure the scientists can do their thinking instead of playing around with the computer system, or running around the computer room.

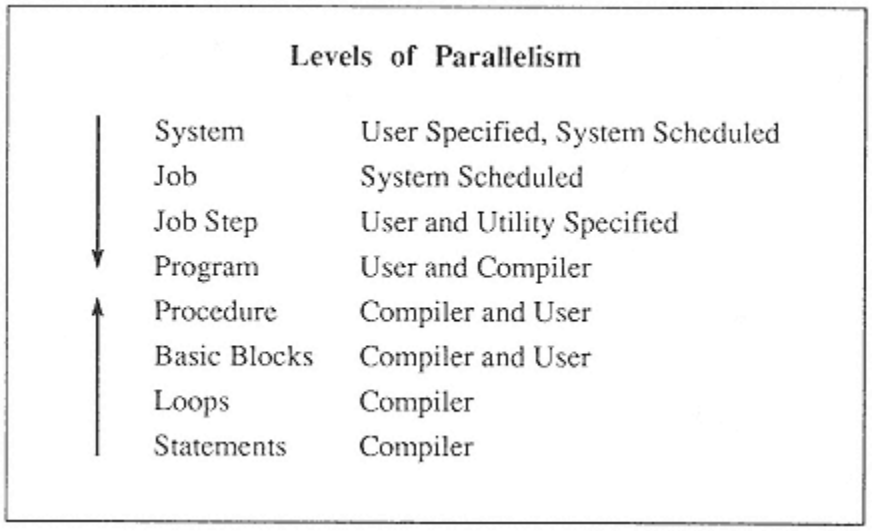
If I give a machine to an aircraft designer, that person should be able to construct a model, pick a grid point, describe the air foil, wing, and tail, and then simulate it to see if the design is correct. The model should include structure, air flow, and control and other interdisciplinary conditions that have to be satisfied in one design. The designer should be able to define this design process from beginning to end and measure the machine performance by the total time that must be spent completing this design process. This measurement is called solution time. The solution time includes all of the following elements:

- Data acquisition/entry;
- Data access/storage;
- Data motion/sharing;
- Data computation/process; and
- Data interpretation/visualization.

How to capture the raw and digitized design data, how to store it, and how to move it efficiently in and out of the disk, solid-state secondary memory, and main memory during computation are all essential to the solution process.

Then, after all that has been done, how quickly can the results be interpreted? When data can be generated very rapidly, a whole week may be required to digest the numbers. I would rather see the visual: the underground picture, or the heat flow on the surface of the integrated circuit chip. When the alpha particle hits the electronic device, I want to see the electromagnetic field moving while I watch. I want to be able to start, or stop and restart again, the simulation process any time I want. While I am simulating an air foil for an aircraft, I want to see if a particular region of the air foil is subject to high pressure or temperature. If I feel something is going wrong, I want to zoom into a particular area to test it again or try out a different algorithm or analysis. I need to have an

interactive design or analysis capability on the system. And last but not the least important of all, I want to be able to complete all this process without leaving my own design station.



I hope these examples illustrate the important difference between the computation time and the solution time that involves the whole process. Whoever designs it, the machine with minimal solution time win be the best system in real application.

Exploitation of Parallelism

To achieve high performance on future parallel systems, we should work from two directions (see box). From the bottom-up, we should continue to improve the compiler techniques to exploit automatically the parallelism in user programs. This includes extending vector detection capability to the detection of parallel processable code. From the top down, we should provide system and applications support in terms of libraries, utilities, and packages, all designed to help users prepare their applications to get the most performance out of the parallelism existing at the highest level.

One way to think of a parallel application in the future is as a multiple-domain approach. We have many, many processors at our disposal. How do we decompose a problem and make it 99 percent parallel? It is not difficult. If we look at natural phenomena, most are parallel. Unfortunately, we are trained to think sequentially. Take the aircraft design example again. We simulate one wing, then another wing, then the body, the tail. Each part is called one domain. We can now simulate all domains at the same time.

We can also think of a parallel application as a multiple-stage pipeline approach. Take the seismic processing example. First we start with tape input, and then comes data verification and alignment. The next step is analysis and simulation. The final step is data interpretation and visualization of the underground picture. All stages of the whole process can be done concurrently on the system. The first stage can be performed in groups of a few processors, with data flowing continuously to support the next stage on another group of processors, and so on.

Take this one step further. If we look at the future application development, we can bring different disciplines into one design solution, a multi-discipline approach. For example, in the design of a space shuttle, materials, structure, aerodynamics, and control problems can all be evaluated at the same time with various design criteria. The analysis step of each discipline can be processed in parallel by different groups of processors.

These examples are just a few of the ideas for exploring future parallel systems to achieve much higher system performance through a top-down application decomposition than can be obtained only by the bottom-up compiler approach. The key to success is the adaptability of the system architecture. Users should not have to change application algorithms when they migrate to higher parallel machines.

Application Technology Development

Many examples indicate that supercomputers have proved very useful in various industries—in the defense, petroleum, aerospace, automotive, meteorological, electronic, and chemical segments. Today, all the industrial countries of the world are developing their own application techniques using supercomputers. These tools improve their competitiveness in creating new materials, developing better processes and products, or making new scientific discoveries. We see existing applications expanding to include more complex geometry or more refined theory as machine capability and capacity keep improving.

We also see the potential in new areas, especially materials science. We need help to find new materials, whether we are designing integrated circuits for supercomputers or developing industrial products. Other emerging application areas include biomedical engineering, pharmaceuticals, and financial analysis. New applications will also evolve from interdisciplinary areas.

We have to think about how to develop future application technology along with future system design. We must start earlier to interact with leading application scientists and engineers to develop the next generation of algorithms to make the greatest use of parallel processing. These efforts

will also help to speed up the migration of existing application codes onto new machines.

Our challenge is to start using these machines in production as soon as they become available. The worst thing we can do in this country is to design the best machines but then not use them. Then some other country will jump in to make use of them ahead of us. We have already seen this happening in some industries with the current generation of supercomputers. We certainly want to keep our leadership position in application technology development for future machines.

SUMMARY

New Directions

In summary, I will point out a few new directions that may evolve in supercomputing:

- Comprehensive support for parallel processing;
- Development of open systems that enhance productivity and competition;
- Total system design to minimize solution time;
- Seamless services environment and distribution of functions; and
- Wider applications in scientific, engineering, and commercial fields.

In the future there will be more comprehensive support for parallel processing from very primitive to very sophisticated levels. This means that more compiler and system software features will be made available for supporting users in parallelizing their application algorithms as well as developing and debugging parallel programs.

The open system concept is spreading rapidly. Participants are working from many directions to exchange ideas and codes. An open system environment will allow us to concentrate our development and application resources only on those extension areas related to performance or functionality. This will prevent the "reinventing the wheel" syndrome and enhance our productivity in delivering competitive products.

A total system design that minimizes solution time is an important key. We will measure machines by solution time instead of by computation time.

The user will see a seamless services environment with distribution of functions—the supercomputer merged with mainframes and workstations. Users won't have to tackle different kinds of environments. Instead, an integrated design, engineering, and manufacturing computing environment will emerge, greatly enhancing user productivity and industry efficiency.

We will also see a broad expansion of applications for science, engineering, and commercial endeavors. Scientists and engineers will explore the unknown and develop new technologies. Industry will be more competitive and productive through its development of new products or processes.

Potential Impact

Developments in supercomputing technology strongly influence not only the competitiveness of key industries in our national economy but also the vitality of the computing industry itself. This influence on the computer industry can be shown in a simple triangle (Figure 5.1). The base of the triangle represents personal computers and workstations. The middle section contains mainframe or mid-range computers. At the top is the supercomputer. All three levels of technology are interacting heavily. For example, the basic component technology, parallel architecture concepts, and software and hardware design exploited in the supercomputer arena will trickle down to the mainframe and workstation level; vice-versa, the user interface software and application tools commonly seen at the workstation level will be introduced at the supercomputer level. As a result, the supercomputing technology pulls the computer industry upward, creating new market opportunities and enhancing user productivity.

Need for Technological Leadership

I used to say, "How do we stay there?" I have changed my mind. Now I say, "How do we get there?" The race is too close to call at this time.

I don't think we have too much leadership in component technology. I have worked on this problem for many years. Each year I become more humble when I see how difficult it is to build this kind of machine without a competitive and sustainable technology base.

We are losing by months from many points of view. We are starting to lose some of the critical components. We have tried to help U.S. companies, to work with them, to drive their capability forward to meet with us. But sometimes it is like wrestling with a big boat.

Our competitors have the advantage. Their work is integrated. They can focus on something and stay in there for a long time. They can sacrifice one segment of their industry to pay for another one as long as it is strategically important to their long-term technology objectives. In the past, we in the United States seemed not to be able to do that no matter how hard we tried. Thus, to reverse this trend, some component and computer industry leaders need to work together intensively to develop and maintain a strong component technology base in this country.

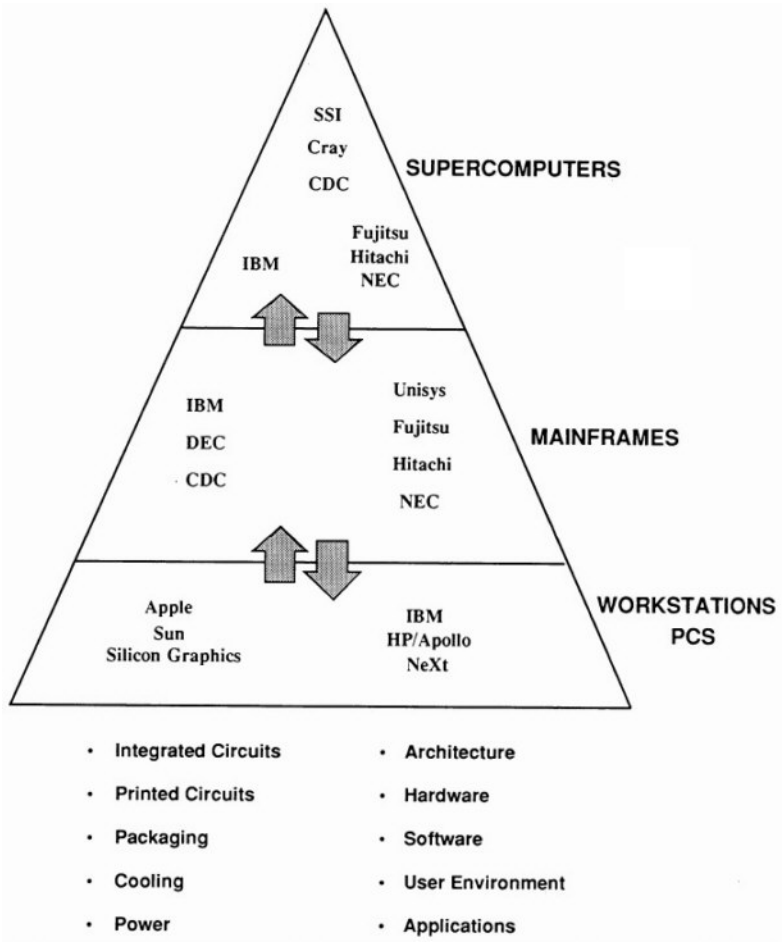


Figure 5.1
Impact of supercomputing technology. (Note: Manufacture of supercomputers by CDC was discontinued in April 1989.)

Fortunately, we still have some lead in software and application technology, especially with respect to parallel processing. My hope is to combine our resources with those of government, universities, and industry. It is important for us to keep this cooperative development effort moving. In 5 years, we can design a machine that is 100 times faster than today's, but nobody will be able to use it unless we ship it with good software and application tools.

We must start working with users today. It may take 5 years to develop an application. Beginning now, while users are developing their next-generation applications for a high-performance parallel machine, we can be developing our next-generation system software and application libraries and tools for a high-efficiency user environment. We are entering a new paradigm of supercomputing in which user application (and productivity) is in the center, instead of hardware (peak rate) as in the last decade.

That is my goal. We have to keep this technology leadership. We can accomplish it as long as we have a common view of the future. In order to develop and sustain supercomputing technology, we must take a long-term view. We must be willing to take risks. We have learned from our past experience. Also, most importantly, we should have a focus. We have many resources in this country, but they are scattered and never focused enough. That is why we are losing step by step in some areas.

These are just some of my personal observations and experiences that I would like to share with you. Certainly, I am not done yet. I am still chasing that dream machine!

DISCUSSION

Michel Gouilloud: Steve Chen, you have come with a long list of challenges and problems. Can you suggest some priorities, in other words, some of the problems you see as the most critical for you in the path of developing your next generation of machines?

Steve Chen: I think the underlying component technology is the most critical problem. For example, in silicon technology I see a plateau for speed and power. The next-generation chip we see is denser but not faster, and it requires more power. We certainly don't want to have a machine that is 100 times faster but needs 100 times more power. We may have to build a power substation next to the computer room. That problem is real. We need a breakthrough in this area.

Another critical area is high-density cooling. We have to be able to cool a small area that has very dense heat dissipation, e.g., 10,000 to 20,000 watts.

The next area is application. We need to work with users to design machines that are balanced, while at the same time preparing their future applications to take full advantage of parallel processing.

Michael Teter: We from Corning Glass are interacting fairly heavily with the Cornell Supercomputer Facility. We seem to notice that, independent of the size of the supercomputer there, as soon as users start competing for time, the amount that any individual scientist has for his own research becomes essentially negligible, and he would almost be better off buying a VAX and working by himself.

Larry Smarr: The largest university user of the NCSA has received 10,000 hours in the last year. Several users have used over 1000 hours. Kodak uses more than 100 hours a month. It is management of the allocation of time that is important. The national centers are still learning how to do this. In fact, it is only within the past several months that the blue ribbon peer review boards for each center have taken over completely the allocation of time. Previously, individual program officers at the NSF simply forwarded any good proposal they received, and that caused some real saturation problems.

Our goal is certainly to upgrade the facilities as rapidly as we can. That requires leadership and support from Congress and the NSE I believe we are all now beginning to pull together on that. Our goal is to give to those users who are on the machine both supercomputer response time and supercomputer power, even if that means that we have to limit by strict peer review the number of users on the system.

Arthur Freeman: I would like to add to the discussion about whether it

is better to use a VAX. If you can use a VAX, don't go to a supercomputer. One thing that is very clear is that 100,000 VAXs don't add up to a supercomputer in terms of capability, just as 100,000 Volkswagen engines don't add up to a Saturn engine. Supercomputers are very different from VAXs. I think people have to understand this difference between capacity and capability. Capability just is not there on a VAX. It is there on a supercomputer. We want to increase that capability all the time.

George Kuper: My question addresses a concern outside the operational discussion that has just been going on. Steve Chen, you very accurately described that one of the major challenges you face is decomposing problems and understanding how to think differently about solution sets. You said that we need to increase the demand function, because we are possibly at a stage now in our society where our supply of supercomputing capacity exceeds our ability to use it wisely.

I wonder if you think that we are facing a major intellectual challenge, a computational mechanics challenge that is even greater than the technical challenge of building faster machines?

Steve Chen: Yes, we face a psychological challenge. I was joking with Jack Worlton. For many years, every time I spoke with him, he always said he needed a machine 100 times faster. Now I say, "I will give you that machine, but tell me how to use it." Each time I have given him a machine at Los Alamos, it was already too slow. But, at the same time, the machine was not used to exploit its full performance features. We had a four-processor system for more than 5 years. But the users were still using the system as a throughput machine without going to parallel processing. This was because it was so easy to port all the existing application codes onto the new system, to run it as a four-way throughput machine instead of a four-way parallel machine.

In contrast, the overseas users are more aggressive. A good example is the European Consortium for Medium-Range Weather Forecasting. In anticipating the future performance required by finer-resolution forecasting models and upcoming parallel machines, they have already decomposed their problems with a general *n*-way parallel approach where *n* is greater than 1. They had demonstrated their parallel algorithms in a research model before the next machines arrived. Hence they were able to continue to upgrade their production forecast model from 1 processor to 2 processors, to 4 processors today; next they will have 8, 16, and even higher numbers of processors as soon as those become available. Their transition from a research to a production model has been quick and successful, because they took a long-term view and broke that psychological barrier very quickly. We in the United States are behind in this respect. We have got to catch up in this area.

John Riganati: Steve, in the earliest days of vector architecture, Seymour Cray made a presentation to Lawrence Livermore National Laboratory. At the end of the presentation he was asked what made him believe that the vector architecture he was discussing was really a general-purpose machine-it didn't exist at the time-and whether the problems at Livermore would be able to map into those.

The way Harry Nelson tells the story, Seymour just smiled enigmatically and said, "We'll see." Well, we did see, and the vector architecture has proven to be quite general purpose. But the architectures that are evolving now are one step more difficult to understand. Can you help us, especially from a user point of view, to understand why the parallel architectures, the cluster architectures, really will be general purpose in the sense that they will be able to map general applications onto those architectures?

Steve Chen: Yes. Let's refer to my earlier remarks. You can think about your applications and decompose them from the top down, e.g., using the multiple-domain approach, the multiple-stage pipeline approach, or the multi-discipline approach. These are natural approaches by which you can easily map many applications on to the parallel architecture. You get the best performance that way. With the proper tool set, the user should be able to exploit this high-level parallelism in a simple and general way without entanglement with the lowest-level machine complexity.

Mark Karplus: You gave us the hope that in 5 or 6 years you might have a machine that is 100 times faster and that will combine some improvements in technology, plus minor parallelism. What many people wonder about is doing much better. I think there will be people who will very easily figure out how to use a machine that is 100 times faster and who will want more. But there is very little discussion of massive parallelism, and many people say from the computer point of view that the future is to get machines that are 1000 or 10,000 times faster.

Steve Chen: I can only give you my personal viewpoint. I think those are worthwhile research activities at this moment. I would like to see that effort moving forward. But as far as putting 1000 microprocessors together, I don't think you can achieve the same capability we're talking about in solving general applications problems. I would rather evolve from the currently available smaller parallel machines to larger parallel machines, step by step. We have to move the whole community, instead of just one or two very bright scientists. A few people might be able to sit down at a terminal and decompose a problem into 1000 parallel tasks. That would be very good. But I don't think we can bring in the whole community that way in a short period of time. However, I do see the possibility of special-purpose massively parallel machines cooperating with the general-purpose supercomputer.

Edward Mason: At Amoco Corporation we use supercomputers and

massive computation for geophysics, but we also have a chemical company and a refining company. One of the biggest problems is retraining or educating people who are very good in particular fields of science but who have not used supercomputers, to solve problems by taking advantage of the opportunities provided by computational science and, when appropriate, by supercomputers.

Visualization and transparency are crucial. Parallel computing has been discussed a lot here, but the biologist or the chemist could care less how it is done. The concern is what can be done. And the problem is to have those experts in chemistry, biology, and other fields become familiar with how to simply, from their point of view, exploit supercomputers.

Larry Smarr: Critical to the success of that education and training, which I think is issue number one, is having the industrial users live and work in the university environment where, because of the NSF initiative, we have such a vast number of faculty and students who are not having to relearn but are very energetically going directly into using supercomputers. Having them work shoulder to shoulder with the people from industry is proving to be very effective in bringing about that technology transfer. I would very much like to see more support from the government for this education and training part of the program.

PART C

EXISTING APPLICATIONS OF SUPERCOMPUTERS IN INDUSTRY

6

Deciding to Acquire a Powerful New Research Tool—Supercomputing

Beverly Eccles
Abbott Laboratories

Abbott Laboratories is a health care company that has five major divisions. Each division behaves somewhat as a separate entity. I belong to the Pharmaceutical Products Division, and I provide support to the computational chemistry area of "discovery" research.

USING COMPUTATION TO PRODUCE PHARMACEUTICALS

In the Pharmaceutical Products Division, computation is done in two distinct areas that are functionally, physically, and managerially separate. The first is corporate computation, which supports payroll and production and sales and is primarily IBM based. The second is R&D computation, which is done primarily on Digital Equipment Corporation VAX computers.

Research and development computational support spans several areas. A network of VAX systems forms the backbone for general-purpose support functions: electronic mail, graphics, word processing, and so on. Clinical and toxicological data processing is another area of support. Government regulations require that these data be kept in archives and that statistical analyses be done on the data to prove Abbott's claims of drug effectiveness and safety. Laboratory automation systems include many special-purpose instruments that connect to computers for data collection, analysis, and possibly delivery via the computer network to a central processor or database. Computational chemistry, the area in which I am directly involved, will be the focus of the remainder of this discussion.

Computational chemistry spans three primary areas that utilize computationally intensive methods to solve problems:

1. *Computer-assisted molecular design*. This is the computer version of putting together the ball-and-stick plastic models of chemical compounds. The computer further enables the researcher to visualize a molecular structure, examine it, compute its theoretical properties, and attempt to determine what it is about the molecule that makes it particularly useful for a drug application.
2. *X-ray crystallography*. This area deals with analysis of the crystalline form of materials to determine the structure of the molecules of a chemical substance. Again, chemical structure is very important.
3. *Nuclear magnetic resonance spectroscopy*. This involves another physical measurement that helps the chemist determine the structure of a molecule, the spatial relationships between portions of molecules, and the relationships between a molecule (a drug) and a substrate.

The common theme is to try to understand the structures of molecules and how they relate to the chemical actions of these molecules. These computational activities are far from what would commonly be considered wet chemistry. Although computational chemists do perform experiments, the context of their experiments is the computational domain. Modeling is their world of chemical reality. How well a model matches observed chemical behavior is a good measure of the usefulness of the model.

The current methods used in computational chemistry give a broad picture. The goal is not to model every minute detail of a molecule and its behavior, but rather to gain an understanding of how structure and function are related. Two methods can be used to perform experiments to gain such understanding. The first involves interactive processing, which may include querying a database or looking at a graphic display of the model of a molecule—a three-dimensional depiction of the structure—on a high-resolution, high-performance computer graphics workstation and interacting with the structure in view (using techniques of rotating, zooming, color coding, and so on). Rather than physically holding a plastic model of a structure, one can turn dials and cause the structure to move on a display screen to try to answer a question such as, How do these two molecules fit together? This is the inspection and manipulation stage, the manual part of getting acquainted with these chemical structures.

The second computational method involves batch processing. A computational chemist who has an interesting structure to investigate can do any of a number of things that come under the classification of batch computations. These do not necessarily take very long to run. However, many such computations do. Some of the computations running on VAX/785-type equipment can take from a week to several months of elapsed time-shared

computing time. Such time requirements are not uncommon. Basically, our computers at Abbott are kept running night and day, every day, doing these sorts of computations.

Batch computations permit the computational chemist to evaluate the theoretical physical properties of compounds of interest and to simulate the behavior of these compounds, both individually and in interaction with other molecules. The ultimate goals of interactive and batch processing are to elucidate chemical structures, derive theoretical physical properties, relate these to observed chemical behavior, and deduce what aspects of a chemical compound result in a desired drug action. With the understanding gained from such information, it may perhaps be possible to develop a better drug, a drug that perhaps is more easily absorbed by the body, or will not break down in body tissues, or will not have serious side effects, or will be more specific in its behavior.

The computational method is a cyclical one of interactive processing interleaved with batch processing. The computational chemist views the results of the batch processing and may make adjustments in the structure, test new hypotheses, or simply resume the computation where it left off, all in the spirit of experimentation. The computational chemist works in collaboration with the more traditional bench chemist, aiding in developing rational approaches to drug design based on physical and chemical principles.

The computational chemist uses the computer daily in experimental work and conducts computational experiments on computers of varying power and function. Much of the work proceeds in a direction based on chemical intuition from accumulated years of experience. This science of computer modeling and simulation of chemical behavior is still something of an art. One cannot compute with great precision everything about a reasonably sized molecule because there are not enough computer resources in the world to do so. One has to make approximations involving adjustable parameters. One must sacrifice accuracy for feasibility of computation. In the course of experimentation, the computational chemist must manipulate many interrelated experimental parameters according to a "try-it-and-see" methodology.

ASSESSING ADVANTAGES OF SUPERCOMPUTING

Now the question is, with this cyclical methodology to performing computational experiments, What can a supercomputer do for us? The simplest advantage is that a supercomputer can speed up most calculations. Computations that take 30 days can possibly be accomplished in 1 day, or even in hours. But simply speeding up a calculation is not enough. We want to look for new ways of analyzing our data.

Immediate Feedback for Rapid Results

If we can tighten the loop between the experimental design (the interactive step) and the experimental outcome (the batch step), thus shortening the time between asking the question and seeing the answer, then we can very quickly ask the next question, and the next, thus making creative thought flow more easily. It is similar to the difference between writing a letter and making a telephone call. The feedback is immediate, and channels are followed that perhaps would not be if results came back in a week. When results come more slowly, we are more conservative in the questions we can ask, and we leave many more stones unturned. When the answers come back while the questions are still fresh, the train of thought can continue, and less time is spent trying to remember where one left off and what line of reasoning was being explored. A very important thing that a supercomputer can do is to tighten the loop between inception of an experiment and the final outcome.

Increased Human Input

A second important thing that a supercomputer can do is to put a human into the loop in some of what is now batch, iterative computation. Right now the human element in the process exists only at the initial point when the input data are assembled for a batch computation. At this point a decision must be made as to how far to carry an iterative calculation, and often this decision is based on arbitrary criteria. Then one must wait until the batch calculation is completed, only to find, perhaps, that some of the computational time was wasted on exploring an unfruitful avenue. A human can get into the loop if the time it takes to calculate each iterative step is reduced to the point that a batch calculation becomes interactive—to the point that the iteration interval is short enough that a human can monitor the progress of the iterations continuously and can intervene to take a corrective or exploratory action. There are several examples of batch activities that can profit from this. An additional benefit of continuous feedback of results is that the computational chemist can gain new chemical insight when the temporal aspect of a computation is compressed to the point that new principles can be inferred that could not otherwise have been inferred from a postprocessing review of the results of a batch computation.

Visualization of Results

A third advantage of a supercomputer is that it can enable the visualization of scientific results. Present-day computational chemists are

accustomed to viewing the ball-and-stick models, both in real plastic and in computer-produced graphic displays. This is not the only way to view a molecular structure. An atom is not a ball, nor the bond between atoms a stick. On the contrary, a molecule is more correctly viewed as a malleable cloud of electrons surrounding a number of atomic nuclei that are constrained to move in some rather preferred distances and orientations with respect to one another. From these fundamental particles, physics allows us to derive some physical properties that vary continuously throughout the volume of the molecular structure. We are looking for new ways to visualize these physical properties to gain new insights. The computational chemist who can view the data in just the right way can perhaps discern new patterns, derive new hypotheses, and explore new directions. We at Abbott are looking forward to the enhanced visualization possibilities that a supercomputer will afford us in the display of derived physical data to lead us to the development of new algorithmic approaches.

CREATING AN ENVIRONMENT FOR SUPERCOMPUTING

At Abbott Laboratories we have convinced ourselves that we do need a supercomputer for all of the reasons I have listed. However, we are also convinced that we cannot just put a supercomputer on the floor and turn the users loose on it. We need the whole integrated supercomputer environment. The environment must take into account the fact that if it is too much trouble to put a supercomputer to work for the scientists, they will not use it. They are used to sitting down to their computer terminals every day. They read their electronic mail. They work comfortably at graphics workstations. They manage input and output data sets stored in data files and databases on disk. If they are handed a supercomputer, they must be able to make use of it in the same way that they currently use existing computer resources. It must fit seamlessly into their environment. This means that we need to be able to supply interactive access to the supercomputer. It cannot just be a batch machine.

With the coming of age of Unix in the supercomputer and workstation marketplace and, more importantly, in the physical and chemical sciences, a standard interactive operating system is now a reality. But we want to do more than just interactively submit batch jobs. We also want to run those jobs interactively. We want to be able to tie the computation in progress on the supercomputer into the graphics workstation at our desk, permitting a two-way flow of data: a real-time display of intermediate results of calculations to monitor progress, and interactive input from the user to modify the course of the calculation. We want the flexibility of being able to derive new ways to visualize the data. We need the ability to program the environment to suit our evolving needs and experiments;

hence we need a development platform. The supercomputer is a new kind of tool. It is going to stimulate new thinking if it is put into the hands of people who can use it easily, and this will lead to the development of new computational algorithms.

Gaining Acceptance

At Abbott Laboratories we have had to deal with two obstacles to bringing in a supercomputer. The first, selling it to upper management, was, surprisingly, the easier to overcome. To a great degree this was due to our ability to easily demonstrate the supercomputer's usefulness in a particular pilot application. We were able to show how a supercomputer could save approximately 1 year of labor for a crystallographer in doing a refinement for determining a crystal structure. Management can see the dollars and cents of that. We were able to assert the competitive advantage to being the first pharmaceutical company to purchase a supercomputer. Also significant was the fact that we already had some champions in the ranks of upper management who had drawn their own conclusions, early on, about the scientific promise of a supercomputing environment.

The second, and greater, obstacle to acquiring a supercomputer has been gaining acceptance by the user community. We have heard of this difficulty a number of times from other organizations trying to accomplish the same thing. The problem is that the users do not see a cost-justifiable way of owning this technology and incorporating it into their research. They have short-range goals that involve a slight profit from the application of a supercomputer. Their long-range goals do not include the supercomputing environment as a necessity. A supercomputing environment is new and unfamiliar to them. We need to educate the users as to what they can do; we need to open up the creative flow. The users perceive a supercomputer not as an opportunity but as a responsibility and a burden thrust upon them. They feel that if the supercomputer is not fully utilized or that if no great breakthroughs come because of its presence, they will be responsible for a perceived failure. They feel that the expectations of management will have to be high to match the capital outlay and operating expenses of a supercomputer. They are researchers and cannot predict or engineer breakthroughs, so it is very difficult for them to stand up and say that they need and can justify the acquisition of a supercomputer. We have yet only partially sold acceptance of supercomputers to the users.

Defining Computing Requirements

At Abbott we have gone through the standard steps for acquiring any kind of computer system, starting with defining our computing require

ments. Interviews with the computational chemists and other researchers indicated the need for a machine with a performance class well into the range of supercomputers. Given this, the next step was to define other measurables and deliverables that would affect vendor selection. For Abbott, the first of these was a list of a number of turnkey applications—three or four third-party packages that we said we absolutely must have. These codes are our bread and butter in pharmaceutical computational research; they already exist commercially; we know how to put them to work in our research today. Next, we specified that we must have a strong program development platform: an interactive operating system (that works), a compiler (that works) with the capability to optimize code to suit the computer architecture, file management and data integrity, program optimization tools, subroutine libraries, and so forth. Making use of this development platform will allow Abbott to realize a competitive advantage in new computational methods. Next, we insisted on connectivity, the ability to make this machine talk to all of the computer hardware we have on our site.

Protecting Corporate Investment

Finally, we laid out the specifications for things that are not so easily quantifiable but that have played a very large role in our decision-making process for vendor selection. To try to help protect our investment and achieve our desired goals, we considered the following:

1. *Upgrade path*. What new hardware will be developed, and, more importantly, how will everything done this year move to that new piece of hardware? How long will a machine be down while it is being upgraded?
2. *Support*. Will the vendor help researchers who encounter problems? Will help be available when a machine's performance is less than optimal?
3. *Does the vendor understand the purchaser's business*? This is extremely important. A vendor that does not understand your business will not understand what you need, and the vendor's development strategies may not support your strategies. Assessing vendor understanding and support has been very important to those of us in the computational chemistry area.
4. *Do the vendor and the purchaser share a common vision*? There are so many pieces of hardware available now, including the whole hierarchy from personal computers to the supercomputer, networks, and so forth. Does the vendor share your vision of what you are trying to put in place in your company? Every company has different attitudes about departmental computing versus central computing, operating systems, graphics requirements, and so on. Making sure that the vendor and purchaser share a common vision ensures a greater ability to achieve desired goals.

MAKING A DECISION

After this whole process at Abbott of evaluating computers, evaluating vendors, and evaluating whether or not we really want to acquire our own supercomputer (or whether we want simply to get some time-sharing on another computer)—after all this, we have decided that the time is now. The simple reason is summed up in an adage that applies to everything from personal computers to supercomputers: There will always be something better next ____. We at Abbott have made that decision even though there are several players in the picture, each with distinct advantages, whether available today or promised for the future. There are the Crays that have been available for a long time and some machines that are just 80 percent developed now. In selecting a vendor it is necessary to consider the whole environment, the whole picture, and to keep in mind that as soon as a piece of hardware is bought, next month, or next year, there will be something better that can be taken advantage of when fiscal possibilities allow that in the next round.

7

Using Supercomputing to Transform Thinking About Product Design

Clifford R. Perry
Eastman Kodak Company

I am delighted to share with you why Eastman Kodak Company scientists are using the awesome power of supercomputers and associated visualization systems. I will also discuss some recent applications by Kodak scientists in multiple disciplines throughout the Kodak research laboratories. But I will begin by briefly discussing the issue of communication and assimilation of the use of supercomputers within our industrial sector, because I believe that our failure to communicate how we assimilate the use of computer technology is principally responsible for the very slow rate of application of supercomputers to industrial R&D problems.

UNDERSTANDING THE NEED FOR COMMUNICATION

Our number-one priority, in my view, is to create better ways of communicating not only how we use supercomputers but also how we encourage assimilation of their use by potential practitioners.

Although I will be sharing with you what we have been doing at Kodak, much more needs to be done. Communication and assimilation are industry-wide problems. We cannot continue to learn on our own in this highly competitive global marketplace—which brings me to a very brief, personal story.

Some 30 years ago when I was a college freshman, I accepted a job that required only that I "play" with the university's newly acquired IBM-650 computer, a so-called first-generation computer. My job was to invent applications and to prove the computer's usefulness to the university's

research community. They had been given the IBM-650 but had few ideas about what problems to apply it to. These people were scientists interested in science and not in tools that had not yet proved their direct and useful applications in helping them pursue their science.

This was an example of computer technology preceding useful application, and there was no known way to assimilate its use into its intended environment. There was no form of communication and no way to learn from others. We each had to learn on our own. Some 7 years later I accepted a job at the General Motors Technical Center's Computer Technology Division, where I was again to play the role of a researcher using computers: I was to find useful applications within General Motors' R&D community for their newly installed IBM-360 computer, a second-generation computer. But the first-generation problem still existed: the latest computer technology again had preceded development of a process to assimilate its use into its intended environment. There was no form of communication and no way to learn from others. We each still had to learn on our own.

Some 21 years after that I was asked to facilitate the use of supercomputing within Kodak's R&D community. We were in the process of signing a 3-year contract with the National Center for Supercomputing Applications (NCSA) with Larry Smarr at the University of Illinois. The readiness for supercomputing at Kodak, as I initially had surmised, was once again an example of availability preceding a process for complete implementation. There was still no known or established process to assimilate the use of supercomputing into its intended environment, and there was no form of communication and no way to learn from others.

Would we still, after 30 years, have to learn on our own? In 30 years computing technology had advanced tremendously, but there was still no organizational process or procedural framework for making its applications clear. How then was supercomputing to fulfill its promise as a problem killer and as a tool with the potential to transform thinking? There was still no approach to rapidly and effectively making its enormous capacity understandable to its potential users, and hence there was little hope of making its use pervasive within its potential market.

I believe that this past and present bumbling about is a direct result of our failure to communicate within our own organizations and with one another. Perhaps the greatest failure to communicate is between the practitioners and the laity, who fail to understand the promise, the payoff, the problems, and the limits of computational science and who, more importantly, fail to understand—because of the lack of effective ways to communicate—the synergy achieved in attacking problems with the combined methods of theory, experiment, and computation.

In this current era of intense worldwide competition, we cannot afford

to learn alone, organization by organization, one at a time. Although we may have communicated results obtained from applying supercomputer technology, we have not communicated approaches that deal with the sociology or social psychology of scientists who are the potential practitioners of supercomputing.

We have been given a remarkable tool, the supercomputer, that holds great promise for us in industry, but we cannot assume that accommodating the supercomputer requires only minor changes in the way we assimilate new technology into our R&D activities. How do we help foster the cultural change that is required?

I cannot speak of applications without also speaking of communications that are both internal and external to our organizations. We must, as supercomputer stakeholders, grow to understand the commonality of our endeavors through communication.

USING SUPERCOMPUTERS TO INCREASE PRODUCTIVITY

I think it is important to realize that supercomputers do not require visualization systems, and visualization systems do not require supercomputers. However, we have found that when they are combined into a system that includes the scientist or engineer, we have new opportunities to transform the potential of our R&D opportunities for significant discoveries and breakthroughs.

Visualization as a Stimulus for Creativity

This synergistic system, visualization and the interpretation of what we visualize, can lead to new theories and scientific paradigms, that is, the set of beliefs, values, and techniques shared by members of the scientific community and new tools for the advanced engineering sciences.

I think that the power of computer-generated scientific visualization is best summarized by Herbert Butterfield as quoted in Thomas S. Kuhn's *The Structure of Scientific Revolutions* (University of Chicago Press, 1970). Butterfield described science's reorientation by a change in paradigm as "picking up the other end of the stick, a process that involves handling the same bundle of data as before but placing it in a new system of relations with one another by giving it a different framework."

Supercomputer visualization is that system. It fosters different interpretations that can lead not only to insight and understanding but also to a shift in paradigms. It has been said that when Aristotle and Galileo looked at swinging stones, Aristotle observed a falling object that was constrained,

whereas Galileo saw a pendulum. Priestley and Lavoisier both saw oxygen, but they interpreted their observations differently.

The richness of visualization as part of the process of scientific inquiry is that it allows us to differ in our interpretations of what we have seen. These different interpretations can lead us to new theories and new scientific paradigms, and then to new technologies and new products.

Supercomputing and visualization systems at Kodak have not caused shifts in paradigm as significant as those caused by the Newtonian or Einsteinian revolutions, nor have they caused relatively small changes in the paradigms generated by the wave theory of light, the dynamical theory of heat, or Maxwell's electromagnetic theory. Supercomputing has, however, allowed us to transform the way we think about problems involving polymers, crystalline compound dies, photographic imaging systems, and manufacturing technology development.

For example, using supercomputer simulation, our engineers at the Kodak Park Division were able to test and then to redesign the delivery system used in a critical photographic film manufacturing process. This eliminated the need to build a prototype of the first flawed design and resulted in cost savings of hundreds of thousands of dollars. These new theories and processes will accelerate Kodak's ability to continually enhance the quality of its products.

I will now discuss a few of the many examples that illustrate how the supercomputer has allowed us to change the way we think about specific problems and has led to new solutions that would not have been possible without supercomputing and visualization systems. I hope that these examples will serve as evidence that we are making progress through the effective use of supercomputing to enhance the quality of our products and our manufacturing processes.

Visualization and Simulation of Physical Processes

The following three examples of supercomputer-assisted science involve simulations of physical processes. The graphics allow us to visualize the gigabytes of scientific data generated by the supercomputer simulations. These computer-generated images illustrate physical qualities as well as quantities.

The first example illustrates how we use supercomputers in fundamental research. At Kodak, we synthesize polymers for many purposes. Thus we need to better understand the properties of these complex molecules in order to design new and better plastic materials. Supercomputers and visualization systems allow our researchers to study the physics and chemistry of basic polymers in new and different ways. For example, these researchers

are investigating how a polymeric network behaves when stressed, the mechanisms of self-diffusion, and the effects of polymer blending. Results from these studies will enable prediction of polymeric behavior in the design of materials with specific performance criteria. In a real sense, these polymer researchers are designers of new materials.

The goal of another investigation is to understand the diffusion of a polymer chain when it is entangled with other chains. The interactions and intraactions of the chains significantly influence the ability of a polymer chain to diffuse. It is possible to see on videotape one polymer entangled in a network of other polymers. We can then follow the path the polymer takes while wandering about the material. The sooner the polymer is able to diffuse away from its initial configuration, the sooner it is able to relieve the stress. This important information will enable scientists at Kodak to infer the behavior of the material when it is under stress as well as its viscosity and other viscoelastic properties.

Developing this knowledge is a challenge. Although the questions appear to be simple, obtaining answers requires well-crafted computer simulations involving many hours of computer time. Visualization is again required to gain insights from the gigabytes of data generated by the simulation, and it is often very helpful in determining the validity of the models.

Visualization Applied to the Manufacture of Products

A second example illustrates the use of supercomputers at Kodak in applied research at the engineering level. It is no surprise that Kodak has a very keen interest in the manufacturability of plastics. Plastics can be found in many of our products, from cameras and film spools to copiers and mass memory products. We also produce bulk plastics such as acetate fibers and PET products that are used extensively by the bottling industry. We are always looking for new plastics to enhance product quality and to reduce costs—both the costs of materials and the costs of manufacturing.

A simple example illustrates how supercomputing and visualization can help. Suppose we want to make an improved plastic part for one of our products, such as a camera body. We also desire increased productivity in the process used to manufacture the parts of the camera body. We assume that we will use a totally new plastic, but first we must familiarize ourselves with its characteristics. What happens when the plastic flows into the mold? How long will it take to fill the mold? How long will it take to cool the plastic and to eject the plastic part from the mold?

We also need to know about the heat conduction—the temperature, the pressure, and the velocity at every point of the mold. Prior to the use

of supercomputers and visualization systems, we had little idea how to even think about these questions, other than to build a physical prototype that was extremely expensive and time-consuming. It was virtually impossible for mold designers to physically see into the mold and observe the physics at work.

Now for the first time our designers can see these phenomena via supercomputing and visualization. A new vocabulary is emerging based on visualization of the computer simulation. More importantly, new insights are emerging. As a result, manufacturing productivity is increasing significantly.

A third example of the application of supercomputing involves research on color, a vital element in many Kodak imaging products. Our photographic scientists are keenly interested in matching physics with perception. This is important because it can lead to increased flexibility in developing color reproduction processes. Color reproduction almost always involves creating color in a constrained way. Since only three dies are used in a photographic paper, or three phosphors in a cathode-ray tube, it is important that these choices be made so that the image quality and color rendition are not compromised in the eye of the observer.

Color theory attempts to describe causal relationships between physical color stimuli from the environment and psychological color sensations evoked by these stimuli. Color stimuli are radiations within the visible spectrum and are described by radiometric functions, whereas color sensations are subjective and are described by words such as *red, blue*, or *green*.

Kodak research scientists, in collaboration with faculty at the University of Illinois through Kodak's partnership in the NCSA, are exploring an elegant mathematical color theory to enable the computation of all possible ways of evoking a given color sensation. The key to this theory is the mapping of color stimuli to the color sensations they evoke. With the supercomputer, researchers can determine how different dies can be used in color reproduction by simulating a standard observer's response to the colors produced by the die mixtures. This flexibility could lead to more cost-effective and higher-quality color reproductions. Although it is still in the early stages, this theory is showing much promise.

ASSIMILATING SUPERCOMPUTING INTO THE R&D CULTURE

Lacking an existing framework for assimilating supercomputing technology into our scientific and industrial culture, a situation that I described earlier, we at Kodak created an approach or a process. While we believe it is *a* model and not *the* model, it is an approach that has served us well and one that we seek to continually improve.

Obviously we realize that we cannot displace whole cultures overnight, so we initiated what might be called a supercomputer assimilation program. We started by organizing a program of on-the-job seminars on the latest applications and developments in supercomputing. In short, we communicate how other scientists use supercomputers. These computational science seminars have featured such distinguished experts as Kenneth Wilson, a Nobel laureate formerly from Cornell University and currently director of a supercomputing center in Ohio, and other leading scientists from supercomputing centers such as NCSA, our valued partner. Donna Cox, for example, has visited Kodak three times in the last 18 months and has held seminars with literally hundreds of Eastman Kodak Company scientists and engineers.

We are also taking advantage of a long-standing forum of exchange within the Kodak research laboratories, the Interplant Technical Conference Series. Kodak scientists and technicians gather three times annually for these conferences to foster and to profit from a greater sharing of ideas, techniques, and applications across a wide variety of scientific disciplines and technological frontiers. The 77th Interplant Technical Conference, titled *Supercomputing and Science and Engineering*, will focus on the developing role of supercomputing technology in R&D at Kodak and will communicate how Kodak scientists and others use supercomputers.

In addition to seminars and conferences, Kodak has another mechanism to keep R&D personnel involved. We disseminate in the R&D community the many technical reports that have been written by those who have used the supercomputer, and we have had over 55 R&D personnel that have traveled from Rochester, New York, to Urbana, Illinois, who have written trip reports about their experiences at the center. We have since installed a T-1 link, but the trip reports have provided valuable insight into how we can facilitate the use of the supercomputer by other Kodak scientists. These reports document the results of the R&D activities and the techniques and computer tools used to generate those results. The terms *supercomputing* and *visualization* are appearing more and more in our everyday vocabulary.

We also have a supercomputer technology planning process, a participatory planning process that facilitates communication and involvement. We recently completed and disseminated to several hundred people, including top management throughout the company, a Kodak supercomputing requirements and provision plan that profiles our past needs and projects our future needs in specific areas of application. This report, which also details a plan to supply computing capacity and visualization systems to the worldwide Kodak R&D community, involved the direct participation of more than 50 Kodak scientists and engineers covering a broad spectrum

of R&D from life sciences to photographic imaging systems to manufacturing technology. This Kodak R&D activity worldwide involves some 8000 scientists and technicians working in the United States, England, France, Germany, Australia, and Japan.

By definition, the supercomputer represents the state of the art. As such, it is used by an elite set of pioneers, people willing to put in the tough work required to tame the tool in return for major payoffs. We call these pioneers computational scientists. They are role-model supercomputer users who communicate and demonstrate the usefulness of the supercomputer. Thus as an additional step, Kodak research management also established the Computational Science Laboratory within the Information and Computing Technology Division.

This laboratory provides one-on-one collaborative assistance to facilitate the effective integration of supercomputing technology into R&D activities. While we do research using computational science methods, we also serve as in-house advocates for the use of supercomputing. Our mission is to be a catalyst. We promote advanced R&D computing technologies and systems that will help people increase their ability to do creative, cost-effective, business-focused research. And we follow up by keeping another of our important stakeholders, top management, involved.

Because it is especially important to communicate supercomputing benefits to our management stakeholders, we formed a Supercomputing Technology Board, consisting of the directors of several research and engineering divisions, to inform management of the experiences gained and the successes realized through the use of our supercomputing program. As Kodak people become more involved with supercomputers, management is continually made aware of the tangible successes resulting from the company's investment in advanced computer technology.

We have made extraordinary progress in supercomputing in a relatively short time. There have been both a remarkable symbiosis and a synergy among scientists and technologists in their applying of computational techniques to their respective disciplines. On balance what has come out of this symbiosis has been beneficial to each faction, be it academic or industrial. Yet there has not been a universal benefit. Such a benefit can be achieved only by an informed and self-confident scientific and technological population that can leverage the kind of cultural change necessary for growth. And that change in growth is predicated on improving communication at all levels.

In closing, I leave you with the word *communication*, because communication is the keystone. We must communicate outside of our organizations, as we are communicating and learning from one another in this symposium, and we must orchestrate change within our organizations by creating new processes of internal communication. I believe that the means

to achieving sound communications are forums such as this and the establishment of processes and organizational advocates that facilitate the use of supercomputing technology within our organizations.

Let us have more of these forums. The results will be the increased efficiency and enhanced effectiveness of supercomputing scientists and engineers in the American workplace.

8

Achieving a Pioneering Outlook with Supercomputing

Lawrence G. Tesler
Apple Computer, Inc.

Apple purchased a CRAY-XMP/48 computer a little more than 3 years ago. It has taken a while for us to develop a range of applications. Currently about 20 different projects are using the Cray, and they involve about 50 engineers. Most of the applications that we have are proprietary and cannot be talked about, but fortunately there are some recent applications that I can discuss for this symposium.

EXTENDING THE RANGE OF APPLICATIONS

The main reason we bought a Cray was to make applications possible that were previously impossible because of the time they took to run. We had circuit simulations, for example, that would have run for 2 months, and it was easier to actually build the circuit and try it out than to wait the 2 months to run the simulation on, say, a VAX. The Cray has helped us a lot, because now we can run those simulations in 1 day.

We had applications that would run overnight, and you had to really plan them well, start them running, and then come back the next day to see the results. If there was a mistake, you had to make a little change and run the application again. Those we can now do in a few minutes, and we can try many variations quickly, as the other speakers have mentioned.

More importantly, there were things that previously had to be done in the batch mode that we can now do interactively, a requirement that Beverly Eccles of Abbott Laboratories talked about very well. But the main

reason we bought the Cray is that we ourselves are a computer company, and our job is to create the computers of the future.

The group that I run, the Advanced Technology Group, is not designing products for Apple. We are doing research on and prototyping of technologies that will apply to future products. So we are looking 3, 5, or 10 years ahead for what might be applicable in future Apple products. For us, the supercomputer is a way to experience the kinds of speed that will be on the desktop in the $1,000 to $10,000 range in several years.

To make that possible, we have created a somewhat unusual setup. The network that we have at Apple includes the CRAY-XMP. We recently upgraded the disk storage on that to about 30 gigabytes. In addition we have an EN-641 that connects us to Ethernet, and we have VAXs and many Sun and Apollo workstations on Ethernet as gateways into the AppleTalk network, which allows us to connect up to the many thousands of Macintoshes that are all around the Apple campus, including more than 1000 in engineering.

On the Macintoshes we have software, for example, NCSA-TELNET, as well as a product from Pacer software that allows us to do terminal emulation, file transfer, and so on, by using convenient menus. We are able to produce not only text but also graphic displays on the Macintosh to access the power of the supercomputer.

We also have, in addition to the standard 50-megabyte hyperchannel, an 800-megabyte-per-second ultrachannel that gives us very high bandwidth video out, essentially, to a number of high-resolution monitors, so that we can get direct interaction with the Cray. When we do that, we are temporarily trying down the entire machine for one user. If someone is rotating an image in three dimensions, the machine is dedicated as a personal computer to that user for a few seconds while that is going on. Some of the applications I will discuss rely on that capability.

We use the Cray both in product development and in research. We use it for circuit design simulations, and that has saved a lot of time in proving designs. Disk head design is an application I will discuss; industrial design is another.

The disk head design project is an interesting one. The goal is to make the recording head fly at a constant height over the disk surface, on the order of 10 microinches. The shape of the head and the shape of the medium and the aerodynamics all interact. If the system is not set up well, then the head will crash, or the head will be too high above the disk to get a clean signal. We wanted to know what the effects of various parameters were.

An interesting problem we ran into was that if the head itself, the air bearing, has a resonant frequency that corresponds to the frequency of the rotation, oscillations result. To understand that better, we worked with Jim

White from the University of Santa Clara. The disk head can have little levels and other shapes that affect the aerodynamics. Our engineers came up with a set of equations based on the geometry of the head.

One of the problems is that the medium itself is not completely flat. It can be a thousandth of an inch off flatness, and when the head is flying a few hundred thousandths of an inch over the disk, that variation can cause a problem because essentially, the head is like a cruise missile trying to go over peaks.

In a simulation, the head can be shown as flying between 150 and 350 or so nanometers over the surface. By varying the shape of the head, the engineers can run the simulation over and over. The simulation takes only 20 minutes to run, and the engineers can keep playing with it until they achieve satisfactory results.

Another concern is that there may be a problem caused by a slight bump in the medium. A little of the oxide may have a small bump in it, and a result may be that the head can really bounce. And of course if it bounces too much, it will crash.

Another area to explore is what happens if there is a jolt to the head, which can happen because someone moves the drive while it is running. After a seek, when the head comes to a stop, there is a similar jolt. We need to know how long it will take the oscillation of the head to settle down so that we can actually start to do a read or write.

Why is Apple studying all these things when in fact we do not manufacture disk mechanisms? The reason is that we work with vendors of heads and media, and vendors of drives, and they come to us with claims of why the next generation of disks is going to be so much better than the last. We need to be able to evaluate their claims, because if we simply go along with them and something does not work well, then we might have to shut down our production, and that is a serious consequence.

Product Design

We have also used the Cray for product design, or packaging. We have used three different applications: (1) thermal analysis, (2) structural analysis, and (3) mold flow similar to that discussed by Cliff Perry.

For thermal analysis we have used a package called ANSYS, which is a finite element program displaying the output graphically on a Macintosh II. For example, we have modeled a personal computer board, with major heat sources displayed as small blue areas. A simulation is run until it settles into a steady state, which occurs after the computer has been on for a while. Then the task is to see what temperatures the various components have reached.

It is possible to tune the heat flow in the system—by adjusting the cooling air flow and the layout—so that it is all in the range of about 143 to 152°F, but some hot spots may remain. By playing with the parameters, engineers can try to get the temperatures within an acceptable range for the components.

Another application, called NEKTONics, is a finite element program that is used for structural analysis related to the cooling problem. A small object represents the edge of a cooling vent. Just above and below that object is a vent; a piece of plastic separates the vents. As air is drawn in through the package, the flow is depicted. The question is, What will the temperature be after a certain period of time, given certain assumptions about air flow and the initial conditions?

This program can show potential problems. For example, if air flows past a particular point and loses velocity, it also loses its ability to cool. It is possible to play with the shape of a particular edge and solve the problem of reduced air flow. By manipulating with a computer-aided design (CAD) program the shape of a vent edge, a different flow pattern was obtained, and the result was that the velocity loss was reduced.

A third application is used for a mold flow problem. What is interesting in this example is that the product we used this application for was the large-size, extended keyboard for the Macintosh II. A keyboard for a Macintosh has various places on the surface that, if looked at in just the right way, are small dark areas. These are weld lines where the plastic has come through the mold and welded together, and they are not very good to look at. The problem was to try to improve the keyboard's appearance so that people would stop telephoning Customer Support to ask why they couldn't clean their keyboards.

The approach to this problem was to break the keyboard's surface down into very small polygons and then to run a simulation that showed the filling of the plastic in the mold. The point at which the plastic in two paths merges together becomes a weld line. The idea is to try to control conditions so that the temperature of the two is about the same and the weld occurs in a place where it will not be noticed by the user. This is the case in the newly designed keyboard, not in our original design.

In a two-dimensional display of mold flow, different colors represent different time periods so that it is possible to see the history of the flow. Now there is an interactive program that shows the process in real time. The engineer can use a mouse to select a specific part of the picture and then can view a blown-up zoomed-in view of just that part. This gives the engineer the ability to focus on parts of the process. Our application does not have the aesthetics of the visualization that Cliff Perry described or the ability to show multiple parameters at once. Instead, we traded that off to be able to get interactive capability for the engineer.

We also can do pressure and temperature plots, and so on. What is important is that in the end, the engineer gives to the plastic maker a drawing that shows the key things that have to be done. Basically the approach to solving the problem of getting the plastic to flow at the rate that we wanted was to indicate places where the inside of the mold was narrower, which slowed down the plastic flow so that we could catch up in other places. This approach gave the result we wanted; the weld lines were exactly where we wanted them. The illustration was done with a program called PIXELPAINT on the Macintosh II, by starting with the CAD diagram and simply taking the data that came out of the simulation.

The benefits of using the supercomputer for product design are increased savings of time and money—hundreds of thousands of dollars—made possible by fewer tooling runs plus the much greater advantage of getting products to market faster. We can get products to market months faster because we know that the likelihood that the first mold is going to work is much higher. We also do not have to wait months for another mold if there is something wrong with the first one, and we can improve the various parameters of the design and get better quality.

Research

Now we also use our supercomputer in research. At Apple, we have been doing neural network simulations to better understand how to use different neural net models for learning. In addition, we have simulated a cochlear model that is used in a speech recognition project. The idea is that, to be usable, any speech recognition system has to be able to work in a noisy room. One approach to achieving that is to try to actually model the human ear, which has a comb of hairs that is able to sort out different frequencies and to measure, essentially, the energy at each different frequency.

Some work had been done at Schlumberger Research by Dick Lyon, who recently came to Apple. What we decided to do was to take the same type of model that he had implemented, implement it on the Cray, and then animate the results.

The result is a plot, called a correlogram, that shows low frequencies at the top and high frequencies at the bottom. Viewed from left to right, it shows various correlations of different timing sets. Essentially it enables the engineer to "see" what the ear "sees" when it hears a sound. An utterance can be visualized as pillar-shaped forms that represent the main frequency and as other forms that represent other, weaker frequencies. If there is noise in the room, the background becomes fuzzy, but it is still possible to see a pattern of frequencies standing out. This is the beginning of a very

long research project to try to emulate the power of the human ear to sort out noise.

For neural net simulation we have been able to get very high rates on the Cray. Using even only one processor on the X-MP, we have been able to do about 10 million connection updates per second and also to animate the results to get a feel for how a neural net learns.

So the benefits for research are, again, more rapid prototyping. We can try many alternatives. People are willing to try things if they can get results in a few minutes, or even interactively, but the main thing is that we are now much bolder. We will try things that we would not have tried before. One of the chips that we are designing currently is one we probably would not have attempted to design previously because people thought it would not work. When we simulated it, we found that it would work, and we went ahead and built it and in fact it did work So I would say that the main impact of the supercomputer is that it makes us more comfortable with taking bigger risks.

ADDING SUPERCOMPUTING CAPABILITY

Visualization, as everyone participating in this symposium has explained, is an absolutely key capability. Having a fast network is very, very important, and we continue to upgrade the speed of our network so that people can get higher bandwidth between the user interface and the supercomputer. One big problem is simply operating the supercomputer center. It accounts for a major portion of our budget, and we are always under pressure to add new power to it. It is competing always with other needs such as upgrading the network and adding minisupercomputers and workstations. The operations end is something that anyone thinking of buying a supercomputer really must consider.

Two years ago we brought in a person from our Management Information Systems Department to manage our supercomputer center, and we have hired several people who are expert in using the Cray and other supercomputer engineering systems to work there.

The last hurdle, as other people have mentioned, has been to get the users to use the supercomputer. The way we do it is that we have a small group of people we call Cray evangelists. They do not appear on television; they walk around. They go to engineers and try to find out what those engineers do, and then they try to match them up with applications on the Cray or help them write their own applications on the Cray. All of the applications I have discussed in this symposium have come from that effort, which is very similar to what was described as the effort that goes on at Kodak also.

DISCUSSION

Edward Abrahams: Have you who have tackled this problem of convincing users to use the supercomputer found some techniques that were not so productive? Presumably you have mentioned some of the ones that are productive. What techniques did not work, so we can avoid them?

Lawrence Tesler: Trying to convince somebody who is very negative is probably the one thing that isn't worth doing. In other words it's important to find people who immediately see the benefits of supercomputing and to get them to start using it. Then their colleagues will realize that they can use supercomputing also.

Beverly Eccles: Yes. Seek the champions for the cause.

Clifford Perry: I don't have too many keys to failure, but one key to success is involving the users from the very beginning in participative planning. We actually sent out letters to literally hundreds of the heavy users of our traditional high-end mainframe computing facility, asking them to participate in an idealized design of a supercomputing facility and to think about what the attributes of that particular center should be. Would it offer one-on-one collaborative assistance? Would it offer transparency vis-a-vis using that computer or the high-end mainframe? How would it be administered? How would it be charged out? What help would be rendered to the users?

When only top-down decisions are made, people don't use the computers. The decision-making process has to be top-down, bottom-up, middle-out. We focused on the bottom-up and middle-out, and then when Larry Smarr came and mapped what he had to offer against the idealized design that was documented and was formulated by the participation of those whose lives would be affected by the advent of the supercomputing facility, we found that we had an immediate, captured market.

Generally, it takes about 2 years to justify the use of a supercomputer onsite, and it takes upwards of $250,000 to $300,000, as has been published by the Minnesota Supercomputing Consortium. It has to be done in a participative manner, in my opinion, or it won't work.

Mel Schmidt: Could all the panelists briefly describe how they determine allocations within their organizations? Are there any mechanisms for billing the users?

Beverly Eccles: Within Abbott, computer resources are basically free. The resources are supplied and the expense goes into the budget, but individuals are not concerned about how much disk space or how much of the central processing unit they are using. Those resources are simply available for us, and the scientists feel very comfortable in that environment.

Clifford Perry: At Kodak we have an arrangement with NCSA that every user who logs on—and we have an administrative procedure to do that—is billed directly in their division. We have allocated, if you will, $100,000 chunks to sets of people.

Lawrence Tesler: At Apple as at Abbott, all the shared computer resources that are used by more than one department are essentially free. On our financial reports from the Apple Product Division, we break out the entire budget for this computer operation. It is weighed as a whole as a percent of the entire R&D budget.

PART D

CONCLUDING REMARKS

9

Summary

Doyle D. Knight
Consortium for Scientific Computing
The John von Neumann National Supercomputer Center

These proceedings of the symposium on supercomputers have focused on two major topics. First, Jack Worlton and Steve Chen described the evolution of supercomputer technology over the past 20 years and projected future trends in improved processor performance, increased memory, and rapidly expanding parallelism. They also emphasized the role of algorithm development, noting that improvements in performance associated with the development of new algorithms are increasingly important.

Second, three presentations focused on current applications of supercomputers in industry. Beverly Eccles (Abbott Laboratories), Clifford Perry (Eastman Kodak Company), and Larry Tesler (Apple Computer, Inc.) provided a series of examples of applications of supercomputers in their corporations. Two key points were emphasized:

- Supercomputers provide the opportunity to design new products, ranging from film emulsions to computer keyboards, with a degree of accuracy heretofore unachievable with conventional computers and at a cost oftentimes far lower than that associated with experimental methods (e.g., development of prototypes).
- Supercomputers can improve the productivity of designers by significantly reducing the time required to evaluate a new idea. The rapid feedback of results enhances creativity.

These three researchers also addressed the issue of integration of supercomputer technology into industry. Four important points were stressed:

- Industry must first recognize the potential benefits of supercomputer technology in research and development.
- A core of supercomputer "evangelists" must be established initially within a corporation. This core group—people who are dedicated to using supercomputing as well as to explaining and communicating its advantages to other scientists in industry—will provide the leadership and incentive for the adoption of the technology by the larger group.
- "Bottom-up" planning is necessary for successful incorporation of supercomputers into industry. The current computer "habits" of researchers and designers must first be understood before any major changes can be implemented.
- Close collaboration between academia and industry is needed to provide improved software tools and training for students, who will become the researchers in the industry of tomorrow.