The Pennsylvania State University

The Graduate School

Department of History

ARGONNE NATIONAL LABORATORY AND THE EMERGENCE OF COMPUTER AND COMPUTATIONAL SCIENCE, 1946-1992

A Thesis in

History

by

Charles Nelson Yood

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

August, 2005

The thesis of Charles Nelson Yood has been reviewed and approved* by the following:

Robert Proctor
Professor of History of Science
Stanford University
Thesis Advisor

William Pencak
Professor of American History
Chair of Committee

Gary Cross
Distinguished Professor of Modern History

Rosa Eberly
Assistant Professor of Communication Arts
and Sciences, and English

Sally McMurry
Professor of American History
Head of the Department of History and Religious Studies

* Signatures are on file at the Graduate School.

Abstract

This dissertation uses the Applied Mathematics Division (AMD) of Argonne National Laboratory (ANL) as a window to explore the emergence of computer and computational science as independent scientific disciplines. The evolution of the computing activities at Argonne reflects broader issues concerning technology, identity, professionalization, and the social organization of science.

While Argonne's development of digital computer technology is a significant part of this story, I focus on the AMD's efforts to integrate computers – and their attendant personnel – into the scientific process. In particular, the pursuit of "computational science" required that applied mathematicians be incorporated in all stages of science and engineering practice -- from problem formulation to the definition of what constituted a solution. Arguments for such a collaborative structure drew on Cold War rhetoric, debates within the mathematical profession, and issues surrounding the increasing quantification of the sciences. Simultaneously, applied mathematicians sought to define a new research agenda that balanced their duties to provide mathematical expertise to other scientists with their desires to conduct their own research.

Despite the intentions of AMD directors, the interdisciplinary collaboration that computers were supposed to foster failed to materialize as envisioned. The emergence of an independent computer science, technological innovations, and the development of computer expertise by other scientists effectively limited the extent of collaboration. Beginning in the mid-1970s, though, the development of supercomputers, together with a new federal emphasis on high-speed computer networks created new opportunities for mathematicians, computer scientists, and scientists to work together. Impetus for collaboration was fueled by a number of different national concerns, including the Japanese Fifth Generation program, the need to support the domestic supercomputing industry, and pressures to make supercomputers readily accessible to American scientists. The federal government responded by creating the High Performance Computing program in the late 1980s, followed by the Grand Challenge Program of the 1990s in an effort to foster computational science – considered a third methodology, alongside theory and experiment, for doing science . Along with enabling computational scientists to tackle problems with broad implications for science, economics, and national security, another result was a significant reorientation of computer science research.

Table of Contents

List of Abbreviations

ACF: Advanced Computing Facility
ACM: Association for Computing Machinery
ACRF: Advanced Computing Research Facility
AEC: Atomic Energy Commission
AECCAG: Atomic Energy Commission, Computer Advisory Group
AFIPS: American Federation of Information Processing Societies
AMD: Applied Mathematics Division
AMP: Applied Mathematics Panel
AUA: Argonne University Association
AVIDAC: Argonne's Version of the Institute's Digital Automatic Computer
CSCC: Concurrent Supercomputing Consortium
CTD: Computing and Telecommunications Division, Argonne
DARPA: Defense Advanced Research Projects Agency
ECF: Experimental Computing Facility
ERDA: Energy Research and Development Agency
FCCSET: Federal Coordinating Council for Science, Engineering, and Technology
HEP (DENELCOR): Heterogeneous Element Processor
HPC: High-Performance Computing Initiative
HPCC: High-Performance Computing and Communications Initiative
MCS: Math and Computer Science Division, Argonne
NAML: National Applied Mathematics Laboratory
NATS: National Activity to Test Software
NDRC: National Defense Research Council
NSF: National Science Foundation
ONR: Office of Naval Research
OSRD: Office of Scientific Research and Development
OSTP: Office of Science and Technology Policy
SCS: Scientific Computing Staff, Department of Energy

List of Archive Abbreviations

AUA: Argonne University Association

COHC: Computer Oral History Collection, Smithsonian Institute

DOE Archives: Department of Energy Archives, Germantown, MD

GLFRCNARA: Great Lakes Federal Record Center, National Archives and Records Center, Chicago, Illinois

MCS Archives: Math and Computer Science Archives, Argonne National Laboratory

NARA: National Archives and Record Center, Washington, D.C.

NARA II: National Archives and Record Center, College Park, MD

List of Pictures

Acknowledgements

I would like to thank the following institutions and people who made this project possible. Shortly after beginning my project in 1998, it became clear that the Department of History at Penn State University was reluctant to provide funding for research in the History of Computing. In fall 1999, I received a Mark and Lucy MacMillan Stitzer Program Support Endowment in History. This small grant enabled me to visit the Charles Babbage Institute at the University of Minnesota, where I began the preliminary research for my project. In spring, 2000, I applied for and received a newly created fellowship at the Department of Energy for the History of Computing. At the DoE headquarters in Germantown, MD I met Skip Gosling, the Chief Historian for the agency and the assistant historian, Terry Fehner. Although there was no set project in mind, they were excited by the possibilities of producing a historical account of the DoE's role in computing and seemed excited by my ideas and enthusiasm. I owe them a tremendous debt of gratitude, for without their support it is unlikely that I would have continued in my graduate program.

The DoE fellowship was the brainchild of Jim Corones of the Krell Institute in Ames, Iowa. Dr. Corones had been a computational physicist at the National Laboratory in Ames and was keenly interested in producing a scholarly account of the emergence of computational science. It was his suggestion that I focus on Argonne National Laboratory because of its long history of computer use, plus the fact that many of the computer pioneers from the lab were still alive and willing to be interviewed. It was Jim who first brought it to my attention that computer science and computational science, although closely related, were two distinct entities. The DoE/Krell Institute funding was substantial, and I was able to begin week-long research visits to Argonne beginning that summer.

At Argonne, I met many interesting, supportive, and friendly people. I would especially like to thank Margaret Butler, a founding member of the Applied Mathematics Division (AMD) in 1957. She willingly submitted to three long interviews, many emails and phone conversations, and donated to me personal materials related to computing. Margaret was also instrumental in putting me into contact with current and retired members of the AMD, including Joe Cook, William Cody, Wayne Cowell, Paul Messina, and Bill Miller, and she also read and commented on my first two chapters. In addition, I would like to thank the aforementioned members of the AMD for agreeing to be interviewed at length and for donating their personal papers to me as well. Gail Pieper was particularly helpful for sharing her office at Argonne with me and for providing encouragement and support during my visits to Argonne. Her knowledge of the history of the AMD, and especially its personnel, was invaluable. Judy Beumer, my main contact within the Math and Computer Science Division at Argonne, was tireless in her efforts to make my visits to Argonne productive and she spent many hours of her own time facilitating my research.

My research activities attracted the attention of Dr. Catherine Westfall, the Chief Historian at Argonne. In early summer, 2002, she offered me an ANL Graduate Fellowship -- the first such fellowship awarded to a historian. This fellowship allowed me to finish my research in Chicago, Urbana-Champaign, Germantown, and Washington, D.C. Catherine has been actively involved in my progress throughout; not only did she read and comment on drafts, but she also made an effort to include me in scholarly conferences which she helped to organize. It was at her initiative that I met other historians doing work on the national laboratories. Catherine's support and enthusiasm for my project kept me going during the long writing stage. Elizabeth Paris, another historian hired by Catherine, provided critical comments in the earliest stages of my project and also introduced me to many of her colleagues at the History of Science Society conference.

In 2003-2004 I received a National Bureau of Economic Research, Science and Engineering Workforce Project Dissertation Fellowship. I want to thank Dr. John Trumpbour, the only historian in a "sea of economists" on the SEWP panel, for supporting my project. His astute comments on my proposal and progress report enabled me to get additional funding from the NBER in spring 2004.

Lastly, I received a dissertation Fellowship from the National Science Foundation in fall 2003 and summer 2004. This fellowship enabled me to finish the draft of my first two chapters and provided support while I was unemployed in fall 2004.

While financial support was crucial to the success of my dissertation, without the support of my advisor, Dr. Robert Proctor, none of this would have been possible. Dr. Proctor's unflagging support during my years at Penn State was often the only thing that kept me in the program. Robert exposed me to opportunities which I would not have had otherwise – most especially by bringing me along in 1999 to the Max-Planck-Institute für Wissenschaftsgesch., Summer Academy in Berlin. In addition to making me feel like a true scholar for the first time, I also had an opportunity to spend time with the person who would later become my spouse. I am sure that my slow progress was exasperating at times for Robert, but throughout the years he continued to offer unflagging encouragement. Moreover, he was always interested in my ideas and ready to offer his own suggestions about how they might be refined or extended. His advice over the years remained consistent -- "just start writing" -- although I didn't quite take it to heart (or understand it) until mid-way through the dissertation. There were many occasions that Robert's "invisible hand" made it possible for me to complete my graduate program. I would also like to thank two other members of my committee, Drs. William Pencak and Rosa Eberly, for their last minute work on my behalf and for their incisive comments on my dissertation.

Finally, I want to thank Sarah Goodfellow, whose love, patience, support, and companionship have made the past five years a wonderfully exciting adventure. Her various roles as scholar, confident, partner, and mother sustained me throughout the process and continues to inspire me.

To Sarah and Elliot, who enrich my life immeasurably.

Introduction

Computational Science Enters the Laboratory

Today, if you leave downtown Chicago and drive south on the ten-lane wide Interstate 55 for twenty-two miles, you pass a seemingly endless stretch of concrete pavement, industrial sites, commercial centers, and, eventually, suburban homes. Near the end of the drive, which at any time of the day may take upwards of an hour, is a small non-descript sign – the kind used by authorities who need to indicate the location of a site but do not want to advertise it – for Argonne National Laboratory (ANL). Exiting and turning south on Cass Avenue quickly brings you to yet another sign, this time advertising a forest preserve. Interestingly, the entrance to the preserve is also the Main Northgate Entrance to Argonne. For almost a mile, Northgate Road winds through heavy woods where it is common to see deer, and especially the albino deer which have found Argonne a haven for over fifty years. Depending upon the time of the season (or how tired one is), it is easy to imagine that the road will continue ever deeper into the woods. The guard station, however, and the United States soldiers manning it bring one back to reality. Argonne National Laboratory is, after all, a government research facility established in 1946 by the Atomic Energy Commission to assist in the development of nuclear reactors. As such, its history has been inextricably intertwined with Cold War politics and science, national security, and, most recently, the War on Terror.[1]

The drive to Argonne can be seen as symbolic. When ANL was established these 1500 acres of DuPage County were in the boonies-- working with radioactive materials and the construction and operation of experimental nuclear reactors was considered too risky to be near populated areas. Yet in the same way that Chicago evolved and expanded to envelop the Laboratory, the activities of the Laboratory have also evolved and expanded. Argonne's main activities have expanded beyond the development of nuclear reactors to include everything from studies of the atomic nucleus to global climate change. It currently employs 2,900 people, including almost a thousand scientists (600 with doctorate degrees), and its operating budget of almost half a billion dollars supports over 200 separate research projects. The lab also proudly proclaims that since 1990 it has worked with more than 600 companies as well as numerous federal agencies and other organizations.[2]

Argonne's website uses 1990 not only because it is a nice round number. That was the year that Alan Schriesham, Argonne's director, submitted a funding proposal from the Math and Computer Science (MCS) Division to the Department of Energy; Schriesham's letter of support stated:

[1] When I began my research at Argonne pre 9-11-2001, passing the guard gate required a flash of my visitor's pass. After 9-11-2001, my car was routinely searched, even by people who knew me and saw me daily. The post 9-11 climate also made it more difficult for me to access particular archival materials related to Atomic Energy affairs at the National Archives – even if these archives had previously been cleared for researchers.
[2] www.anl.gov

> High-performance computing has become firmly established as the third mode of scientific research. Specifically, it has led to the development of *computational science* as a new methodology of scientific inquiry that complements and broadens the traditional methodologies of laboratory experimentation and theoretical analysis.[3]

This short passage is more than a proclamation of computational science as a new way to do science; it also hints at the historically significant convergence of technology, human organizations, and funding structures that made this new methodology and discipline – computational science -- possible. The proposal, submitted as part of an effort to establish a parallel supercomputing facility at Argonne, was inspired by the new $4.7 billion, five-year federal High Performance Computing and Communications (HPCC) program that was pending in Congress and would be passed the next year. Argonne wanted to generate a strong presence in computational science, and access to supercomputers was crucial to this endeavor. Establishing a center within the MCS would thus ensure the lab's future relevance while also building on a long tradition of computer research at Argonne.

"Computational" implies computers, and here Argonne's history is deep; so deep in fact that it provides an excellent environment in which to explore the evolution of computing in general, and in particular, the emergence of computational science. Some of this history can be seen in the material culture of the laboratory itself –in buildings for instance. In 1950, if I had visited the computing facilities at Argonne, I would have found myself in the basement of the Physics building or in a room in the Reactor Engineering facilities. In the mid-1960s, this visit would have taken me to a brand new building for the Applied Mathematics Division (AMD) that housed the main computers for the laboratory along with the offices of mathematicians and computer scientists. In

[3] Alan Schriesheim, to James F. Decker. "A Parallel Supercomputing Facility for Computational Science," March 22, 1990. Meetings with Don Austin, Friday April 13, 1990, box 2, MCS Archives, Argonne National Laboratory.

1983, this same building was reserved solely for the Math and Computer Science (MCS) Division, and the computers there were primarily of experimental designs.

The evolution of the computing activity from playing a minor support role within physics and reactor engineering, through achieving Divisional status in 1956, to its latest incarnation in 1983, suggests that studying this change over time can provide insight into broader currents effecting science during the same period-- and the emergence of a new science.

In this dissertation, I argue that computers are a unique scientific technology in that they have spurred the creation of entirely new scientific disciplines and new methodologies for scientific investigation. Cyclotrons did not produce "cyclotron *science*" nor did particle accelerators create "particle accelerator *science;*" computers did produced computer *science* as well as computational *science*. However, my dissertation is not a history of computers *per se*. On the one hand, I tell a story about the history and evolution of the Applied Mathematics Division at Argonne. But this narrative simultaneously provides a framework for exploring some of the ways in which computers have provided the material basis on which different professional identities within the sciences have been constructed, and also how computers have provided a material basis for interdisciplinary collaboration. In the late 1940s computers were perceived as ultra-fast calculators which could be used by scientists to solve complex problems that were analytically intractable, but could potentially be solved using numerical approximations. Almost immediately however, people engaged in the construction and operation of these new electronic digital computers realized that their efficient use required a corresponding effort in mathematical research. This marked the beginnings of the different activities that would eventually coalesce into the discipline of computer science.

Whether or not computer science was a *science* was hotly contested, and not just by those outside the field. As late as 1989, the computer science profession was still issuing reports with titles such as “Computing as a Discipline” in which its authors presented “a new intellectual framework for the discipline of computing.”[4] Part engineering and part mathematics, computer science did not fit neatly into either of the traditional categories of theoretical or experimental science. I argue that much of this debate over the status of computer science was fueled by differing conceptions of the *meaning* of computers. For those who used computers in their scientific research, computers were a tool, and for a variety of reasons these people were unwilling to support research that detracted in any way from their use of this tool. Using a computer, however, required experts who could translate scientific problems into a form suitable for computation. In contrast, mathematicians and computer scientists considered computers objects of scientific investigation in their own right, and thus felt that they (the mathematicians and computer scientists) should not be relegated to providing support services to other scientists, but should be allowed to carry out their own research in the foundations of computing.

Further informing the debate over the meaning of computers were larger visions for how these engines of computation could facilitate collaborative, interdisciplinary research efforts. In the early 1960s, there was little consensus as to how computers might be integrated into scientific research. The solution to this, and other questions, was not

---

[4] Peter Denning and others, "Computing as a Discipline," Communications of the ACM 32, no. 1 (1989).

obvious. Understanding the history of computational science, I propose, requires recognizing that, by its very nature, computing was a *social activity*.

As a social practice, computing called on the skills of engineers, mathematicians, programmers, and operators. Furthermore, because applied mathematicians had strong mathematical foundations, an understanding of the physical sciences, and, most importantly, a knowledge of computers, some believed that they would be the key players in scientific research that used computers. Facilitating such collaboration, however, proved to be quite difficult. As I will show, throughout the 1960s and 1970s both disciplinary boundaries and technological change created barriers that impeded the realization of this vision. As a result, in the 1970s computer scientists were placed in the unenviable position of having to justify their existence. In a perpetual state of professional "crisis," computer scientists sought to articulate the epistemological foundations of their discipline. In this they were relatively successful, but even as recently as 1984 it was largely unclear what role computer scientists played in the creation of new scientific knowledge.

Tracing the professional trajectory of computer science within the context of changes in both technology and disciplinary identity is but one of my goals. My more ambitious project is to argue that computational science constitutes a "third mode" of scientific inquiry that emerged in the mid to late 1980s. In particular, I argue that computational science is *the methodological extension* of what scholars refer to as "Big Science."[5]

---

[5] Derek J. de Solla Price, Little Science, Big Science (New York: Columbia University Press, 1965), Alvin M Weinberg, Reflections on Big Science (Cambridge: MIT Press, 1967). For a more recent historical examination of Big Science, see Peter Galison and Bruce Hevly, eds., Big Science: The Growth of Large-Scale Research (Stanford: Stanford University Press, 1992).

If we parse Alan Schriescham's statement above we see this basic argument:

high-performance computing[6] → computational science = a new methodology of inquiry

This formulation was by no means original; these same words were repeated in countless proposals and eventually became the basic premise underlying the $4.7 billion, 5-year High Performance Computing and Communication (HPCC) program in 1991.[7]

I maintain that the direct relationship posited by the above equation obscures the contingent nature of computational science. Computational science did not emerge fully formed from the technology of supercomputers like some modern-day Athena springing from Zeus' head. Supercomputers did make computational science viable because of their ability to simulate almost any phenomenon. And computational scientists, who came from almost every scientific discipline, made their use of simulations the centerpiece in claims for methodological distinction. Scientists, however, had been conducting simulations on high-performance computers since the early 1960s, and yet it would take another twenty-five years before "computational science" exploded onto the scientific scene. What finally crystallized computational science, I argue, was the coalescence in the mid-1980s of a shared vision among scientists, computer scientists, programmers, funding agencies and, above all, the federal government. This shared vision, in turn, was contingent upon events external to the world of science.

By far the most significant of these external events was the launching of Japan's Fifth Generation Computer Project in 1981. The project intended to create "intelligent

---

[6] The terms "high-performance computing" and "supercomputer" are interchangeable. Both, however refer to a constantly changing object; given the rapid pace of technological innovation, yesterday's supercomputer is today's Palm Pilot. I discuss this issue at some length in Chapter 3, but in general I use supercomputer to refer to the most advanced computers in existence at any given time.

[7] For an analysis of how sentences, when repeated again and again in different contexts, gain a scientific and cultural currency which they may not deserve, see chapter 2, Bruno Latour, Science in Action (Cambridge: Harvard University Press, 1987).

machines" and challenge American supremacy in supercomputing. The computer and computational science communities made sure that policy makers in the United States perceived the Japanese initiative as nothing less than a direct threat to national security as well as the country's technological, scientific, and economic future. Reaction was swift, stimulating a broad response among federal agencies such as the Department of Defense, Department of Energy, the National Science Foundation, NASA, and Congress. This was part of a broader mobilization; in the private sector, monopoly laws regulating industrial consortia were relaxed in an effort to speed the development of new computer components. Money for the creation of experimental computing facilities, education in computer science, and the development of computer networks was also forthcoming from various agencies. The interests of computer scientists, computational scientists, and the federal government converged in response to the Japanese challenge, and thereby set the stage for the HPCC seven years later.

Before this convergence could occur however, kinks needed to be worked out. In particular, although federal funding for supercomputers had increased, there was little coordination and much duplication of effort. In an attempt to address this problem, after years of planning and negotiation with the computer and computational science community and industry, Congress passed the federal High Performance Computing and Communication (HPCC) program in 1991. The HPCC explicitly recognized computational science as a "new paradigm" for scientific investigation that complimented theory and experiment.[8] The central goal of the program was to develop the hardware, software, and human resources necessary to solve "Grand Challenge" class problems.

---

[8] D. Allan Bromley. The Federal High Performance Computing Program. Washington, D.C.: Office of Science and Technology Policy Executive Office of the President, 1987. HPCC Program Correspondence Files 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

These were defined as "fundamental problems in science and engineering, with broad economic and scientific impact that could be advanced by applying high performance computing resources."[9] Initial Grand Challenges included problems in: materials sciences; semiconductor design; superconductivity; weather, climate, and global change prediction; structural biology; drug design; transportation; oil and gas recovery, nuclear fusion; combustion efficiencies; and radio astronomy.[10]

While the Grand Challenge subjects might seem open-ended, their methodology was not: computational science would provide the answer. HPPC funding to address Grand Challenge problems, however, was contingent upon a computational science group demonstrating a strong interdisciplinary effort that included academia, government labs, and above all, industrial partners. Thus, the HPCC did much more than endorse computational science as a third methodology; it also institutionalized the socio-scientific environment and funding structures in which it would be developed. Where previously these relationships had often been negotiated and then renegotiated by participants, the HPCC provided the guidelines for not only what kinds of problems would be addressed, but also who would participate and in what manner:

> Collaborative groups will include scientists and engineers concerned with Grand Challenge areas, software and systems engineers, and algorithm designers. These groups will be supported by shared computational and experimental facilities, including professional software engineering support teams, linked together by the National Research and Education Network. Groups may also create a central administrative base, which can be located anywhere on the network.[11]

Computational science, as promulgated by the HPCC, was, therefore, in every respect, the embodiment of what scholars refer to as "Big Science." Historian Dominque

[9] Ibid.
[10] Ibid.
[11] Ibid.

Pestre encapsulates well the characteristics of Big Science that have increasingly come to inform modes of scientific inquiry in the second half of the 20th century:

> It [Big Science] refers to heavy equipment (reactors and accelerators, for example) whose use is shared; the 'science made by committees' (which will now decide who is permitted to experiment and under what conditions); work in teams (a collaboration at CERN in the 1980s could involve 400 scientists and last ten or fifteen years); national programs financed by contracts and aimed at resolving practical, military or technological, problems; and of course the state, the increasing important if not exclusive supplier of funds.[12]

It should not be surprising that supercomputing is a big science. However, unlike scientific fields such as high-energy physics, which started small and then became big science during and immediately after World War II, from the beginning both the technology and organization of computing were *products of big science.* The first digital computer, the ENIAC had been funded by the Navy's Ballistics Research lab (BRL) and built by teams of engineers at the University of Pennsylvania's Moore School of Engineering. Seen as a means to automate the creation of ballistics tables, as proposed in 1943, the ENIAC represented an engineering project of an unprecedented scale that would be almost one hundred times larger than any electronic device then in existence. When finished, after the war in 1945, it had cost the federal government almost a half a million dollars (original estimates were in the neighborhood of $150,000). Clearly, digital computers were an offspring of Big Science. "Without the vast research funding and the atmosphere of desperation associated with war," historian Paul Edwards writes, "it probably would have been years, perhaps decades, before private industry attempted such a project. The ENIAC became, like radar and the bomb, an icon of the miracle of

---

[12] Dominique Pestre, "Science, Political Power, and the State," in Companion to Science in the Twentieth Century, ed. John Krige and Dominique Pestre (London: Routledge, 1997), p. 72.

government-supported 'big science.'"[13] It was the descendents of the ENIAC, both technologically and socially, that made computational science possible.

While the intellectual lineage of theoretical science is traced back to the pre-Socratics, and the origin of our experimental method is said to be found in the small-scale experiments of Galileo, Boyle, and Newton, computational science is an entirely different kind of animal. This "third branch" of science is the methodological product of a particular way of conducting science that is high-tech, collaborative, interdisciplinary, and very expensive. The emergence of computational science was contingent; disparate groups of scientists, mathematicians, and computer scientists found common cause with federal funding agencies in the face of a perceived threat to America's technological supremacy. These groups rallied around supercomputers, a technology created and fostered by the dynamics of big science. What makes computational science especially rich for historical analysis, though, is that it *was not new*. Scientists had been using computers to do simulations – the basis for computational science – since the ENIAC was used to test the feasibility of a thermonuclear weapon in 1946. What was new however, were the *claims* of computational scientists that they were the vanguard of a new kind of scientific researcher. Thus, I examine why these claims surfaced when they did and why they attracted so much attention.

In my approach to this topic, I find much inspiration in the brilliant work of historian Paul Edwards. His 1997 book <u>The Closed World: Computers and the Politics of Discourse in Cold War America</u>, is a deliberate attempt to write a "counter history" to the two main threads of computer historiography that fall roughly into the categories of

---

[13] Paul N. Edwards, <u>The Closed World: Computers and the Politics of Discourse in Cold War America</u> (Cambridge: The MIT Press, 1996), p. 51.

intellectual studies on the one hand, and engineering/economic history on the other. As he points out, these parallel stories reflect the hybrid nature of computers in that computers are constructed of both "hardware" and "software".

The hardware genre falls solidly within the history of technology. These histories tend to focus on the ideas and techniques that contributed to the development, advancement, and diffusion of various engines of computation. With nods to the abacus and other early calculating machines, these stories generally pick up with the Difference and Analytical Engines of Charles Babbage in the 1820s, followed by the punched-card machines of Herman Hollerith in the 1890s, and continue to the construction of analog differential engines of the 1930s. Not surprisingly, World War II is a watershed event in these histories as electromechanical calculators began to be challenged by the development of electronic digital computers. Following the war, the stories branch out to include histories of commercial equipment manufacturers such as IBM and Sperry-Rand, the influence of John von Neumann on computer architecture, and the eventual spread of computing out of military labs and into academia and businesses. The development of the personal computer in the late 1970s and early 1980s quickly leads into histories of the internet and the emergence of the information economy.[14]

---

[14] A representative sample of this genre includes Janet Abbate, Inventing the Internet (Cambridge: the MIT Press, 1999), William Aspray, From Mathematical Constructivity to Computer Science: Alan Turing, John Von Neumann, and the Origins of Computer Science in Mathematical Logic (Ph.D Diss, University of Wisconsin, 1980), William Aspray, John Von Neumann and the Origins of Modern Computing, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge: The MIT Press, 1990), Thierry Bardini, Bootstrapping: Douglas Engelbart, Coevolution, and the Origins of Personal Computing (Stanford: Stanford University Press, 2000), Claude Baum, The System Builders: The Story of Sdc (Santa Monica: System Development Corporation, 1981), James Chposky and Ted Leonsis, Blue Magic: The People, Power, and Politics Behind the Ibm Personal Computer (New York: Facts on File, 1988), I. Bernard Cohen, Howard Aiken: Portrait of a Computer Pioneer (Cambridge, MA: MIT Press, 1999), Herman H. Goldstine, The Computer from Pascal to Von Neumann (Princeton: Princeton University Press, 1972), Michael Hiltzik, Dealers of Lightning: Xerox Parc and the Dawn of the Computer Age (New York: HarperCollins, Inc., 1999), David E. Lundstrom, A Few Good Men from Univac, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge: The MIT Press, 1987), Emerson W. Pugh, Building

Software, in contrast, has its roots in mathematics and is often seen as the instantiation of particular ideas about information, symbols, and logic. Its chroniclers tend to be computer scientists, cognitive scientists, and historians interested in tracing the intellectual threads of software. From Plato's queries into the foundations of knowledge to Leibniz's rationalism, to Boole's Laws of Thought in the nineteenth century, this genre seeks continuities leading to the mathematical expression of the essentially philosophical issues of perception, cognition, intelligence, knowledge, and communication. Moving into the twentieth century, the studies focus on the contributions of British mathematician Alan Turing, Norbert Weiner's cybernetic theories, Claude Shannon's theory of information, the proposition of neural nets by McCulloch and Pitts, von Nuemann's direct comparison of computers to the brain, and the beginnings of research in artificial intelligence. A general thread in many of these histories is the gradual convergence of cognition and computers that lead to new insights in such diverse fields as psychology, linguistics, and neuroscience.[15] Uniting these traditional histories are what Edwards identifies as the complementary tropes of "progress" and "revolution." In both cases,

---

Ibm: Shaping and Industry and Its Technology, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge: The MIT Press, 1995), Emerson W. Pugh and William Aspray, "Creating the Computer Industry," IEEE Annals of the History of Computing 18, no. 2 (1996), Kent C. Redmond and Thomas M. Smith, Project Whirlwind: The History of a Pioneer Computer (Beford: Digital Equipment Corporation, 1980), Raul Rojas and Ulf Hashagen, eds., The First Computers-- History and Architectures, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge, MA: MIT Press, 2000), Alex Roland and Philip Shiman, Strategic Computing: Darpa and the Quest for Machine Intelligence, 1983-1993 (Cambridge: The MIT Press, 2002), Nancy Stern, From Eniac to Univac: An Appraisal of the Eckert-Mauchly Computers (Bedford: Digital Equipment Corporation, 1981).

[15] A representative sample of this genre includes James Baily, After Thought: The Computer Challenge to Human Intelligence (New York: Basic Books, 1996), Jay David Bolter, Turing's Man: Western Culture in the Computer Age (Chapel Hill, NC: University of North Carolina Press, 1984), Daniel Creview, Ai: The Tumultuous History of the Search for Artificial Intelligence (New York: Basic Books, 1993), John Haugeland, Artificial Intelligence: The Very Idea (Cambridge, MA: MIT Press, 1985), Donald MacKenzie, Mechanizing Proof: Computing, Risk, and Trust, ed. Weibe E. Bijker, W. Bernard Carlson, and Trevor Pinch, Inside Technology (Cambridge: The MIT Press, 2001), Pamela McCorduck, Machines Who Think (New York: W.H. Freeman, 1979), Howard Rheingold, Tools for Thought: The People and Ideas Behind the Next Computer Revolution (New York: Simon & Schuster, 1985), Frank Rose, Into the Heart of the Mind (New York: Harper & Row, 1984).

steady progress in hardware and software created the material basis for a computer revolution and the emergence of an information society. Edwards is justifiably critical of these histories, stating:

> The tropes and plotlines of both genres impose requirements that lead authors to ignore or downplay phenomena outside the laboratory and the mind of the scientist or engineer. . . There is little place in such accounts for the influence of ideologies, intersections with popular culture, or political power. Stories based on the tropes of progress and revolution are often compatible with these more contingent forms of history.[16]

In response to these weaknesses, Edwards seeks to reintroduce contingency into the history of computing. Change is tied to choice, both political and social. The result is a new take on the history of computing in which the technology is "[rendered]…as a product of complex interactions among scientists and engineers, funding agencies, government policies, ideologies, and cultural frames."[17] I seek to make a contribution along these lines. However, where Edwards seeks to locate how these above mentioned complex interactions shaped the technology, I attempt to address how they gave rise to the science of simulation.

One of the difficulties in writing computer history is that it deals with complex technological systems with multifarious roots.[18] Although modern computers were initially developed for the military beginning in World War II, and were considered to be primarily of interest to scientists and engineers, by the mid 1950s, they increasingly were adopted by large corporations and applied to business problems.[19] As computing evolved and spread within different contexts, new configurations of people and machines emerged, driven by particular and perceived needs of their users. To examine any one of

---

[16] Edwards, The Closed World: Computers and the Politics of Discourse in Cold War America, p. xii.
[17] Ibid., p. xiii.
[18] Say something about the different roots. For a thorough examination of "technological systems" see Thomas P. Hughes, Networks of Power: Electrification in Western Society, 1880-1930 (Baltimore: Johns Hopkins University Press, 1983).
[19] Paul E. Ceruzzi, A History of Modern Computing (Cambridge: The MIT Press, 1999).

these contexts generally necessitates excluding others. Even within a particular context -- say commercial computing – the ways in which computers were utilized and systems evolved varies tremendously depending upon the size or nature of the industries.[20]

Histories of scientific computing suffer from the same problems. Whether done within a company, at a university, or at a national laboratory, scientific computing was beset by issues that were in some ways unique to that environment. At the same time however, there are threads running through the history of scientific computing that transcend particular contexts. Several of these threads involve issues of the professionalization of computer scientists and their relationships (both professional and epistemological) with other researchers.[21]

Established in 1946, Argonne National Laboratory provides an ideal context in which to explore these topics. As part of the national laboratory system created by the Atomic Energy Commission, it was directed to carry out large-scale, multi-disciplinary research projects related to atomic energy. Historian Peter Westwick has documented the centrality of these labs to the landscape of postwar American science as evidenced by both their level of funding and breadth of research. In 1958 alone, the six AEC multipurpose labs spent some $50 million on basic research in the physical sciences--

[20] James W. Cortada, "Commercial Applications of the Digital Computer in American Corporations, 1945-1995," IEEE Annals of the History of Computing 18, no. 2 (1996), JoAnne Yates, "Co-Evolution of Information-Processing Technology and Use: Interaction between the Life Insurance and Tabulating Industries," Business History Review 68, no. 1 (1993).

[21] Historians of science Nathan Ensminger and Thomas Haigh have attempted to address some of the issues of professionalization as they relate to computer programmers within industry. See Nathan L. Ensmenger, From "Black Art" to Industrial Discipline: The Software Crisis and the Management of Programmers (Ph.D. diss., University of Pennsylvania, 2001), Thomas Haigh, Technology, Information and Power: Managerial Technicians in Corporate America, 1917-2000 (Ph.D. thesis, University of Pennsylvania, 2001).

half as much as all academic institutions combined.[22] In the life sciences, the labs spent about one tenth (or $13 million) of what colleges and universities spent, but in certain areas such as genetics, the AEC supported one third of all research, with the vast majority of work being conducted at the national labs. As Westwick points out, the $206.3 million spent on research and development at the national labs in 1958 "far outpaced the R&D commitment of the largest industrial corporations."[23] By the mid-1960s, the national lab system had spent $4 billion on research and a comparable figure on building facilities. The substantial financial support provided to the AEC labs allowed them to influence the direction of research in many fields, including high-energy physics, solid-state physics and materials science, nuclear medicine, and radiobiology.[24]

In addition to the design of reactors and atomic weapons, the AEC also influenced the direction of research through the distribution of low-cost radioisotopes to physicists, chemists, metallurgists, biologists, meteorologists, and health physicists at universities, in industry, and at other national labs. Because computers were well-suited to modeling stochastic phenomena such as radiation, they became an increasingly important tool for a broad range of scientists investigating, for example, the effects of radioisotopes on biological, environmental, and physical systems. The use of computers for atomic research was further encouraged by the AEC's willingness to fund the construction of computers at labs such as Argonne and Los Alamos, and then later to grant contracts to the budding computer industry to build ever more powerful machines.

---

[22] Peter J. Westwick, The National Labs: Science in an American System, 1947-1974 (Cambridge: Harvard University Press, 2003).
[23] Ibid.
[24] Ibid.

Chapter 1, “Building Big Iron,” is primarily a narrative that describes Argonne’s early initiatives in computing. The Lab’s first reactor development project was for Admiral Hyman Rickover of the Navy, a science-trained man who was interested in nuclear propulsion for submarines. As part of this effort, the lab began to develop extensive computing capabilities to assist the engineers who were designing this system. Although there were initially two computing groups at Argonne -- – one in reactor engineering and the other in physics – the two groups actively collaborated on computational problems applicable to both.[25] Beginning in 1949, the computing groups initiated a program to build a digital computer for use in reactor engineering and physics problems that would be similar to a machine being constructed by Princeton University’s Institute for Advanced Study.[26] Building on the success of the AVIDAC (Argonne’s Version of the Princeton Institute’s Digital Automatic Computer), the computing section continued to build experimental computing equipment-- and then look for ways to incorporate it into scientific activities around the Lab.

Almost immediately, the computing needs of Argonne scientists outstripped the capacity of these computers. The vastly increased demand for computing coincided with the emergence of a commercial computer market. In late 1957 the computing section at Argonne acquired its first commercial machine, an IBM 704 (the twenty-fifth delivered), and this effectively ended their activities in large-scale computer construction. The move to commercial computers also coincided with the consolidation of the two computing sections into a single Applied Mathematics Division (AMD); both changes reflected

[25] Interview with Margaret Butler by author 11-27-2001, Argonne National Laboratory.
[26] Aspray, John Von Neumann.

efforts by the Lab to improve computational service and apply computers to a range of problems that were shared by a number of divisions at Argonne.[27]

Chapter 2 “Applied Mathematics, ‘Hybrid Areas’ and the Social Organization of Computational Science,” addresses the creation of the Applied Mathematics Division and the visions that informed its operation. While the advancement of digital computer technology is a significant part of this story in that it allowed more work to be done on computers, more important were the AMD’s efforts to integrate computers – <u>and</u> their attendant personnel – into the scientific process. Early on, the directors of the AMD envisioned a new role for applied mathematicians vis-à-vis scientists and engineers in the development of mathematical models suitable for digital computers. In particular, it was believed that "computational science" required that applied mathematicians be incorporated more directly in all stages of scientific and engineering practices -- from problem formulation to the definition of what constituted a solution. Arguments in favor of such a collaborative structure drew on Cold War rhetoric, debates within the mathematical profession, and issues surrounding the increasing quantification of the sciences. Here, historian Michael Mahoney’s conception of a disciplinary *agenda* provides a useful analytic tool for analyzing the activities of the AMD. Mahoney defines a disciplinary agenda as: 1) what practitioners of a discipline agree ought to be done; 2) a consensus concerning the problems of the field; 3) their order of importance or priority; 4) the means of solving them; and most importantly, 5) what constitutes solutions.[28]

---

[27] See Preface to first Summary Report, H.R. Crane, to L.R. Kimpton. "Review Committee Report," June 1, 1958. Box 11, RG 326: Records of the Policy Advisory Board, 1957-1967, GLFRCNARA.
[28] Michael S. Mahoney, "Computer Science: The Search for Mathematical Theory," in <u>Science in the Twentieth Century</u>, ed. John Krige and Dominique Pestre (Amsterdam B.V.: Harwood Academic Publishers, 1997).

Lacking a disciplinary agenda of its own, but staffed with people attuned to the computational needs within a variety of different fields, AMD researchers initially relied on physicists, chemists, biologists, and metallurgists to define the areas for which theoretical work in machine computation needed to be done. In essence, the fundamental research priorities for the Applied Math Division emerged from its work in helping other disciplines conduct research. Over time however, applied mathematicians and nascent computer scientists began to stake out a new, shared research agenda that balanced their service requirement to scientists and engineers with the necessity to conduct research in the foundations of mathematics and the theory of computing. I argue that their efforts at self-fashioning are reflected in a series of organizational changes made within the AMD which attempted to institutionalize the disciplinary independence of computer science.

Finally, I contend that as computer science began to emerge as a distinct discipline in the mid-1960s, it erected professional barriers that impeded the creation of the kinds of collaborative research projects that had been envisioned earlier in the decade.

Chapter 3 "Computing at the Edge: Grand Challenges and the New Hybrid Area," addresses efforts by computer scientists to define further the intellectual framework for their discipline or face scientific semi-irrelevance. Unable to attract long-term support for theoretical research, practitioners increasingly began to emphasize the experimental component of their work. This reorientation was significant for several reasons. Epistemologically, the identification of theoretical and experimental practices within computer science was offered as justification for why it should be considered a "science." Secondly, experiments required equipment. Just like physicists and their colliders, computer scientists claimed that they too needed expensive experimental equipment. Perhaps most importantly, computer science experiments promised to make a contribution to the bottom line. By the late 1970s, practitioners could point to widely-heralded successes in the development of time-sharing computers and the creation of the ARPANET as instances of experimental research making a direct contribution to scientific work. Although computer scientists still had to defend their theoretical projects, in general, the reorientation in how the scientific work of the discipline was presented to funding agencies was effective, and by the end of the decade practitioners could agree that the "crisis" was easing.

The chapter then shifts focus to address the impact of the Japanese Fifth Generation project on national policy, computer science research, and computational science. The central goal of this narrative is to flesh out the characteristics of computational science and explain its subsequent endorsement in the 1991 federal High Performance Computing and Communications program.

Chapter 4," The Computational Machine" is a preliminary investigation of the effects of the High-Performance Computing and Communications program on computer science research. I argue that computational science produced its own ideology, its own way of conceptualizing how science should be done, and it was this ideology that found purchase in federal funding agencies. To its sponsors, computational science promised to connect and coordinate technologies, people, and disciplines into a well-oiled computational machine that could solve problems deemed significant to its sponsors. As a result, federal funding agencies like the DoE were willing to reorient computer science research in order to support the needs of computational scientists. Computer scientists, in turn, were forced to choose between aligning their research with the goals of the computational science machine or face a sever decline in funding. Some of the implications of this choice are addressed in the conclusion.

A Note on Sources:

One of the more difficult tasks involved in framing the human dimensions of scientific computing is determining the relevant interest groups. Before computational science was a recognized discipline, computer specialists were scientists trained in another discipline. Physicists in particular, seemed to be drawn into computing early, but they were quickly joined by chemists, engineers, mathematicians, biologists, linguists, and even philosophers. My effort to discern relevant interest groups at Argonne is made more difficult by the general lack of quality archival material about the AMD from this time period. Responding to a Reduction in Paperwork Act under the Johnson Administration, the AMD purged most of its internal documents that would have been invaluable for this study. Furthermore, the Mathematics and Computer Science Division – the most recent incarnation of the AMD at Argonne – maintains only eleven boxes of documents in their archives, all of which focus on the period 1988-1995.

This dearth of direct archival evidence does not mean that conclusions cannot be drawn from what remains. Many of my sources are from files donated by several of Argonne's Laboratory Directors to the Great Lakes Federal Record Center National Archive in Chicago. Housed in a cinderblock building on Chicago's Southside, the documents related to the AMD tended to be correspondence with people outside the laboratory that had been cc'd to Argonne's director. In many cases, the letters referred to juicy attachments that, alas, were not included in the archives. Finally, personal files donated to me by key individuals involved in the history of the AMD have provided insights not otherwise apparent in officially archived sources.

Every historian dreams of finding a complete and untouched archival record for their subject. My experience is no doubt much closer to the norm. At the same time, there is enough material between the Laboratory Director's files and Annual Reports of the AMD, records of the Review Committee for ANL (housed in the special collections archive of the University of Illinois, Urbana-Champaign), records of the Atomic Energy Commission in the National Archives in Washington, D.C., the Department of Energy archives in Germantown, MD, and personal interviews to suggest strongly my following analysis.

Chapter 1

Building "Big Iron"

"A microprogrammable computer on a chip!" proclaimed an advertisement in the November 15, 1971 issue of Electronic News.[1] And indeed the Intel 4004 to which this advertisement referred was a technological marvel worthy of exclamation. Engineers at Intel had succeeded in fashioning a thin silicon wafer that contained all the basic registers, arithmetic unit, input/output capabilities, and control mechanisms of a general-purpose, stored-program computer. This new "microprocessor" became the crucial component necessary to making personal computers, and by extension, making computing personal. Thanks to increasing miniaturization, a typical desktop personal computer of today weighs about twenty pounds, is housed in a durable plastic case, and has computing power many orders of magnitude greater than the huge mainframes of the 1950s and 1960s. Outdated PCs are regularly sold at garage sales or even donated. It is safe to say that, aside from the environmental damage they cause in landfills, computers today are disposable.

[1] Cited in Paul E. Ceruzzi, A History of Modern Computing (Cambridge: The MIT Press, 1999), p. 200.

The personal computer stands in sharp contrast to the early days of electronic computing. Just two decades earlier, the same components found on the microprocessor would have filled a large room and would have necessitated industrial-strength air-conditioning to keep them cool. A far cry from the portable computers of today, these early computers were huge, hot devices that required a team of operators and maintenance staff to keep them functioning. In a moment of reflection, one computer scientists recently referred to this computing environment as the days of "Big Iron."[2]

Reminiscent of the days of steam engines and the iron horse, "Big Iron" seems especially appropriate when one considers the sheer size of the computers built from the 1950s to the mid-1960s. With their thousands of vacuum tubes, miles of wire, auxiliary units for input, output, and storage, air conditioning requirements, and weight sometimes in excess of several tons, "Big Iron" captures both the proportions and the character of these calculating machines.

This chapter, "Building Big Iron" traces the efforts of engineers and scientists at Argonne to design and build their first electronic computers. In particular, I discuss how Argonne's mission under the Atomic Energy Commission to develop nuclear reactors was the catalyst for building a computing program at the Laboratory. Faced with an ever-increasing computational load related to the design of nuclear reactors and research in the nuclear sciences, it became clear that faster engines of computation would be desirable. Several members of the Argonne staff, attuned to the emerging field of electronic digital computers after World War II, lobbied for and received authorization from the AEC to design and build a computer for the laboratory. Between 1949 and 1963, researchers at Argonne produced four large-scale electronic computers which were used as both objects

[2] Interview with Dr. Mike Minkoff, Math and Computer Science Division, Argonne National Laboratory, 1-11-2001.

for, and objects of, scientific inquiry. In 1957, the computer construction activities were curtailed as Argonne began to use commercially produced computers from IBM and later Control Data Corporation. The decision to acquire commercial computers was not just economic; there was also a growing recognition among computer specialists at Argonne that there were certain advantages to be gained by having a computer that was widely compatible with other computers at other laboratories. Reflecting the hybrid nature of computing as consisting of both hardware and software, the following chapter, “Applied Mathematics, ‘Hybrid Areas’ and the Social Organization of Computational Science,” examines the “software” side of the AMD. While the theme of “Big Iron” is still appropriate, instead of looking at computer technology, I instead examine the extensive human organization that was created to manage these computational resources and how this group interacted with scientists and engineers at Argonne.

From the Met Laboratory to Argonne National Laboratory, 1942-1948

By the time Argonne was designated the nation’s first national laboratory on July 1, 1946 it had already played an important role in the research, design, and development of nuclear technologies, including the design and construction of both power and production reactors. The name “Argonne” comes from a thinly populated forest preserve about twenty-five miles southwest of Chicago. Its relative isolation from Chicago made it seem an ideal location for researchers from the University of Chicago to conduct experiments on nuclear fission. As part of an effort to expand atomic energy research at the University of Chicago in early 1942, Arthur Holly Compton, the chairman of the physics department and a Nobel Laureate, had succeeded in concentrating chain reaction studies being conducted at over a dozen universities within his so-called “metallurgical project" at the University of Chicago. By the end of the year the Metallurgical Laboratory, as it was called in order to disguise its true purpose, had a staff of over four hundred and had became one of the government's largest scientific operations. Its administrators coordinated uranium chain-reaction research at Chicago, Berkeley, Columbia, the University of Minnesota, Princeton, and the National Bureau of Standards in Washington, D.C.[3] The goal of this group was to demonstrate the feasibility of a sustained chain reaction using uranium oxide pellets to produce plutonium in quantities sufficient for military use. If this could be established, as theorists predicted, it would be a major step toward demonstrating the feasibility of building an atomic weapon.[4]

Under the direction of Italian émigré physicist Enrico Fermi, engineers, physicists, and chemists worked feverishly to build the first atomic pile. Scientists at the Met Lab, concerned about the health hazards associated with radioactive materials, planned to assemble their first pile, CP-1 (Chicago Pile 1), at the Argonne site where it would be isolated from civilian populations.[5] However, war time pressures cast the entire Met Lab project into jeopardy and, as labor problems slowed the preparation of the Argonne site, Compton ordered CP-1 to be constructed in the squash courts underneath Stagg Field on the University of Chicago campus. After some intense calculating, Fermi was confident that he could build CP-1 with low power so that the chain reaction could be demonstrated without endangering the city. Given the go-ahead by Compton, Fermi's crews began working night and day from November 16 to December 1, 1942 to assemble

---

[3] Jack M. Holl, Argonne National Laboratory, 1946-96 (Chicago: University of Illinois Press, 1997), pp. 6-8.

[4] Richard G. Hewlett and Jr. Oscar Anderson, The New World Order, 1939-1946, Volume I of a History of the United States Atomic Energy Commission (University Park: Pennsylvania State University Press, 1962), pp. 109-10.

[5] This concern about the harmful effects of radiation was not entirely about protecting the civilian population. The scientists involved in the Met Lab were concerned with their own health. Consequently, the Met Lab established a health physics laboratory to monitor exposure rates of workers and to conduct their own studies on the effects of radiation exposure on animals and plants. See Holl, Argonne National Laboratory, pp. 11-12.

45,000 graphite bricks drilled with 19,000 holes filled with uranium oxide pellets into an ellipsoidal-like lattice pile 20 feet high, 25 feet across at the equator and 6 feet wide at each pole. In total, 400 tons of graphite, 6 tons of uranium metal, and 50 tons of uranium oxide went into the construction of CP-1.[6] As part of the building process, the laboratory instruments group had to develop and build special circuits and equipment to monitor the pile, including fashioning cadmium control rods and a principle safety rod, ZIP, which would be inserted into the pile should the radiation levels become too high.[7]

On December 2, at 9:45 a.m. the process of bringing the pile to criticality began as Fermi ordered all but one of the control rods removed. Inch by inch, the last rod was withdrawn by hand as Fermi monitored the neutron readings and repeatedly calculated the multiplication factor on his slide rule, even jotting down some figures on its ivory back. As the pile slowly approached criticality, each measurement matched those predicted by Fermi's theoretical calculations.[8] So sure was the physicist, that he even had the rods replaced so that the crew could enjoy a leisurely lunch at about 11:30. Returning to the experiment at 2:00, Fermi once again had the rods withdrawn until at 3:00 p.m., the last rod was removed. Within 20 minutes, the Geiger counters began clicking away, indicating the beginning of a chain reaction. By 3:42, as one onlooker reported, the pile was "really cookin." For the forty-two observers, the rapidly escalating clickety-clack of the counters increased the tension in the room and all eyes turned towards Fermi for signs that he would end it. Finally, with the words "Zip in" Fermi called a halt to the

[6] Hewlett and Oscar Anderson, The New World Order, p. 112.
[7] Holl, Argonne National Laboratory, p. 16.
[8] Corbin Allardice and Edward R. Trapnell. The First Reactor. Washington, D.C.: United State Department of Energy, 1982.

successful experiment and the intensity of the pile abruptly dropped below critical levels.[9]

The success of CP-1 demonstrated clearly to the military that a controlled chain reaction was feasible and opened the possibility that a reactor could be used to produce plutonium for a bomb. The results of the chain reaction experiments became the basis for the full-sized plutonium production reactors that were, after intense negotiations, to be built at Clinton, Tennessee rather than at the Argonne site due to Clinton's isolation from population centers. With the production piles to be built elsewhere, the Met Lab dismantled CP-1 beginning in February, 1943 and reassembled it in a modified form at the new Argonne site. As the Manhattan Project shifted into high gear, CP-2 (Chicago Pile 2), as it was now called, served as an experiment station to provide information for the production piles at Clinton and Hanford, Washington. This included reviews of reactor designs, development of improved methods for separating plutonium, research into radiation safety, and waste disposal.[10]

Because the Met Lab had produced the first chain reaction, it seemed natural that it would continue to be a leader in the research and development of nuclear reactors. However, the transfer of leading researchers to the Clinton and Hanford facilities called into question whether Argonne, as the proposed successor to the Met Lab, could continue in this capacity. Undaunted, Walter H. Zinn, Fermi's deputy at the yet-unfinished Argonne Laboratory received authorization from the military to build a heavy water reactor at the new site, and by working around the clock beginning New Year's Day 1944, CP-3 (Chicago Pile 3) reached criticality on May 15, 1944. Despite CP-3's success as an experimental reactor, the exodus of talent to Los Alamos and Hanford

---

[9] Holl, Argonne National Laboratory, pp. 18-19.
[10] Ibid., pp. 20-7.

continued, leaving Argonne with a skeleton crew during the summer of 1944. As the physics and chemistry divisions were gutted, the only Met Lab division to see an increase in work was the health division, which grew from 173 employees in November 1944 to 196 in March 1945.[11]

With the war over in August 1945, deciding the fate of the Met Lab and its Argonne campus became even more imperative. In early 1946, the University of Chicago agreed that Argonne, now conceived as a regional laboratory, would absorb the programs, buildings, equipment, and staff from the Metallurgical Laboratory when the latter ended operations in the middle of the year. A second advisory committee, which included Compton and six other scientists from the Manhattan Project, recommended that Argonne instead be constituted as a *national* laboratory to conduct unclassified basic research requiring the "use of piles and other expensive large-scale equipment."[12] With the army's approval for this plan, Zinn quickly received permission to build a fast-fission reactor at Argonne, provided that it could be done safely at the site.

As a general policy for the national laboratory system began to emerge in 1946, and as the United States government debated the best ways to manage and control nuclear energy research and development, Argonne administrators struggled to integrate the Met Lab legacy into a set of organized and coherent research programs. The principal mission of Argonne was to carry out basic research in nuclear reactors, including their design and development. This mission included research in reactor chemistry, health physics and radiobiology. Overlooked, but important to the future of Argonne programs, the lab inherited a high-caliber instrument section that had achieved some renown during the war for its design and manufacture of precision equipment for the Manhattan Project.

[11] Ibid., pp. 28-32.
[12] Ibid., p. 40.

With the return to peacetime status, the production activities of the instrument section were curtailed in order to let private industry enter the market, but it continued to operate on a research and development basis, well-integrated into the overall program at Argonne.[13]

Appointed as the first Director of Argonne National Laboratory, Walter Zinn and his deputy, Norman Hilberry who had been Compton's assistant at the Met Lab, presided over a research institution responsible for research in the physical and biomedical sciences and the development of nuclear reactors. Given wide latitude to organize the laboratory as he saw fit, Zinn divided it into two groups: basic research and reactor development. The former group, which included physics, chemistry, biology, mass spectroscopy and X ray, instrument research, information, patent, and medical and hazard evaluation, he placed under the direction of Hilberry. Zinn maintained his position directing the basic research in reactors, while the actual construction of piles, metallurgical research and instrument fabrication was assigned to Harvard Hull, who had been an administrator at the Clinton works during the war.[14]

Throughout 1947, Argonne's work in reactor development remained in a holding pattern as advisory committees at the newly created Atomic Energy Commission (AEC) tried to formulate policies to govern research and development in the nuclear sciences. Zinn, recognized as the leading expert on reactors, stressed to the committees the need to develop a variety of reactors to suit different purposes. While the weapons programs needed reactors to produce plutonium, others would be needed as radiation sources for the testing of materials and producing radioisotopes for basic research. In addition, Zinn

[13] Ibid., pp. 42-6, F.R. Shonka, to N. Hilberry. "Objectives of Instrument Research and Development Division," Feb. 15, 1949. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.
[14] Holl, Argonne National Laboratory, p. 51.

carefully explained to the AEC committee some of the critical issues that had to be solved before nuclear reactors could be used for the production of electricity. Foremost among these, according to him, was the scarcity of fissionable material. The current stocks of uranium ore were barely sufficient for the production of a small number of nuclear weapons, let alone fueling power reactors. Thus, Zinn proposed, among other designs, the development of "breeder" reactors, a type of reactor that produces more fissionable material than it consumes, and suggested that Argonne was well suited to carry out this research.[15] Although in principal breeder reactors were feasible, Zinn noted that the factors to be considered in order to actually build one were extremely complicated. His experience building CP-3 several years earlier made it clear that piles could no longer be built on a trial and error basis as had the first three Chicago reactors.[16]

As 1947 drew to a close, the Atomic Energy Commission recommended that the agency's reactor development program be consolidated at Argonne and proposed an overall plan that followed Zinn's suggestions closely. This decision caught Zinn off guard as he had not expected nor desired that Argonne be designated the center for reactor design. At stake, he felt, were the other research programs at the lab which would inevitably have to take a back seat to the reactor program. Making matters worse, as 1948 began, the facilities at Argonne were still incomplete and many of the research units were housed in temporary Quonset huts. Although Zinn successfully resisted the AEC's efforts to centralize reactor design at Argonne, he was forced to accept responsibility for the design of a reactor for naval submarines.

In the spring of 1948, Captain Hyman G. Rickover, an engineering officer who had coordinated a naval study on the prospects of submarine nuclear propulsion system,

---

[15] Ibid., pp. 59-61.
[16] Ibid., p. 28.

arrived at Argonne to initiate a program to develop a suitable reactor. The advantages of nuclear propulsion over the diesel/electric systems for submarines was obvious; they would be faster, quieter, and able to remain below the surface for a much longer time. For the past year, Rickover had been working with a science and engineering group at Clinton, TN conducting theoretical studies for his propulsion system and they were now ready to move toward engineering research and development. With the scheduled transfer of the Clinton group to Argonne in summer 1948, Rickover put pressure on Zinn to give the project high priority.

The Increasing Computational Load

With the design of the fast breeder reactor already underway, the naval propulsion project suddenly placed tremendous pressure on the computing group at Argonne who supported the reactor engineering division. In particular, the reactor computing group needed to increase their capabilities to produce mathematical support for the engineering projects. In the short term, the reactor computing group hired mathematicians below the PhD level to do hand calculations, but this was only a temporary measure.[17] Similarly, a second computing group located within the Physics Division was also being pushed to its limits as the mathematics involved in nuclear science research nearly overwhelmed its capabilities to provide results quickly. Thus, by early 1949, it was evident that Argonne's computational resources would need to be expanded for the Laboratory to fulfill its mission to the AEC.[18]

[17] Henry Tropp. Interview with Margaret Butler, Jim Butler, Dave Jacobsohn, Charles Harrison, Claire Kilty, Burt Garbow, Stan Zawadzki, Bob Kroupa, Franz Morehouse, and Wallace Givens. 1972. box 1, COHC #196, Series 1: Transcripts, subseries B: Research Transcripts, American History Museum, Smithsonian Institute.
[18] Ibid. A similar situation arose during WWII at Los Alamos when the computational needs of the Implosion group outstripped the capacity of desk calculators to complete the required calculations quickly.

Initially, the two computing groups were able to increase their production by employing punched-card tabulating machines, a practice that had first been applied to scientific calculations by L. J. Comrie, superintendent of the Nautical Almanac office, Greenwich, England in 1929. Comrie was the first to adapt commercial accounting machines to scientific purposes, but his techniques were soon adopted by Wallace J. Eckert, an astronomer at Columbia University who applied them his own studies. When Eckert became the director of the Scientific Computing Bureau at Columbia in 1934, IBM donated punched-card equipment to assist the bureau in their investigations of harmonic analysis and the integration of differential equations. Renamed the Thomas J. Watson Astronomical Computing Bureau to reflect IBM's contributions, the bureau became a testing ground for new applications of punched-card equipment for scientific and engineering calculations. Eckert himself became one of the chief architects of IBM's first sequence-controlled calculator, an electronic machine capable of performing a calculation with up to fifty arithmetic steps, and later he organized the design and construction of IBM's Selective Sequence Electronic Calculator (SSEC) built in 1948.[19]

At Argonne, several members of the computing groups had extensive experience applying punched-card equipment to scientific problems. Foremost among these was Donald (Moll) Flanders, a mathematician who had worked at Los Alamos during the war organizing a computing section with hand calculators (generally Friedens) and tabulating equipment.[20] By training Flanders considered himself an applied mathematician and was

---

According to Aspray, this need happened to coincide with von Neumann (who was part of this group) becoming aware of the ENIAC project at the University of Pennsylvania, and his thought that this new electronic computer might solve the computational crisis at Los Alamos. See William Aspray, From Mathematical Constructivity to Computer Science: Alan Turing, John Von Neumann, and the Origins of Computer Science in Mathematical Logic (Ph.D Diss, University of Wisconsin, 1980), pp. 216-7.

[19] Martin Campbell-Kelly, "Punched-Card Machinery," in Computing before Computers, ed. William Aspray (Ames: Iowa State University Press, 1990), pp. 147-50.

[20] William Aspray, John Von Neumann and the Origins of Modern Computing, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge: The MIT Press, 1990), p. 29.

interested in finding numerical solutions to problems originating within the physical sciences and engineering.[21] After the war, Flanders was hired as a mathematician by the Met Lab to organize a computing group at the laboratory. With the Argonne facilities in transition, for several years he and the staff ran computations on the tabulating department machines at the Museum of Science and Industry in Chicago which was where Argonne's administrative offices were located. However, this was a stop-gap measure and eventually Argonne was able to acquire two IBM 602s, one IBM 604 calculating machine, and even two IBM Card Programmed Calculators (CPC) or "combo" units as they were called.[22] While the combo units were programmable to a certain extent, the lab had tremendous difficulty setting them up and getting their two units to operate correctly.[23]

Even with the dual CPCs functioning well, it was evident that the computational desires of the nuclear scientists and engineers were far greater than what these machines could provide. For example, the Opacity Group at Argonne, which was investigating the opacity of various materials for Los Alamos and the Reactor Engineering Group, kept one of the two CPC units busy from December 1950 through the middle of 1952.[24] Such a monopoly on machine time by one group necessarily limited the computational resources available to other scientists at Argonne.

[21] Constance Reid, Hilbert-Courant (New York: Springer-Verlag, 1986), p. 388.
[22] Donald A. Flanders, to Jean Hall. June 21, 1951. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA. For more about the development of the Card Programmed Calculator, see Charles J. Bashe and others, Ibm's Early Computers (Cambridge: The MIT Press, 1986), pp. 68-72.
[23] Donald A. Flanders, to George Shortley. Nov. 10, 1950. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.
[24] R.E. Meyerott, to Louis A. Turner. "Estimate of Computing Required by Opacity Group," Feb. 26, 1951. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA. Opacity is another way to measure the reactivity of materials (?)

With electro-mechanical techniques inadequate, in 1948 Frank C. Hoyt, the head of the Physics Division, suggested to Zinn that Argonne consider building its own digital electronic computer.[25] Hoyt, Dave Jacobson, and Jim Alexander, two other Argonne employees, had run several complex Monte Carlo simulations for the physics group on the ENIAC after the war and were impressed by the unprecedented speed and flexibility of this new calculating engine.[26] The ENIAC, the first general-purpose, programmable electronic digital computer was commissioned by Army Ordnance in June 1943, which wanted to increase computing power at the Ballistics Research Laboratory (BRL). As discussed earlier, it was designed and built by an engineering team at the University of Pennsylvania's Moore School of Electrical Engineering.[27] It was a huge device: with its forty panels, 1500 electromechanical relays and 17,000, 8-inch vacuum tubes, it filled a large room. More than twenty different units handled addition, subtraction, multiplication, division, and square roots, while input/output was handled by IBM punched-card equipment. Although programming was labor-intensive, requiring the setting of hundreds of switches and plugging numerous cables, by working at electronic speeds the ENIAC was hundreds of times faster than any other calculating machine. Two, ten-digit decimal numbers could be multiplied in less than 1/300 of a second.[28]

---

[25] Holl, Argonne National Laboratory, pp. 76-7.

[26] Donald A. Flanders, to J.W. Givens. Nov. 29, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA. The Monte Carlo method was borrowed by physicists from statisticians. It is a method for studying mathematical or physical problems by stochastic processes. A probabilistic model is constructed, corresponding to the mathematical or physical problem, and random samples are then taken within the model. By taking more samples, a more accurate estimate of the result can be obtained. This technique was especially useful for scientists trying to determine the likelihood that a neutron would react with a given material to produce fission.

[27] ENIAC stands for Electronic Numerical Integrator and Computer. Nancy Stern, From Eniac to Univac: An Appraisal of the Eckert-Mauchly Computers (Bedford: Digital Equipment Corporation, 1981). For more on the ENIAC, see Herman H. Goldstine, The Computer from Pascal to Von Neumann (Princeton: Princeton University Press, 1972).

[28] Aspray, John Von Neumann, pp. 34-5.

Hoyt was interested in this new field of electronic computing and argued, in part, that building a computer at Argonne would fit within its mission by providing special purpose equipment that universities could not afford. In addition, the interdisciplinary nature of a computer project was in itself appealing as engineers, chemists, physicists, and mathematicians at Argonne were all interested in the design of computing equipment.[29] From a mathematical standpoint Flanders had already begun to develop ideas for how an electronic computer might be organized from the point of view of the logical structure. During the war he had produced several long papers on binary arithmetic even before the ENIAC project began, but because they were all agency reports, they remained unpublished. When it appeared that, for a variety of technical reasons, binary rather than decimal representation of numbers would become the basis for computer arithmetic, Flanders saw the computer project as a way to develop further his earlier work while still remaining within the AEC mission.[30] Moreover, from his experiences directing the hand computing section at Los Alamos he was keenly aware of how specific computational problems within the nuclear sciences might affect the design of an electronic computer.

Despite the dire need for improved computational capabilities, plus the interest and expertise already at Argonne, it is not clear that approval for such a project would have been given in 1949 had there not been another computer project at Princeton already underway. Back in 1944, with the ENIAC project forging ahead, John von Neumann

[29] Tropp. Interview with Margaret Butler, Et Al.
[30] Aspray, John Von Neumann, p. 66. Binary representation of numbers was favored by von Neumann and Goldstine primarily because of the binary nature of the early computer memory units – in this case mercury delay lines. In addition, binary representation could be implemented with fewer components, allowed faster execution of the basic operations, and worked well with the computer's logical structure. The main drawback – that binary numbers had to be converted to decimal numbers for output – was compensated for by the small amount of input/output in scientific calculations relative to the number of computations performed on the data. The higher processing speeds of the binary system more than made up for the binary-to-decimal conversions.

visited the Moore School to study the machine. Although too late to affect the design of the ENIAC, von Neumann participated extensively in discussions about a second machine that would address several perceived design deficiencies, in particular the ENIAC's inadequate memory and programming difficulties.[31]   The outcome of these discussions was von Neumann's "First Draft of a Report on the EDVAC (Electronic Discrete Variable Automatic Computer)" written in early 1945. This report described the configuration of a very high-speed automatic digital computer, meaning that once the instructions were described and feed into the computer in exhaustive detail, it would carry out these instructions without human intervention.[32] The machines also had to be automatic because the intrinsic speed of their operations was so great that human intervention was simply impossible.[33]

Perhaps the most important contribution of the EDVAC report to would-be computer designers was its description of the "logical" structure of a computer system. Von Neumann envisioned a system composed of five different units: a central arithmetic unit to handle the four basic arithmetic operations and some higher order functions such as roots, logarithms, and trigonometric functions; a control unit to ensure the proper sequence of operations and to make the different units operate together; a memory system to store instructions and data, and separate devices to handle input and output tasks. Von Neumann saw little point in providing an engineering description of this machine, and chose rather to frame his conceptions around the abstract parts and their tasks within the

[31] Ibid., pp. 35-7. For example, inputting a new program on the ENIAC required manually changing hundreds of wires on a plugboards.
[32] John von Neumann, "First Draft of a Report on the Edvac," in Papers of John Von Neumann on Computing and Computer Theory, ed. William Aspray and Arthur Burks, Charles Babbage Institute Reprint Series for the History of Computing (Cambridge: The MIT Press, 1987), p. 17. Unlike the ENIAC, where arithmetic was based on the decimal system, the EDVAC was a binary machine and would have its own high-speed memory.
[33] John von Neumann. Memorandum. 1945. John von Neumann papers, Library of Congress. Quoted in Aspray, John Von Neumann, p. 269.

overall system. Thus, von Neumann drew an analogy between the associative, sensory, and motor neurons of the human nervous system with the central processor and input/output devices of his computer.[34] Framing his report using idealized components with biological analogies had the advantage of distinguishing the logical structure of the computer from its engineering design.[35] This approach allowed designers to separate the different functions of the computer from technological limitations of components available at any given time.[36] A second advantage, and one that was not recognized until later, was that by discussing his system in neurological rather than conventional technological terms, von Neumann circumvented military security which would normally have kept his report secret.[37]

Von Neumann also devoted a large section of a second EDVAC report to the problem of programming a digital machine, such as he was proposing. As a conceptual test of his organization of computer processes, von Neumann sketched out a code for sorting non-numeric data into a nondecreasing order. In part, his choice of problem was to see if his design could handle complex processes such as sorting. But the existence of special purpose sorting equipment from IBM also provided him with a benchmark upon which to measure the proposed computer's speed.[38] In the process of writing this

---

[34] Arthur Burks, "Introduction," in Papers of John Von Neumann on Computing and Computer Theory, ed. William Aspray and Arthur Burks, Charles Babbage Institute Reprint Series for the History of Computing (Cambridge: The MIT Press, 1987), p. 7.

[35] According to William Aspray, von Neumann's insight to separate the logical structure of the computer from its engineering design was crucial to the establishment of computer science as a discipline. By drawing a distinction between computer logic and its implementation in hardware, it allowed mathematicians and scientists to study computing as an abstract concept with requisite theories. See Aspray, From Mathematical Constructivity to Computer Science.

[36] Burks, "Introduction," p. 7.

[37] Emerson W. Pugh and William Aspray, "Creating the Computer Industry," IEEE Annals of the History of Computing 18, no. 2 (1996): pp. 7-8.

[38] Donald E. Knuth, "Von Neumann's First Computer Program," in Papers of John Von Neumann on Computing and Computer Theory, ed. William Aspray and Arthur Burks, Charles Babbage Institute Reprint Series for the History of Computing (Cambridge: The MIT Press, 1987), p. 83. His program was designed to sort records into order and then merge the two strings of already sorted records into a single sorted sequence.

program, von Neumann discovered a flaw in his logical design whereby for even simple programs, the computer's processing unit would have to wait a long time to retrieve data that was used frequently. Recognizing the inherent inefficiency, von Neumann proposed what today is called the "cache" memory or buffer store. This is a type of memory that can be readily accessed by the processing unit non-sequentially rather than having to retrieve it from a much slower main memory location that can only be accessed serially (i.e. first in last out).[39] In practice, von Neumann envisioned a hierarchy of memories, each one progressively larger but slower than the previous.

After the war, von Neumann wanted to develop further his ideas about electronic computation. After finding a professional home at Princeton's Institute for Advanced Study, he began agitating for funding and authorization to build an automatic computer for scientific purposes. Von Neumann successfully overcame the institutional resistance to a computing project by November 1945, based in part on his arguments that the development of an automatic computer was a legitimate research project in itself. Once completed, he proposed a period of time during which the machine itself would be an object of scientific inquiry. After that, he was confident that the computer would lead to great advances in "hydrodynamics, aerodynamics, quantum theory of atoms and molecules, and the theory of partial differential equations in general."[40]

While this line of argument satisfied Princeton's board of directors, von Neumann offered a different argument to the Army Ordnance Department or the Navy Office of Research and Inventions, which provided needed funds to build the computer. To them,

---

[39] Burks, "Introduction," pp. 10-11.
[40] von Neumann. Memorandum. quoted in Aspray, John Von Neumann, p. 52. Other arguments from von Neumann included the "prestige" factor of having the most powerful computing device in existence at Princeton, a fact that would attract top scientists to the Institute. Interestingly, the prestige factor inherent in having the lastest/greatest computer was a contributing factor to the installation of many computer systems in businesses beginning in the mid-1950s.

the mathematician suggested that the proposed computer could be applied to militarily relevant scientific calculations. More importantly for Argonne, von Neumann advocated that once the bugs were worked out of the IAS machine, copies of it could be built for other government installations. This promise was also influential in garnering the support of the Atomic Energy Commission, which had a tremendous need for faster engines of computation to assist scientists and engineers in the atomic weapons program.[41] Both the AEC and the Navy agreed to fund the project and work began on the IAS Computer with the understanding that its blueprints would be shared with select government agencies.

Although design of the logical system began in 1946, a fully fleshed out account of the arithmetical processes did not appear until September 1947, and even then the description of input/output devices was preliminary.[42] Throughout that year and into 1948, von Neumann's group was occupied with acquiring the necessary equipment to carry out their project and then doing the experimental research on the different memory units and input/output devices. While the Institute's machine began to take shape and move towards the engineering phase, Frank Hoyt at Argonne received authorization in mid-1949 to build a copy of the Princeton machine for laboratory purposes. A call went out for anyone at the lab who was interested in being involved in the development of a computer, with the understanding that the project would take two to three years at most.[43]

---

[41] John von Neumann, to Admiral Bowen. Jan. 23, 1946. John von Neumann Papers, Library of Congress, Cited in Aspray, John Von Neumann, p. 56.
[42] Aspray, John Von Neumann, p. 64. This IAS report was titled "Preliminary Discussion of the Logical Design of an Electronic Computing Instrument."
[43] Tropp. Interview with Margaret Butler, Et Al.

While some alterations to the IAS design were expected, Hoyt envisioned the Argonne machine to be essentially a copy of the Princeton machine.[44]

The Computer Section at Argonne

Despite the interest in computing extant at Argonne, there was no one at the Lab who could direct the engineering project. Thus, Hoyt went headhunting. In 1949, there were few engineers in the world with experience building electronic computers. Therefore, the main conduit for technology transfer was by hiring personnel who had worked on one of the earlier projects.[45] In this case, Hoyt contacted and then hired Jeffery Chaun Chu to direct the project.[46] A gifted engineer who had worked on the ENIAC and EDVAC projects, Chu was assigned to the Computer Project along with Ray Kramer, Les Merrill, Carl Bergstrom, Warren Berge, Darren Hill, and Margaret Butler.[47] Holding down the mathematics end of the project were Jim Alexander and Moll Flanders. As reflects the early days of computing, this was a very interdisciplinary group: Chu was an engineer; Merrill had a physic background; Butler, the first woman on the project, was trained in mathematics and had been a "computer" for Reactor Engineering; and Bergstrom had been a technician and had previously worked for Reeves building analog equipment.[48] The group set up in a Quonset hut and was joined in a supporting role by

---

[44] Frank C. Hoyt, to David M. Rubel. Oct. 18, 1949. Laboratory Director's Reading File ANL, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

[45] For more on how people can operate as agents for technology transfer, see David A. Hounshell, From the American System to Mass Production, 1800-1932: The Development of Manufacturing Technology in the United States, Studies in Industry and Society; 4 (Baltimore: Johns Hopkins University Press, 1984).

[46] Thomas Brill, to P.A. Dana. April 22, 1949. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA. Thomas Brill, the head of the Electronics section, made the final recommendation that Chu be hired at a salary of $7,000, which "is entirely justified on the basis of his unique experience in the field of electronic computers."

[47] Ibid.to

[48] Tropp. Interview with Margaret Butler, Et Al.

the extremely adept members of Argonne's electronics section, most especially Thomas Brill who was outstanding both technically and administratively.

Members of this first computing group remember these early days as "lively" and filled with "tremendous spirit." Apart from Chu being the head of the project and Kramer keeping things in order, there was no formal line of command. Since there was more work than could be done by any one individual, technicians often became designers as ideas flowed freely among the personnel.[49]

Beginning in 1949 and into 1950, the computing group at Argonne received blueprints from Princeton on which to base their design of the AVIDAC (Argonne's Version of the Institute's Digital Automatic Computer, coined by Moll Flanders). These were sufficient to build all the registers and the first step of the control unit, which was called the "cycler." By the end of 1950, the "adder" was also completed, but suddenly the blueprints stopped coming as the engineering group at the IAS ran into their own problems designing the computer storage units.[50] With the construction schedule of the AVIDAC stymied, the Argonne group began to design the rest of the control unit and to experiment with different designs for the input/output devices.[51] One of the biggest obstacles to overcome, as in the IAS project, was the development of reliable, fast, randomly accessible memory for the computer.

In 1948, the Institute (as well as IBM engineers) learned of a promising technique developed by F. C. Williams at Manchester University in England whereby cathode ray tubes could be used to store information. "Williams tubes," as they became known,

---

[49] Ibid.
[50] Flanders, to , Tropp. Interview with Margaret Butler, Et Al.
[51] ANL-5080, "Physics Progress Report," Physics Division Quarterly Report. May 1953. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplimental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

stored information in binary form as electric charges on the phosphor-coated inside face of an ordinary cathode ray tube.[52] Attached capacitatively to the phosphor coating is a fine wire mesh grid (or raster) that interacts with the phosphor coating whenever an electron beam is focused at a given point. As the electron beam intersects the mesh, it produces a signal in the wire, and by changing its polarity, it can be used to produce either a 1 or a 0 at a specific location.[53]

Williams had designed his tube so that an electron beam deflected horizontally could read off the 32 digits stored on that line. However, by deflecting the beam both horizontally and vertically, it was possible to access all the storage locations on each tube. With a 32x32 grid, that meant 1024 separate memory locations where one bit of information could be stored.[54] When the machine needed a new instruction (or "word") it would access the same location in 40 different Williams tubes constituting the computer's main memory, simultaneously. Taken in the context of today's personal computers, this random access memory contained 40 x 1,024 bits, or about 5 kilobytes.[55] My laptop, for example, has 384 megabytes of RAM, or about ten thousand times more bits of memory.

The Williams tube memory, while promising, had numerous technological problems to be overcome. In addition to the IAS and IBM engineers searching for solutions, researchers at the University of Illinois and Argonne were also struggling to make the tubes work correctly. At ANL, throughout 1951 and 1952, the electrostatic

---

[52] Phosphor was used because it emits visible light from any spot struck by fast moving electrons. A familiar example of a cathode ray tube is a television screen.

[53] J. P. Jr. Eckert, "A Survey of Digital Computer Memory Systems," IEEE Annals of the History of Computing vol. 20, no. 4 (1998): pp. 18-20.

[54] "Bit," short for "binary digit" is either of the two digits "0" or "1" that are used in computing for the internal representation of numbers, characters, or instructions. It is also the smallest unit of storage for any binary system within a computer. A byte = 8 bits.

[55] Edwin L. Hughes, "Williams Tubes: A Remembrance," in *National Computer Conference, Pioneer Day*, ed. Margaret Butler (Chicago, IL: American Federation of Information Processing Societies, July 17, 1985). For more technical details on Williams Tube memories, see Bashe and others, Ibm's Early Computers, pp. 104-08.

memory system planned for the AVIDAC occupied most of the time of Chu, Jacobsohn, and Kramer. Originally, the group planned to use 40 Williams tubes of 1024 bits per tube for the AVIDAC. However, there were four general problems that made these tubes unreliable, the two most significant being impurity errors and read-around errors.[56] Impurity errors were the product of phosphor imperfections that arose during the manufacture of tubes by Sylvania, resulting in the inability to write a “1” in some specific location of a particular tube. The second, more pernicious problem was called “read-around errors.” Read-around errors caused a “0” at one location to change to a “1” due to some bleeding of electrons from adjacent spots on the tube. Because the charges (indicating either a 1 or a 0) would fade over time, the contents of each memory location had to be restored before the charge leaked below a critical threshold. However, the rapid process of refreshing each memory location (done 10 times per second) plus reading the information stored there, led to some leakage of electrons into adjacent spots. In practice, this meant that one spot could not be accessed too frequently or the surrounding data or instruction would be destroyed.[57]

The problem of impurity errors was reduced through an extensive testing program designed to weed out poor tubes. In addition, the Computer group requested that Sylvania manufacture these tubes under stricter guidelines and then ship them face-up so that impurities, such as bits of carbon could not fall onto the CRT screen.[58] Testing every tube prior to its utilization became standard, and the electronics division at ANL

---

[56] ANL-5174 "Physics Progress Report, Feb. 1954," Physics Division Quarterly Report. Dec. 1952- Feb. 1953. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplimental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

[57] Hughes, "Williams Tubes: A Remembrance," pp. 17-18.

[58] ANL-5080, "Physics Progress Report," Physics Division Quarterly Report. The computer group discovered that going over each tube with a spark coil could sometimes shake these bits of carbon loose and improve the reliability of the tube.

developed both testing equipment and a machine that could remove some of the impurities. Furthermore, a regular tube-replacement program was initiated to rotate CRTs before they developed problems. It was hoped that with these measures in place there would be fewer read-around errors. Despite these efforts, though, when the Williams tube memory unit was finally installed on the AVIDAC in 1953, it was quickly discovered that the memory was not reliable in its 1024 bit configuration. When 512 memory positions also proved to be overly ambitious, the AVIDAC memory was further reduced to 256 positions per tube, after which it worked reliably.[59] With the proposed 40,000 bit memory so reduced, Flanders suggested that an alternative would be “to get 40,000 unemployed Ph.D. mathematicians out in the parking lot and when someone held up the number 71, the correct forty would hold up cards.”[60]

Working around the clock, the Computer Group announced the completion of the AVIDAC in January 1953.[61] What was supposed to be an identical copy of the Princeton machine, in the end only copied the frame of that machine which was designed to be built in one room and moved to another through a relatively narrow door.[62] In terms of its operation, the AVIDAC benefited from being a technological follower of the IAS computer as several important changes were made affecting the AVIDAC’s performance and its programming system.[63] In particular, the Argonne designers treated overflow situations (when a particular output of the machine was too large for the memory to handle) and shift orders (the movement of data one bit to the left or right in the shift

---

[59] ANL-5140, "Physics Progress Report," Physics Division Quarterly Report. 1953. Folder 2: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplimental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.
[60] Tropp. Interview with Margaret Butler, Et Al.
[61] L.C. Furney, "Lab Announces Completion of Electronic Brain," The Argonne News, Feb. 4 1953.
[62] Tropp. Interview with Margaret Butler, Et Al.
[63] For a discussion on the benefits of being a technological follower in terms of the industrial revolution, see David S. Landes, The Unbound Prometheus: Technological Change and Industrial Development in Western Europe from 1750 to the Present (Cambridge: The University Press, 1972).

registers) differently, and included a breakpoint command which allowed the operator to stop the machine at a specific point in a computation.[64] In total, the machine contained 2500 electronic tubes, about 8000 resistors, and over three and a half miles of electrical wire. Powered by 355 storage batteries constantly charged by generators, the AVIDAC took three years to build and cost $250,000.[65]

Announced to the Argonne scientific community through the *Argonne News*, the new computer was intended "to facilitate the solution of mathematical problems of Laboratory scientists engaged in reactor engineering and theoretical physics research work." Impressively, the new AVIDAC could work "approximately 100,000 times as fast as a trained computer using a desk-type electric calculating machine, [meaning that]… a difficult mathematical problem which might take 20 minutes for the AVIDAC to complete would take, perhaps three years for two mathematicians to do with desk-type calculators." Reflecting the unfamiliarity that many scientists had in regards to these new electronic computers, the article went on to explain that the AVIDAC used the binary number system and gave the example that the number 37 in the decimal system would be expressed as 100101 in the binary system.[66]

Not surprisingly given the rapid advances in the capabilities of computing equipment, by the time the AVIDAC was ready for laboratory use its performance was already exceeded by a second machine, the ORACLE, which had been built simultaneously at Argonne by the Computer section staff.

In terms of programming the AVIDAC, the Argonne staff built upon von Neumann's and Arthur Goldstine's three volume report entitled "Planning and Coding of

[64] Tropp. Interview with Margaret Butler, Et Al.
[65] Furney, "Lab Announces Completion of Electronic Brain."
[66] Ibid.

Problems for an Electronic Computing Instrument."[67] Although this report, issued in 1947 and 1948 did not explicitly discuss input/output equipment since the authors figured this technology would change quickly, it did offer a general set of principles for programming these new machines. In particular, the report described a system of flow diagrams by which the logical structure of programs could be visually described on paper and included a lengthy discussion of a programming methodology based on the use of subroutine libraries. Von Neumann and Goldstine stressed that programming was no longer a straightforward, linear, process as it had been for early calculating machines. For electronic computers to achieve their full flexibility they had to be able to jump backwards or forwards through their instruction set, and even modify those instructions, depending upon the result of a previous computation. Flow diagrams, then, with their arrows, branches, and boxes, could illustrate both the static and dynamic characteristics of programs.[68]

One other significant contribution of this report, and one that would have a tremendous influence on the future work of the Argonne computer staff, was von Neumann's and Goldstine's description of subroutine libraries. The logical architecture of older calculating machines, including the ENIAC, required that an entire program be coded anew, even if the new program shared attributes with earlier programs. However, with subroutines, frequently used programs, such as codes to compute square roots, sines, cosines, and exponentials were written and stored externally to be called upon by the computer whenever these functions were required. Creating libraries of subroutines would thus save programming time.[69]

---

[67] Tropp. Interview with Margaret Butler, Et Al.
[68] Aspray, John Von Neumann, pp. 68-9.
[69] Ibid., pp. 71-2.

Within the Computing group at Argonne, the development of subroutine libraries began almost as early as the design and engineering phases of the hardware. With programming talent scarce, in June 1951 Flanders contacted Jean Hall, one of the women who had programmed the ENIAC, about working with the AVIDAC. Flanders envisioned Hall both advising the lab on its use of computing equipment (including the AVIDAC, the lab's analog computer REAC, and the CPC) as well as programming problems on the various machines, especially the AVIDAC. However, with the staff size of the Computing group limited, Hall was hired on the condition that she also share in the hand computing work at Argonne until internal adjustments could be made.[70] She accepted the job, and soon after relocating from California to Argonne, Hall and Margaret Butler began to develop the first subroutine libraries for the AVIDAC. In addition to the standard sine, cosine, square root, and logarithmic functions, the programming group spent considerable time coding subroutines that could handle floating-point arithmetic, where numbers are expressed in scientific notation.[71]

Programs were entered into the AVIDAC on five-hole punched paper tape. Using a modified teletype and Western Union equipment, programs could be loaded at a "read-in" rate of about twelve four-digit characters per second and a "read-out" rate of about six four-digit characters per second. Inconveniently, the tape reader tended to perforate the tapes preventing their repeated use and requiring that the AVIDAC be used to reproduce new tapes.[72] It was clear that this was far too slow for effective use of the AVIDAC and

[70] Flanders, to Jean Hall, June 21, 1951. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.
[71] ANL-5080, "Physics Progress Report," Physics Division Quarterly Report. Floating-point arithmetic, if implemented efficiently, would allow small straightforward computations to be performed with a minimum of special preparation.
[72] ANL -5031 "Physics Progress Report," Physics Division Quarterly Report. 1953. Folder 2: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

after some searching, the staff settled on a Ferranti tape reader that could handle inch-wide, seven-hole paper tape at a top speed of 200 characters per second and a faster tape punch which could operate at 24 characters per second.[73]

The first full-scale problem run on the AVIDAC was posed by Enrico Fermi, who suggested that the computation of trajectories of pions emerging from the magnetic field of a cyclotron would be a suitable and simple test program for the Argonne computer. The calculation consisted of:

> "determining the intercepts (X,Y) of the line along which the pion travels after it enters the field-free region, as a function of the energy and the original direction it was traveling. The result of this calculation allows one to calculate the direction in which the pions of a given energy are traveling and their intensity at any point in the field-free region relative to the intensity of the radiator."[74]

With the success of this program, the AVIDAC was relocated to the Physics Building and soon was busily at work for the Reactor Engineering division modeling reactor cores. As the design problems became larger and more complex, the staff responded by gluing together several paper tapes representing different aspects of a computation with Duco cement. In many cases, the paper tape would be stretched down the long halls of the physics building so that it would not tear while being loaded into the computer. Likewise, to do multi-group calculations, where multiple iterations were required to solve a problem, the paper tape was spliced into loops so that it could go around the reader in a circle.[75]

The AVIDAC, for all its technical difficulties, was still a remarkable achievement. In terms of its place in the history of electronic computers, it follows

---

[73] ANL-5080, "Physics Progress Report," Physics Division Quarterly Report. The AVIDAC later used the new Teletype DROE Burpee punch which could punch 60 characters per second.
[74] Ibid.
[75] Interview with Margaret Butler, Argonne National Laboratory, by author, 11-27-2001. Tropp. Interview with Margaret Butler, Et Al.

directly on the heels of the IAS machine, and is located on a main branch of a “lineage tree” produced for the American Federation of Information Processing Societies’ Pioneer Day celebration in 1983 (See diagram page 64). For the Argonne Computing Section, building the AVIDAC provided an opportunity to experiment with different logical structures, programming techniques, and physical components that make up an electronic computer. The machine itself was heavily used by Argonne scientists until its retirement in 1957. Its reliability was also improved. With the addition of air conditioning to its arithmetic unit in May 1954, the AVIDAC was available for 231 hours out of a possible 311 from May to July of that year.

An analysis of the usage patterns of the AVIDAC for the year 1956 demonstrates that computer use at Argonne expanded rapidly to fill available hours. While it is not surprising that the Physics and Reactor Engineering Divisions dominated computing resources, it is also clear that the Life Sciences and Chemistry Divisions were beginning to find ways to incorporate this new computational tool in their research. In one month at the end of 1955, the AVIDAC was utilized a total of 624 hours. Of this, roughly 496 hours were devoted to problems from the Physics Division, 143 to Reactor Engineering, 25 to Material Research, but only 3 hours to chemistry.[76] By the middle of 1956 it was clear that computer utilization was spreading among the other divisions. In June of that year, Physics and Reactor Engineering still absorbed the bulk of available hours at 265 and 223 each, but Chemistry Division usage expanded to 148, Analytic Chemistry 175, and biology 10.4 hours respectively.[77] This trend continued throughout the year, and by

[76] Jean F. Hall, to L. R. Wallis. "AVIDAC and Programming Time Report for November 26 through December 25, 1955," January 4, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.
[77] Jean F. Hall, to L. R. Wallis. "AVIDAC and Programming Time Report for April 26 through May 25, 1956," May 29, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

the end of 1956, even the Biology Division used 88 hours, or about 12% of the total time the AVIDAC was available that month.[78]

Looking at these numbers, it would appear that the Reactor Engineering and Reactor Physics groups did not require as much computer time as Physics. In reality, the opposite was true. However, with the AVIDAC supporting the computational needs of all the Argonne divisions, the reactor groups looked elsewhere for computer time. Beginning in 1953 the Atomic Energy Commission arranged computer time for Argonne's reactor groups on the UNIVAC 4 at their computing facility at New York University. Directed until 1958 by the renowned applied mathematician Richard Courant, an old colleague of Flanders', the UNIVAC 4 was heavily used by the two reactor groups to compensate for the lack of time available on the AVIDAC.[79] In addition, the Reactor groups also used UNIVAC machines located at the Bureau of the Census in Washington, D.C., the Nuclear Development Corporation of America in White Plains, New York, and at a U. S. Steel plant in Gary, Indiana.[80]

That the computational needs of the Argonne research community exceeded the capacity of the AVIDAC was no surprise to the members of the Computing Section. Despite the changes made at Argonne to the IAS design, copying the Princeton machine also meant that the AVIDAC suffered from many of the same weaknesses: small,

---

[78] Jean F. Hall, to L. R. Wallis. "AVIDAC and Programming Time Report for the Period December 26, 1956, through January 25, 1957," January 29, 1957. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

[79] Apparently Flanders' relationship with Courant may have been the key to Argonne getting machine time. In general, time on the UNIVAC 4 was allotted to those AEC facilities that did not have their own computer. See Tropp. Interview with Margaret Butler, Et Al. For a short discussion of the placement of the UNIVAC 4 at NYU, see Reid, Hilbert-Courant, pp. 507-13.

[80] Margaret Butler, to Linda Hoof. July 13, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA, Margaret Butler, to Robert Johnson. "Univac Time Records," July 6, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

unreliable memory, slow processing speed, and a relatively inefficient logical structure. As early as 1950, Chu and his engineers had already worked out a better design for a computer, including a more reliable electrostatic memory, but were unable to incorporate many of their ideas into the AVIDAC. However, that year, as the AVIDAC was being built, Argonne entered into a contract with Oak Ridge National Laboratory to design and build a computer for them. Also an AEC laboratory, Oak Ridge (the lab that was built in Clinton, TN) was heavily involved in large scientific calculations involving Monte Carlo simulations and was looking to acquire their own computer. Initially, the Oak Ridge group wanted an exact replica of the AVIDAC, but ANL had no interest in constructing another such machine and offered Chu's services only as an occasional consultant if that was their desire.[81] After some deliberations, Oak Ridge agreed to have the Computer Section, with the help of four engineers from the Tennessee lab, build a new machine that would incorporate modifications to the AVIDAC/IAS design. In particular, the engineering design was changed quite a bit to improve its reliability, speed of operation, and maintenance. Initially projected to cost $200,000, the ORACLE (Oak Ridge Automatic Computer Logical Engine) as it became known, eventually cost $350,000.[82]

The ORACLE's unveiling was timed to coincide with an ANL-sponsored Symposium on Large Scale Digital Computing Machines held at Museum of Science and

[81] J. T. Bobbitt, to W. H. Zinn. "Summary of Significant Factors with Respect to the Electronic Computer Arrangements with Ornl," Oct. 4, 1950. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

[82] R. F. Kramer, to H.C. Nickel. "Development and Construction Costs, Oak Ridge Computer," Oct. 10, 1950. Laboratory Director's Reading File, ANL 1949-1957, RG326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA. Reflecting the difficulty in coming up with original names for computers, the ORACLE was supposed to be called the ORAC, for Oak Ridge Automatic Computer. However, after some investigation, the Argonne team discovered another computer, of the differential type, in Japan with that name. At another point in its construction a radium spill contaminated certain parts of the machine and the name ORXAC was suggested to mean Oak Ridge Automatic Computer with radiation. In the end, following a suggestion from Wallace Givens, a future director of the Applied Mathematics Division at ANL, it was dubbed the ORACLE. See Tropp. Interview with Margaret Butler, Et Al.

Industry in Chicago on August 3-5, 1953. Attesting to the wide interest in digital electronic computers, 164 people attended, representing 18 universities, 29 industries and industrial labs, and 11 governmental agencies.[83] On the occasion of the tenth year anniversary of the completion of the ENIAC, the presenters focused primarily on the engineering aspects of the design of digital computers: six papers were devoted to cathode ray tubes as storage devices; another six were on Williams Tube Memory specifically; four papers were presented on large scale ferromagnetic memory; the staff of Bell Laboratories spoke about the application of transistors to high-speed computers, and there were several papers on the programming and application aspects of these new machines. The list of presenters was a veritable who's who of early computing, including Isaac Auerbach, Jay Forrester, Nicholas Metropolis, Herman Goldstine, and J. W. Mauchly.[84] Thus, it was truly a feather in the cap of the Computing Section at Argonne to announce the completion of the ORACLE to this distinguished group of scientists, mathematicians and engineers.

When completed in August 1953, just six months after the AVIDAC, the ORACLE was the fastest general-purpose electronic digital computer in the world.[85] It contained three key features which distinguished it from other computers. First, it had the largest capacity internal memory system of any machine -- about twice that of the AVIDAC. The redesign of the electrostatic memory was especially significant as the ORACLE's memory consisted of forty separate plug units which could be switched out

[83] ANL-5140, "Physics Progress Report," Physics Division Quarterly Report.
[84] J. C. Chu, ed., A Symposium on Large Scale Digital Computing Machines, vol. ANL-5181 (Museum of Science and Industry, Chicago: Argonne National Laboratory, 1953). The list of attendees is equally impressive, including several people who would go on to make significant contributions to the field: Gene Amdahl, J. Presper Eckert, Jr., and Alan M Perlis among others.
[85] "Scientists Complete and Operate World's Fastest Electronic Brain," The Argonne News, Sept. 2 1953.

with a minimum of effort and down-time.[86] Second, it had the largest on-line intermediate storage which could handle four million words of calculations and data all on an internally-controlled 42-track magnetic tape. Third, it was the fastest general purpose computer, able to multiply 9-digit numbers such as 782,901,231 by 138,166,869 in less than 1/2000th of a second. The addition of two 10-digit decimal numbers only took 5/1,000,000th of a second.[87] Despite containing 3,500 tubes, 20,000 resistors, and seven miles of wire, it was billed as “exceedingly compact” as the arithmetic unit was only twelve feet long, four feet wide, and seven feet high and the internal memory unit was only slightly larger.[88] Relocated to Oak Ridge, Tennessee in October, 1953, the ORACLE operated on a three-shift, five-day per week schedule until it was finally retired in the fall of 1962.[89] The average up-time for all planned work was approximately 95% and for the eight years of its service, it served as the prime scientific computer supporting the research efforts of a 2,000 person laboratory staff.[90]

With the ORACLE earmarked for another laboratory, the Computer Section also began exploring ways to bolster the computing power at Argonne. Initially, the group considered purchasing a newly developed magnetic core memory for the AVIDAC.[91] However, the prospect of placing the large capacity memory system on an outdated computer was unappealing to a staff eager for a new challenge. Sometime during the

---

[86] The new memory units were designed almost entirely by Bud Klein, one of the ORNL engineers who had moved to Argonne for the project. See Tropp. Interview with Margaret Butler, Et Al.
[87] Earl W. Burdette and Rudolph J. Klein, "A Decade of Oracle Experience," in *National Computer Conference, Pioneer Day*, ed. Margaret Butler (Chicago, IL: American Federation of Information Processing Societies, July 17, 1985).
[88] "Scientists Complete and Operate World's Fastest Electronic Brain," p. 3.
[89] Shipping the ORACLE required 105 wooden boxes and crates, loaded on two thirty foot truck trailers.
[90] Burdette and Klein, "A Decade of Oracle Experience," p. 12.
[91] ANL-5237, "Physics Progress Report, Sept. 1954," Physics Division Quarterly Report. Sept. 1954. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplimental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute. A contract was ultimately signed with International Telemeter Corporation to produce a 4096-word magnetic core memory unit.

summer of 1954, the decision was reached that rather than revise the AVIDAC to fit the core memory, a new computer would be built, named "GEORGE."[92] The preliminary logical design was already done, and when completed, it would be twice as fast as the ORACLE and four times as fast as the AVIDAC.[93]

Tentatively projected to be completed in 1955, GEORGE was not finished until late 1957. Building on experiences with the AVIDAC, ORACLE, and the Serial 4 UNIVAC at NYU, GEORGE incorporated a new 4096-word magnetic core memory unit built by the International Telemeter Corporation and included a more versatile instruction set to make programming easier. Additionally, GEORGE was optimized to process quickly half-precision or 19-step multiply and divide instructions, in order specifically to facilitate Monte Carlo calculations.[94]

Once completed, GEORGE superseded the AVIDAC as the main computational workhorse for the entire laboratory. At about the same time, though, the activities of the Computer Section came under greater scrutiny by AEC officials and Argonne managers. During the period (1949-1957) when Argonne engineers and scientists were designing and building their own general-purpose computers, efforts were made to find commercially produced equipment that could handle Argonne's computational load. With the computing industry in its infancy, it was advantageous for labs like Argonne to build their own machines geared towards specific applications in the nuclear sciences. By 1957, however, it was recognized that companies such as Sperry-Rand and IBM had

---

[92] Breaking away from the tendency to name computers such that they end in "AC," GEORGE refers to a common statement in the 1950s "Let GEORGE do it." The significance of this phrase is somewhat lost in translation; however, if one "Google's" it, there are many links to sites that add insight into the "GEORGE" turn of phrase.

[93] ANL-5412 "Physics Progress Report, Sept. 1955," Physics Division Quarterly Report. 1955. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplimental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

[94] Margaret Butler, "Argonne National Laboratory Computing."

made great strides in the design and construction of powerful computers and it no longer seemed economically justifiable or technically desirable for Argonne to continue to build its own machines.[95] Thus, research activities in the development of general-purpose computers were gradually phased out as the lab began shopping for its first commercial computer.[96]

The Move to Commercial Computers

With the computational load of the Reactor Division increasing steadily, Flanders held a meeting in his office in November of 1957 to discuss which commercially-built computer to acquire in order to serve the division's pressing needs. The two candidates at the time were the Sperry-Rand Univac Scientific 1103A and the IBM 704.[97] Inquiries were sent to both companies with a list of specifications for the desired systems and questions concerning the kinds of services that would be provided with the system, such as what subroutines would be included and the form in which they were available. In addition, whichever computer was selected had to be equipped to perform floating-point arithmetic, include magnetic core and magnetic drum storage, auxiliary tape units, printers, and the requisite input/output equipment.

At the time, it was not clear which computer would be better for Argonne. The 1103A was originally the brainchild of Minneapolis-based Engineering Research Associates (ERA) which Remington-Rand (later Sperry-Rand) purchased in 1952 in an

[95] Not surprisingly, the rapid emergence of the commercial computing industry was due in large part to the knowledge these companies gained while helping to design and manufacture computers for the military and federal government. In short, the AEC helped to subsidize, both financially and technically, the development of the computer industry. See Peter J. Westwick, The National Labs: Science in an American System, 1947-1974 (Cambridge: Harvard University Press, 2003), pp. 227-29.; and Kenneth Flamm, Creating the Computer: Government, Industry, and High Technology (Washington, D.C.: The Brookings Institution, 1988), pp. 29-133.
[96] Butler, "Argonne National Laboratory Computing," p. 5.
[97] For more on the development of the IBM 704, see Bashe and others, IBM's Early Computers, pp. 178-80.

effort to bolster its scientific computing division. When combined with the prestige of UNIVAC, which Sperry-Rand also owned, the 1103A was considered a reliable, "scientifically-oriented" computer.[98] Moreover, the Computing Section at Argonne had been using UNIVAC equipment for several years and was familiar with its programming system and characteristics. However, there were rumors that Sperry-Rand was reluctant to fully support the electronic computer division for fear that it would cut into its traditional punched-card business. This left Flanders and his team to wonder whether the 1103A was the correct choice for Argonne.[99]

On the other hand, the IBM 704, available in late 1955, had much to recommend it. In essence, it was a redesign of the IBM 701, which had been a fairly successful computer when first sold in 1953. However, the new version was a substantial improvement over its predecessor, incorporating floating-point hardware, which allowed it to process certain jobs between two and twenty times faster. In addition, the 704 utilized a new, more efficient instruction set, had faster tapes and drums, possessed a 32,768 word core memory, and included new programming tools as well as the expertise of a growing user community called SHARE.[100]

In the end, the decision of which system to acquire had less to do with each system's respective technical merits than it did with the fact that IBM had a considerable

---

[98] As late as August 1955, the reputation of the UNIVAC was such that they were outselling their IBM 700 series counterparts by about 30 to 24. See Martin Campbell-Kelly and William Aspray, Computer: A History of the Information Machine, The Sloan Technology Series (New York: Basic Books, 1996), pp. 111-27.
[99] Ibid., pp. 127-8, Donald A. Flanders, to J. R. Foote. Jan. 28, 1957. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.
[100] Emerson W. Pugh, Building Ibm: Shaping and Industry and Its Technology, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge: The MIT Press, 1995), pp. 179-80. For more on the organization and activities of the IBM user group SHARE, see Atsushi Akera, "Voluntarism and the Fruits of Collaboration: The Ibm User Group, Share," Technology and Culture vol. 42, no. 4 (2001).

lead in developing software programs related to the nuclear sciences.[101] Often referred to as “network externalities” by historians of technology and economists, this meant that the IBM 704 was more valuable to Argonne researchers than the 1103A because more people were using it to do reactor design work. Network externalities are often defined as those features or forces which make compatibility across users important. These forces are particularly important for driving technical standardization. In the case of the 704, there were already more programs available for it and thus easier to get up and running for Argonne’s needs.[102]

At the time Argonne was considering the 704, there were already twenty-four such machines installed around the country. Two of those were located at Los Alamos, three at General Electric plants, and one at Westinghouse.[103] Both GE and Westinghouse were actively developing their own nuclear energy programs and had demonstrated a willingness to exchange information with Argonne researchers.[104] More importantly, members of the Reactor Division weighed in heavily in favor of the IBM 704 because of an on-going reactor safety program being conducted jointly with Los Alamos. In a letter to Flanders, D. Okrent of the Reactor Engineering Division insisted that this collaborative project would be severely hampered unless Argonne had the same type of computing

---

[101] D. Okrent, to Donald A. Flanders. "Choice of Ibm 704 over Other Computing Machines," Nov. 5, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

[102] This issue of network externalities, already evident in 1956, was in part the basis for Microsoft’s dominance in personal computer operating systems. As more people installed the Windows operating system (OS) on their computers, the more likely it was for software developers to provide programs to run under this OS. Likewise, the more programs developed for Windows, the more likely it was that consumers wanted Windows in order to run these programs. The result was that Microsoft Windows came to dominate ninety percent of the personal computer market.

[103] General Information Manual: Machine Information, 704's Installed. 1956. Applied Science, IBM Corporate Archives. As of June 1956, the AEC NYU facility was also scheduled to have an IBM 704 installed. See Donald A. Flanders, to J. R. Foote. "Procurement of a Digital Computer," Nov. 7, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

[104] Margaret Butler, to B. Spinrad. "Reactor Programs for the Ibm 704," January 24, 1957. Laboratory Director's Reading File, ANL, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

machine as Los Alamos. Although the calculations involved in the Reactor Safety Program could be done on an 1103, he warned that the basic programs would have to be re-coded for the new machine and due to the classified nature of the work, his physics staff would have to undertake this job:

> "While the use of cards and the general flow sheet diagrams will be unclassified," he argued, "the details of the codes (present and future) in this field of atomic explosions are likely to be highly classified. Hence we cannot rely on the use of some outside unclassified group to transfer programs from the 704 to some other machine, and, of course, our own programming staff should not be loaded with this major job."[105]

In addition, Okrent also pointed out that the use of different computers by Argonne and Los Alamos would only lead to confusion in the comparison of results.[106]

Okrent also made a second argument in favor of the IBM 704 and one that suggests the importance of the SHARE user group to customers interested in acquiring a large digital computer. He described an automated IBM-based system for analyzing data from transient tests that was operating at Phillips Petroleum. He then suggested that a similar system could be implemented at Argonne related to the processing of experimental data from kinetics experiments in both the fast and thermal reactor systems. While acknowledging that the incorporation of computers directly into experiments for the capture and analysis of data was still some way off, Okrent contended that "it is clear that Argonne must accomplish some similar automation, if the proper theoretical treatment is to be given to the forthcoming experimental data on reactor kinetics."[107] The implication was that the knowledge gained by the Phillips Petroleum people on how to perform this analysis using IBM equipment would be readily available to Argonne researchers, provided they had the same equipment.

---

[105] Okrent, to "Choice of Ibm 704 over Other Computing Machines,"
[106] Ibid. That different computers might yield different answers is one of the core issues of Chapter 5 and at the heart of the Applied Mathematics Division's efforts to develop quality mathematical software.
[107] Ibid.

Despite a strong proposal from Sperry-Rand in which the company offered its computer and services at a lower cost than IBM, Flanders decided to rent the 704. Writing that "anything like an accurate dollars-and-cents assessment of their relative merits is extremely difficult, if not really impossible," Flanders concluded that the 704 was "a somewhat faster and more versatile machine for our purposes." In justifying his decision, he echoed Okrent's arguments that reprogramming current and future IBM 704 programs for the 1103A would be an "undesirable" use of available manpower, and suggested that "the time-lag involved in reprogramming… would be a hindrance to the flow of information between reactor-development groups."[108]

Beyond the availability of reactor programs and the ability to share codes and computational techniques with other 704 users, the design of the IBM machine was better suited to the cramped space in which it was to be placed. Whereas the 1103A units were fixed to a manufacturer-supplied platform and very difficult to move, the 704 components were freely moveable, making it possible to house the 704 in a smaller area, thereby gaining space for staff personnel.

For a rental rate of $37,160 per month, Argonne received its IBM 704, the first unit installed in the Chicago, area in mid- 1957,.[109] In terms of performance, GEORGE and the 704 were approximately the same, but the 704 carried the load for reactor engineering. One of its first tasks was to help scientists at the lab determine the types of materials that could be built into the core of Argonne's Experimental Breeder Reactor II project, while remaining time was spent helping to design a 12.5 Bev particle accelerator.

The shift from "producing" to "consuming" computers also entailed a reorientation of the computer engineering activities at Argonne. After 1957, the primary

[108] Flanders, to J.R. Foot, Jan, 28, 1957.
[109] Ibid.

focus of the group was to design and build special purpose computing equipment to assist scientists at the lab in the capture and analysis of experimental data. However, the computer engineering group did build one more large-scale experimental machine. FLIP (Floating Indexed Point), which was completed in 1963. In a unique setup which was designed to test new system concepts, FLIP was attached to GEORGE in a system called GUS (GEORGE Unified System). While GEORGE was responsible for data formatting, converting, sorting, the handling of input-output functions, and communication with peripheral equipment, FLIP was designed as the principle arithmetic unit in this experimental multi-computer system.

FLIP was the product of intensive efforts by mathematicians to perform error analysis by machine when carrying out floating-point arithmetic. Since only a finite number of digits can be represented in a machine, errors are introduced during arithmetic operations. While good algorithms can keep these errors small, over the course of thousands or even millions of iterations these errors can accumulate, thereby drastically affecting the results of long computations. FLIP provided a unique feature which indicated the accuracy of numbers generated and stored in the machine. The measure of this accuracy was recorded in an "index of significance" which then accompanied the final results of a computation. This facility provided a means for studying the propagation of errors throughout large calculations.[110] Providing an index of significance alongside the computational results allowed better correlation between the computer output and the results of theoretical or experimental calculations, as well as helping mathematicians refine their computer algorithms.

[110] "Introducting Gus -- a Multi-Computer System," The Argonne News, Feb. 1964, pp. 6-7.

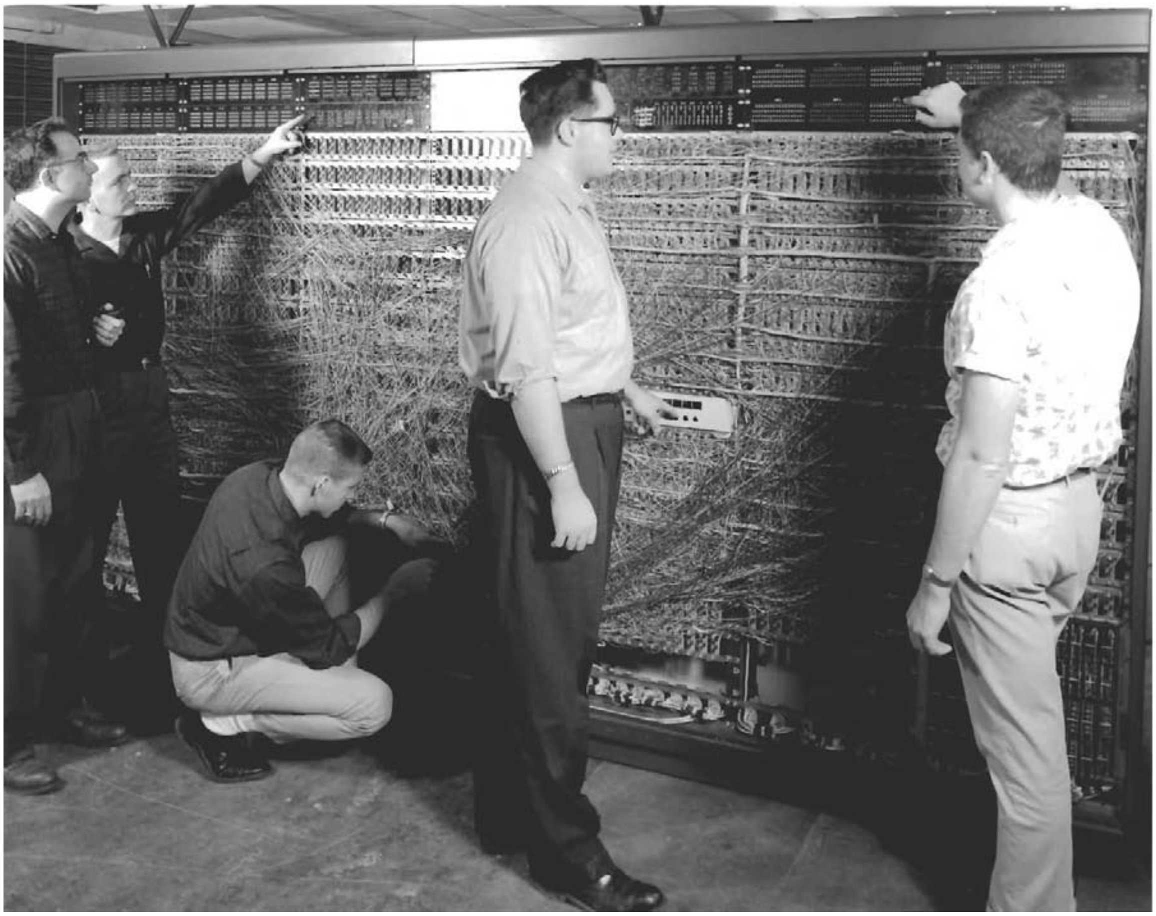
The acquisition of the IBM 704 in 1957 not only signaled the end of the large-scale computer engineering projects at Argonne, but also placed the laboratory on a computer migration path that, with the exception of a CDC 3600 installation in the early 1960s, stayed with IBM mainframes until the 1980s. As in the early days, these large computers were housed in a central facility and were the primary computational resource for the entire lab. Even as computers became smaller and other divisions at Argonne began to acquire their own machines to handle data collection and analysis, "Big Iron" was still the predominant tool for large computations.

Conclusion

Argonne was, without question, a pioneer in the creation and use of digital electronic computers. They built their first large-scale machine, the AVIDAC, at a time when few other federal organizations could undertake such a project and when the commercial computer industry was almost non-existent.

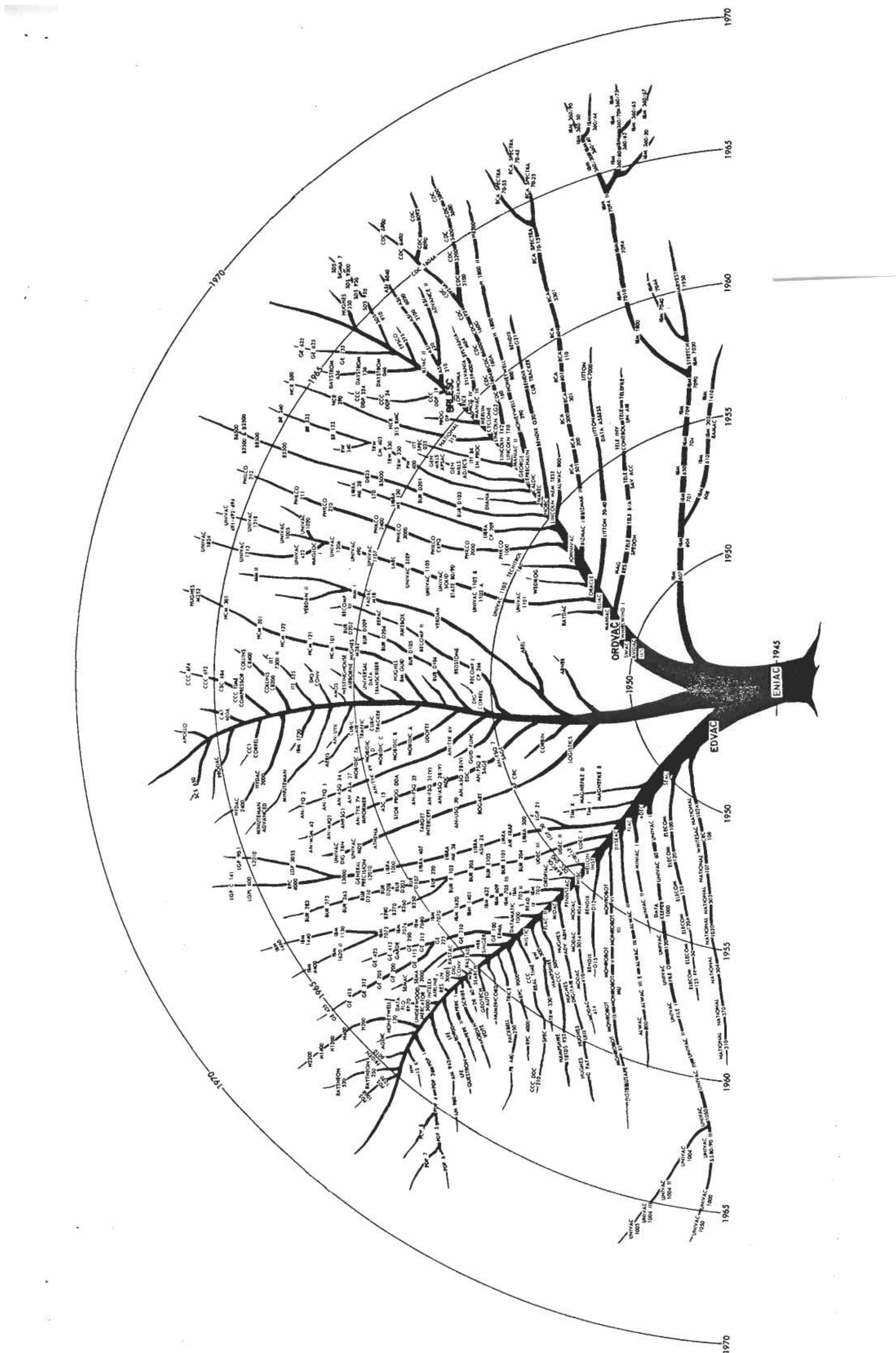
For the engineers, mathematicians, and scientists involved in building the AVIDAC, ORACLE, GEORGE, FLIP, and GUS, this was an exciting time. As previously mentioned, these machines were much more than fast calculators; the fact that computers could be programmed meant that they were objects of scientific interest in their own right. In the broadest sense of the term, these engines of computation were the experimental equipment around which the discipline of computer science would begin to emerge over the next decade. The decision to build computers had been justified by the AEC because they could help scientists and engineers perform calculations quickly and thus they supported the programmatic activities of Argonne. What was not clear at the time, however, was the extent to which the application of these computers required extensive research into such esoteric and uncharted fields as the theory of computation. More troubling still were questions about who would pursue this work, how was it to be organized, and how would this research support the scientific activities at Argonne? The next chapter examines this issue, showing how computers became the technological foundation on which entirely new disciplines were created.

1. Mathematician Donald Flanders and engineer J.C. Chu standing at the AVIDAC. The cylinders protruding from the machine are the William's tube memory units.
Photo courtesy of ANL photo archives.



2. Argonne engineers and scientists working on the experimental computer ORACLE which was built at ANL and then sent to Oak Ridge National Laboratory. At the time of its completion, the ORACLE was the world's fastest computer.
Photo courtesy of ANL photo archives.

3. Members of the AMD gathering around their experimental machine GEORGE-FLIP. Courtesy of ANL photo archives.



4. Genealogy of early computer systems. The AVIDAC falls along the 1950 arc on the third branch from the right. Drawing by Martin Weik. Provided courtesy of Margaret Butler, circa 1986.

Chapter 2

Applied Mathematics, "Hybrid Areas" and the Social Organization of Computational Science at Argonne National Laboratory, 1949-1975

> "In every special doctrine of nature, only so much science proper can be found as there is mathematics in it." Kant, Metaphysical Foundations of Natural Sciences.[1]

> "Past experience has demonstrated that research and development in all scientific fields will become progressively more mathematical in character, and that the rate of this progression will depend in a large measure on the work of applied mathematicians." William Miller, Applied Mathematics Division Long Range Plan, Feb. 28, 1962.[2]

> "Interacting with forces pushing us in the same direction, the advent of the high-speed computer has opened the way for an unprecedented mathematization, not only of fundamental scientific research in the physical and biological sciences but also in the management of our industrial and social systems. This is about to assign to mathematics an entirely new part in our civilization with far-reaching implications on what should be taught, how it should be taught, and to whom." F.J. Weyl. Quoted in [Curtiss, 1957 #257]

On November 1, 1956, Argonne National Laboratory administrators announced the creation of a separate Applied Mathematics Division (AMD).[3] Although the initial announcement was greeted with little fanfare, over the next twenty years the AMD would

[1] Felix E. Browder, "Mathematics and the Sciences," in History and Philosophy of Modern Mathematics, ed. William Aspray and Philip Kitcher (Minneapolis: University of Minnesota Press, 1988), p. 290.
[2] W.F. Miller. Applied Mathematics Division Long Range Plan. Argonne, 1962. AMD Budget Report, 1962, B93-11025 Director's Subject File, MCS Archives, Argonne National Laboratory.
[3] Applied Mathematics Division Summary Report, Nov., 1956- June, 1958. AEC Research and Development Report, 1958. ANL-5954 Mathematics and Computers (TID-4500) 14th ed. MCS Files, ANL.

provide the foundation on which computational science developed at Argonne. The significance of the AMD was not due to its activities as a computing service bureau; applied mathematics "laboratories" had existed in both business and academia for years. Instead, its importance lay in how clearly the AMD's evolving organization reflected competing visions for how computers could aid in the production of science. Thus, it provides an ideal case study in which to examine how disciplinary agendas, technical change, and institutional structures contributed to the emergence of computational science as a third methodology alongside theory and experiment for investigating natural phenomena.

To the reader, sitting beside a powerful personal computer on the desk, the extent to which early computing was initially a *group activity* may not be obvious. From the 1940s through the mid 1960s, the application of computers to questions of science and engineering required collaboration between different groups of people who often had very different disciplinary and professional agendas. In such an environment it should not be surprising that questions of scientific authority and expertise would be contested, especially when it came to questions of *access* to computers.

At the same time, the rapidly changing nature of computer technology itself was a crucial component in how various professional roles in scientific computing were negotiated and then renegotiated. Innovations in computer architectures, software, computer memories, input/output devices, and programming produced corollary changes in the dynamics of interdisciplinary collaboration that made up early computing. Whereas the preceding chapter focused on efforts at Argonne to design and build electronic digital computers for scientific work, in this chapter I shift my lens to examine the extent to which computers became the material foundation on which new professional identities were built.

Although my argument remains centered on the AMD at Argonne, the issues confronted by members of the division were by no means unique to their situation. As computers quickly became indispensable to many scientific investigations, an increasingly difficult question arose: what role should computer specialists and mathematicians play in the construction of new scientific knowledge? This question is more than academic; particular configurations of computer, computer specialist, and scientist had significant implications on how scientific computing, and later computational science, was done.

The approach seemingly most profitable in understanding the contested terrain of scientific computing as a social activity is to divide the relevant interest groups roughly into *producers* and *consumers*. While this is certainly not the only analytical framework which can be applied to this topic, bifurcation into these two categories does provide a convenient way to identify the various positions of the people involved in scientific computing at Argonne. The notion of producers and consumers has also gained currency within the discipline of the history of technology as a way to investigate the interplay between these two groups and how it shapes technologies (and vice versa).[4]

In the case of scientific computing, these categories are somewhat slippery. Many scientists were both producer and consumer. I therefore qualify my two groups: producers are those researchers engaged *primarily* in the development, application, and diffusion of new computers, programming methodologies, and computational tools.

---

[4] Ruth Schwarz Cowan, "The Consumption Junction: A Proposal for Research Strategies in the Sociology of Technology," in The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology, ed. Wiebe E. Bijker, Thomas P. Hughes, and Trevor F. Pinch (Cambridge: The MIT Press, 1987), Nelly Oudshoorn and Trevor Pinch, eds., How Users Matter: The Co-Construction of Users and Technology (Cambridge: The MIT Press, 2003).

Consumers, while they might do some programming and develop limited computational techniques, generally rely on others to create the software and computer hardware that they use in their research.

The producer/consumer dichotomy at Argonne makes one thing clear: there was a fundamental struggle between these two groups over the meaning of computer technologies *writ large*. The organizational changes evident in the Applied Mathematics Division from 1957-1975 correspond to a dynamic vision for how computers could be integrated into science and engineering practice. For AMD directors, computers could be applied to the investigation of almost any natural phenomena. Doing so required above all the creation of new mathematical tools and techniques useful to scientists and suitable for machine computation. For producers of computational tools, computers held a particular meaning that transcended the hardware. Applied mathematicians, especially, saw in the machine a world of possibilities: fundamental research in what would be called the science of computing, the potential for interdisciplinary collaboration, and maybe even heightened professional prestige.

Consumers, on the other hand, had a different understanding of computers that also went beyond the machine. To them, computers were simply a *tool*. Consumers did not envision computers as a means to professional empowerment or as a basis for interdisciplinary collaboration. Instead computers, *and the people who tended them*, were to act in a support capacity producing computational solutions as needed. Scientists tended to want the same thing from the providers of computational services that they expected from their maintenance staff: problems solved on demand.

At the AMD's inception in 1957, this service component was made quite explicit in its list of responsibilities. At the same time, the directors of the new division attempted to meld both producer and consumer visions together. As one of the pioneering computer facilities in the world it was thought by some that the AMD might serve as a model for how to organize and apply computing technologies to science and engineering research. As initially formulated, the charge to the AMD was a mixture of concrete computational services and more abstract research-oriented activities in mathematics and machine design.[5] Although their charter included provisions for mathematical research, for the most part laboratory management, scientists, and the AEC emphasized the division's service duties. In this conception the Applied Mathematics Division would be the latest incarnation -- albeit with fancier equipment -- of the mathematical service bureaus that had dotted certain American industries since the 1920s.

However, for the first two directors of the new division, Donald "Moll" Flanders and William Miller, the service component ascribed to the new division was secondary to what they considered a larger project. To them, it seemed that the computer held considerable promise for energizing old disciplines like numerical analysis, creating new areas of expertise, and possibly destabilizing disciplinary boundaries. The key to such a future lay in the increasing mathematization of research and development in almost every scientific field.[6] Because computers were essentially mathematical instruments, it seemed reasonable that they could lever to dissemination of mathematical expertise and techniques into other scientific disciplines. Through the Consultation and Research section of the AMD, Flanders and Miller thought that mathematicians would be called on to participate in collaborative research projects that required computers. But realizing

---

[5] Applied Mathematics Division Summary Report, Nov., 1956- June, 1958.

[6] An NSF study funded in 1953 on the status of applied mathematics in the United States noted that an "unprecedented mathematization" was occurring, "not only of fundamental scientific research in the physical and biological sciences but also in the management of our industrial and social systems." See F.J. Weyl, "Summary of Conference Discussions and Proposals: Panel Discussion; Opening Remarks," in *Proceedings First Conference Training Personnel for the Computing Machine Field* (Detroit: Wayne University Press, 1955), pp. 84-5.

this vision required more than convincing other scientists to work with mathematicians. It also required the creation of a new kind of professional mathematician: one who straddled the world of pure and applied mathematics, understood the nature of the problems arising in other scientific disciplines, and most importantly, understood the intricacies of computers. In short, this vision called for an elevated role for mathematicians in the conduct of science. And indeed, the AMD's early organization and attendant policies can be seen as an attempt to create an interdisciplinary research space in which computation was the centerpiece and mathematical experts played a more fundamental role in the advancement of science.

Bolstering the initial optimism of the AMD directors was the division's position within a multi-program laboratory conducting research in atomic science and engineering; it offered an particularly rich environment in which to explore new computer applications while in direct dialogue with a diverse scientific community. In its most optimistic conceptualization, it was hoped that the Applied Mathematics Division would provide an organizational framework in which to pursue cutting-edge research in both the science of computing and the application of computers to science.

In question, though, was what such an organization would look like? On the one hand, computing was highly interdisciplinary. The many facets of computing, from hardware design to the construction of mathematical models representing physical phenomena to the coding and processing of programs, called on the combined talents of scientists, engineers, mathematicians, programmers, and operators. It was therefore desirable to create an organizational structure that would facilitate collaboration *within* the division among members having different disciplinary backgrounds. At the same time, the service duties of the division meant that the organization would also have to collaborate actively with scientists from *outside* the division. Ideally, the AMD's organization would allow for a coordinated attack on an entire technological front including the design of components, subsystems, mathematical techniques, programming languages, and systems engineering while, most importantly, working on the real-life problems of scientists and engineers.[7]

For a variety of reasons, collaboration within the division was easier to achieve. As established in 1956, the AMD's organization was based on a division of labor intended to facilitate the management, operation, and application of computers. Computing was very much a social activity. The AMD needed programmers to write the software, key punchers to prepare the punched-cards for the computers, operators to run the machines, and a maintenance staff to keep them running. These different tasks are evident in the AMD's 1958 organizational chart which lists sections for Mathematical Consultation and Research, Programming Research and Development, Applied Programming, Computer Engineering, and Digital Machine Operations.[8]

---

[7] For a full discussion of the development of technological systems, see the seminal work Thomas P. Hughes, Networks of Power: Electrification in Western Society, 1880-1930 (Baltimore: Johns Hopkins University Press, 1983). Broad initiatives in the development and use of computers continues to push the entire field. As Alex Roland has shown, the DARPA funded Strategic Computing program of the 1980s was a similar effort to advance computer technology along an entire front. See Alex Roland and Philip Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993 (Cambridge: The MIT Press, 2002). There is also a large body of scholarly research on how users shape technologies and technological systems. The classic examples is Wiebe E. Bijker, Thomas P. Hughes, and Trevor F. Pinch, eds., The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology (Cambridge: The MIT Press, 1987). For a more recent study, see Oudshoorn and Pinch, eds., How Users Matter: The Co-Construction of Users and Technology.

[8] Nathan L. Ensmenger, From "Black Art" to Industrial Discipline: The Software Crisis and the Management of Programmers (Ph.D. diss., University of Pennsylvania, 2001); Thomas Haigh, "The Chromium-Plated Tabulator: Institutionalizing an Electronic Revolution, 1954-1958," IEEE Annals of the History of Computing 42, no. 4 (2001); Arthur L. Norberg, "High Technology Calculation in the Early Twentieth Century: Punched Card Machinery in Business and Government," Technology and Culture 31 (1990). Applied Mathematics Division Summary Report, July 1, 1958 through June 30, 1959. ANL, 1959. AEC Research and Development Report, ANL-6090 Mathematics and Computers, (TID-4500), 15th Ed. MCS archives. The AMD was also in charge of programming and operating an analog computer facility for the laboratory. However, an analysis of this activity is outside the scope of this paper.

The more difficult task facing the first directors of the Applied Math Division was how best to facilitate the use of computers by scientists outside the division. Some scientists and engineers would seek out the division for help with their work, especially from the mathematical disciplines like physics. However, this was too small a group around which to build a strong computational service. Donald Flanders and William Miller, the first two directors of the AMD, were intensely interested in attracting work from a broad range of disciplines, including biology, chemistry, metallurgy, and medicine. One way to get other disciplines involved in computation was to "sell" the potential of computers to model physical and biological phenomena. Behind the sales pitch though, was their conviction that the computer could be an indispensable research tool for scientists and engineers and would contribute to advancements in both theory and experiment. Successful computer simulations, though fairly rudimentary, suggested that these new engines of computation could inspire new approaches to problem formulation and testing.[9] Despite the limitations of 1950 computers, it was possible to imagine a future in which the investigation of physical and biological phenomena might be approached *almost entirely through computation*. No field would be left untouched; not even the life sciences, which had traditionally eschewed most mathematics outside of statistics.[10]

Selling computational services to scientists at Argonne was only half the battle. To a certain extent, computer specialists within the AMD believed in the need for an accompanying reorientation of scientific and engineering practices, such that experts in mathematics played a more central role in the analysis and solution of problems. I

---

[9] In one classic example, the Fermi-Ulam-Pasta experiment of 1953, a particular computer simulation of a vibrating, non-linear string was allowed to run past its usual stop point due to a heated debate, thereby revealing unexpected characteristics of non-linear systems. See Herbert L. Anderson, "Metropolis, Monte Carlo, and the Maniac," Los Alamos Science no. 14, no. Fall (1986): pp. 104-5.

[10] Joe November, "Dendral: Automating Hypothesis Formation," (Princeton University, 2004).

contend that the AMD's organization and attendant division policies were attempts to create this kind of interdisciplinary research space. Significantly, this early conception of computational science was, above all, collaborative and interdisciplinary; computers would be a technological bridge-builder between disciplines.

Beyond establishing collaboration between different groups of people, computational science also required the development and diffusion of "enabling technologies", especially mathematical tools and techniques to model phenomena in such a way that they could be handled by computers. Aggregated over time, these would provide the material and intellectual foundations for conducting scientific or engineering investigations on the computer. Here, again, mathematical expertise would be crucial to the success of computational science. If implemented well, collaboration and technical advancement would thus be a self-reinforcing process, further encouraging computer use among Argonne's scientists and engineers.

In terms of influencing interdisciplinary collaboration, the AMD's early monopoly of the laboratory's computing equipment provided the leverage needed to position mathematicians directly into the formative stages of scientific problem solving at Argonne. As a service bureau, the AMD offered a desirable *product* to other scientists at the lab, namely computer output. The computer was the bait. However, use of the division's computers required scientists to participate in a particular *process* necessary to translate their questions into a mathematical form computers could understand. It was at this translational juncture, from physical or biological problem to mathematical model to computation, that the mathematician's expertise would be mobilized. In establishing this arrangement, the directors of the AMD sought to define an agenda for computing that would attract top mathematicians.[11] In effect, the organization of the AMD was an effort to institutionalize a particular methodology for the conduct of science that emphasized a mathematical and cross-disciplinary approach to problem solving. But, as the research here shows, over the next decade this planned interdisciplinary research space became contested terrain. On the one hand, producers of computational tools sought their own disciplinary identity. On the other hand, computer users demanded improved computational service. Further complicating the issue was the rapid pace of innovation in technology which eroded the AMD's monopoly on computers. By 1970, the notion that interdisciplinary collaboration would be fostered by the AMD was supplanted by a new vision that saw computing as a scientific enterprise in its own right and independent of other scientists.

Interestingly, my research suggests a strong connection between early conceptions of computational science at the AMD and more widespread concerns about the professional status of applied mathematicians following World War II. The invention of the computer seemed to herald new opportunities for applied mathematicians to invigorate their profession and establish their own research agenda. Consequently, this chapter spends a good deal of time looking at the activities of the mathematicians within the Consulting and Research Section of the AMD because it is in their interactions, or lack thereof, with scientists from other disciplines that the connections between the professional aspirations of applied mathematicians and new computing technologies become apparent.

[11] Historian Michael Mahoney discusses the search for a disciplinary agenda to guide computer science. It would be interesting to apply some of his ideas in the context of computational science. See Michael S. Mahoney, "Computer Science: The Search for Mathematical Theory," in Science in the Twentieth Century, ed. John Krige and Dominique Pestre (Amsterdam B.V.: Harwood Academic Publishers, 1997).

New Opportunities for Applied Mathematicians

Trying to unravel the beginnings of computational science is, at best, a difficult enterprise. Computational science did not even achieve widespread recognition as a unique and valid approach to scientific inquiry until the mid-1980s.[12] The vision of computational science proposed by Flanders and Miller hinged on their being able to integrate mathematicians into interdisciplinary work. Success would depend in part on the revitalization of applied mathematics, a field that was, at best, anemic in the United States. For a variety of reasons, applied mathematics in the United States languished behind pure mathematics in both reputation and pedagogical emphasis. Applied mathematicians were generally considered second-class mathematicians, suited to service work but somehow lacking the intellectual rigor required of pure mathematicians. Even the notion that mathematicians could contribute significantly to cross-discipline, team-oriented scientific investigations was relatively new, emerging only in the twentieth century. While it was clear that *mathematics* was integral to most scientific and engineering disciplines, the role of the professional mathematician was not as clear. This odd position -- that mathematics was vital to the advancement of science, but mathematicians less so -- was in part a result of particular dynamics within the mathematical community itself. Increasingly, in the nineteenth century, professional mathematics was divided into two camps: *pure* and *applied*. Pure mathematics was that which was carried out by the academic mathematician and it implied, above all else, *autonomy* of mathematical research. Pure mathematicians, whose lineage extended as far back as the ancient Greeks, saw themselves as independent of the processes of research in

[12] Robert Pool, "The Third Branch of Science Debuts," Science 256, no. 3 (1992), Kenneth G. Wilson, "Grand Challenges to Computational Science," (Cornell Center for Theory and Simulation in Science and Engineering: 1987).

other scientific disciplines, and not bound by the rhythms of those disciplines.[13] Instead, pure mathematicians were interested in deep, penetrating questions that addressed the foundations of mathematical theories.[14] Applications might *emerge* from the work of pure mathematicians, but it was left to researchers in the sciences or engineering to determine the proper applications.

Interestingly, these distinctions between pure and applied are modern notions that began to appear only in the mid-nineteenth century. In the eighteenth century, as Lorraine Daston has shown, philosophers and mathematicians saw a different relationship between mathematics and the sensory world. During this period, *mixed* mathematics -- those that "mixed" physical and abstract properties -- dominated most mathematical research. As eighteenth century philosophers of mathematics like d'Alembert reasoned, even "abstract" mathematical analysis began with the concrete, and then property by property the features of reality were stripped away until only the barest skeleton of the original problem was left. This skeletal framework, although "abstracted" from reality, was nonetheless still tied to sensory experience and the concrete world.[15]

This conceptual difference is important; to the eighteenth and early nineteenth century mathematician, mathematics was "mixed" *with* its subject matter rather than being "applied" to it. Moreover, abstract (or pure) mathematics did not exist in a vacuum, but was as likely to be informed by insights into the physical world as the other way around. Thus, in both mixed and pure math, there was a connection to the experiential world.

---

[13] G.E.R. Lloyd, Early Greek Science: Thales to Aristotle (New York: W.W. Norton & Co., 1970).
[14] Browder, "Mathematics and the Sciences," p. 285.
[15] Lorraine J. Daston, "Fitting Numbers to the World: The Case of Probability Theory," in History and Philosophy of Modern Mathematics, ed. William Aspray and Philip Kitcher (Minneapolis: University of Minnesota Press, 1988), pp. 221-3, I. Grattan-Guinness, "Modes and Manners of Applied Mathematics: The Case of Mechanics," in The History of Modern Mathematics: Institutions and Applications, ed. David E. Rowe and John McCleary (Boston: Academic Press, 1989), p. 109.

While the mixed approach dominated early mathematical research, one doesn't get the impression that this corresponded to it being more highly regarded than pure math. In fact, it seems that quite the opposite occurred. Despite the prominence of mixed mathematics, the pure was still revered by mathematicians for its seeming clarity and higher standards of evidence, unspoiled as it was by the uncertainty that characterized application areas like physics. Indeed, there was a strong conviction among mathematicians of the time that it was only in the domain of the pure that real, inexorable progress in mathematics could be made.[16]

Throughout the nineteenth century there was a marked shift in the relative status of pure and applied mathematics that corresponded to the professionalization of the field after 1800. For example, in the French mathematical community, those mathematicians doing work in analysis and theory (pure) were usually the better mathematicians and tended to hold a greater proportion of teaching posts than their colleagues who worked in the field of mechanics (applied).[17] This is not to say that applied mathematics was abandoned by Europeans. Yet by mid-century the conception of "mixed" mathematics had been supplanted by the term "applied" and given a decidedly second-class status, below that of pure mathematics.

Similar patterns of development occurred in England and Germany, which, along with France, dominated mathematics in the nineteenth century. By the latter part of the century however, the line between pure and applied mathematics had blurred yet again as a number of scholars began to challenge the neohumanist tradition of mathematics that emphasized disciplinary purism.

---

[16] Daston, "Fitting Numbers to the World: The Case of Probability Theory," p. 223-4.
[17] Grattan-Guinness, "Modes and Manners of Applied Mathematics: The Case of Mechanics," pp. 110-13.

One mathematician in particular, the German, David Hilbert, had a powerful influence on the direction of mathematical research in the twentieth century. Hilbert worked or followed the developments in almost every field of mathematics. His own contributions were enormous, including the development of techniques to describe, in abstract terms, the structure of certain invariant systems, and his work in laying the foundations for algebraic geometry and homological algebra which would be used a half century later. He also did extensive work in mathematical physics, where he developed his ideas of Hilbert spaces and spectral theory which quickly became widely used tools by physicists. His groundbreaking work, Foundations of Geometry, published in 1899, solidified what became known as the axiomatic method, which sought to boil problems down into a set of statements whose validity seemed self-evident or were obtained in a clear, logical manner from other, seemingly self-evident statements. In Foundations, Hilbert reclassified the axioms of geometry so that it became an abstract science, detached from its earlier attachment to physical reality. While one could still think in terms of points, lines, and planes, as had geometricians since the days of Euclid, he argued that one could just as easily substitute "chair" or "table" for these terms. To him, all that mattered was the logical dependence between entities and their compatibility with one another. Hilbert's approach to geometry became an ideal for all mathematics as, increasingly, mathematicians were trained to *think axiomatically.*[18]

While it might seem that the very process of axiomitization prefaced mathematics was developed independently of any external connection to the real world, Hilbert argued that his mathematical approach was intimately applicable to real-world problems (although this was generally left to others). He believed that mathematics was "the

[18] Amy Dahan Dalmedico, "Mathematics in the Twentieth Century," in Science in the Twentieth Century, ed. John Krige and Dominique Pestre (Amsterdam B.V.: Harwood Academic Publishers, 1997), p. 653, Constance Reid, Hilbert-Courant (New York: Springer-Verlag, 1986).

foundation of all exact knowledge of natural phenomena" and viewed the continued split between pure and applied mathematics as potentially threatening to the vitality of the discipline.[19] Hilbert's structuralist approach, rooted as it was in axiomatics, set theory, and modern algebra, was an attempt to insure both the continued development and autonomy of mathematicians. To him the axiomitization and abstraction of mathematical theories served to extend their domain of application by uncovering unifying principles in different scientific disciplines. In short, the more "purified" the mathematical tool, the greater its potential for application.[20] The one disjunction between pure and applied, however, was in the origination of problems. For Hilbert, mathematicians created their own problems rather than drawing them directly from the sciences. In this way, the separation between pure and applied, as well as the autonomy of the mathematician, was perpetuated and reinforced.[21]

From his position as a professor at the University of Göttingen, Hilbert was able to attract and train mathematicians from around the world. Göttingen already had a strong mathematical and scientific tradition going back to 1795, when Carl Friedrich Gauss, the son of a canal tender and bricklayer, enrolled as the protégé of the Duke of

---

[19] David E. Rowe, "Klein, Hilbert, and the Göttingen Mathematical Tradition," Osiris 2nd series, no. 5 (1989).
[20] Dalmedico, "Mathematics in the Twentieth Century," p. 655.
[21] There were other mathematical traditions which offered an alternative to Hilbert's program. Most notable is that proffered by the French mathematician Henry Poincarè. Working at the same time as Hilbert, his main contributions were to revitalize classical nineteenth century mathematical subjects, such as the theory of functions, differential equations, and celestial mechanics, by introducing new conceptual tools, especially the notion of the *group* in his analyses. He also invented *qualitative methods* for the study of dynamic systems which allowed one to look for mathematical solutions to problems that were not quantifiable. In his choice of subject matter, Poincarè reflected the eighteenth century idea of mixed mathematic, interested as he was in "problems which presented themselves and not those which one posed oneself." By this he meant that the more interesting problems for mathematicians were those that emerged from the natural sciences rather than those concocted by the mathematician himself which were the norm. Yet, despite Poincarè's achievements, he worked alone and did not train students. Furthermore, many of his able colleagues, while admiring him and his connection to real-world applications, were reluctant to risk exploring the new fields he had opened. The onset of World War I, and the ensuing decimation of the French mathematical community (including the loss of almost half the students of the Ecole Normale Supèrieure) further reduced his influence (and the French influence) on future mathematicians. See Ibid., pp. 652-3.

Brunswick. By the time Gauss left Göttingen at age 21 he had nearly completed one of the great masterpieces of number theory and of mathematics, the Disquistiones Arithmeticae. After returning to the university a few years later as the director of their observatory, Gauss spent the remainder of his life making substantial contributions to every part of pure and applied mathematics.[22]

One hundred years later, Hilbert continued this tradition, joining Felix Kline in a mathematical program that was tightly integrated with the sciences. Together they turned Göttingen into a world-class center for mathematical training. Their program, which in 1892 had ninety students enrolled in the mathematical sciences, had grown to almost 800 by 1914. Moreover, under their influence the university established research institutes in physics, applied mathematics and mechanics, electrotechnology and geophysics -- all closely aligned with the mathematics department. Yet the placement of applied mathematics alongside mechanics and not within the math department says much about how the field was regarded by “pure” mathematicians.

The Göttingen program also attracted some of the most promising American mathematicians, who absorbed Hilbert’s approach and transferred it to the United States. For the most part, the U.S. was a late bloomer in mathematics; not until the turn of the century did Americans begin to make significant contributions to the field. Absorbing much of Hilbert’s axiomatic approach, the American mathematical community was almost entirely dominated by its emphasis on abstraction.[23] Its leaders, George D.

[22] Reid, Hilbert-Courant, pp. 47-8.

[23] This had not always been the case. Prior to 1876, practical interests had dominated the attention of American mathematicians; however, in that year Daniel Gilman was able to lure the British mathematician J.J. Sylvester to Baltimore to establish a program in pure mathematics at Johns Hopkins University. The success of this and similar programs at Chicago, Harvard, and Princeton reoriented American mathematics away from applications. See Larry Owens, "Mathematicians at War: Warren Weaver and the Applied Mathematics Panel, 1942-1945," in The History of Modern Mathematics: Institutions and Applications, ed. David E. Rowe and John McCleary (Boston: Academic Press, 1989), p. 297.

Birkoff at Harvard and Oswald Veblen at Princeton, presided over an academic mathematical community that generally eschewed applications, despite the rising strength of American industry and its growing need for mathematical expertise. While there were notable exceptions to this, namely Vannevar Bush, who did brilliant theoretical work on electric circuits and the transmission of energy, and Warren Weaver, who co-authored a book on electromagnetic fields, overall there was little interaction between mathematicians and industry.[24]

The American entry into World War I in 1917 provided a strong impetus for the development of applied mathematics as many mathematicians supported the war effort. Ballistics was an area of special concern as the type, range, and behavior of military ordnance advanced rapidly, generating a need for new firing tables. Forest Moulton, an astronomer from the University of Chicago with a penchant for using numerical techniques in his calculations, organized a research team for the Office of the Army Chief of Ordnance to apply these tools to ballistics problems. Likewise, Oswald Veblen gathered another thirty mathematicians at the Army's Aberdeen Proving Ground to work on military problems. This successful coupling of mathematics and applications during the war encouraged companies like GE, RCA, and Bell Telephone Laboratories to hire mathematicians in the 1920s to assist their engineers as they struggled to improve the speed and reliability of electric and communication networks.[25]

The 1920s also saw several unsuccessful initiatives to create mathematical centers dedicated to applied work. At Princeton, Oswald Veblen (who was a great admirer of the Göttingen program) was unable to interest either the National Research Council or the General Education Board of the Rockefeller Foundation to fund an Institute of Applied

---

[24] Dalmedico, p. 656. See also Atsushi Akera, Calculating a Natural World: Scientists, Engineers, and Computers in the United States, 1937-1968 (Dissertation, University of Pennsylvania, 1998), pp. 161-63.
[25] Akera, Calculating a Natural World, pp. 163-67.

Mathematics, a discipline he felt was woefully neglected in America.[26] Even on the eve of World War II, applied mathematics was still being perceived as “something less attractive and worthy” than pure mathematics. The American mathematical profession reflected this point; since 1900 only one applied mathematician had been elected to the presidency of the American Mathematical Society, though before that almost all the presidents had been engaged in applications.[27]

It wasn’t until World War II that the professional aspirations of applied mathematicians began to look more promising. By 1939, military and scientific leaders in the United States recognized that the development and deployment of ever more complex weapons systems required considerable mathematical expertise. For mathematicians, the war seemed to offer possibilities for greater post-war influence in the development and conduct of science in America.[28] In 1940, Vannevar Bush organized a group of mathematicians to do military research under the auspices of the National Defense Research Council (NDRC) a subgroup of the Office of Scientific Research and Development (OSRD). Initially, the NDRC was conceived as an interdisciplinary organization wherein scientists, engineers, and mathematicians could work together as equals to solve problems. Most of the math and computing work was placed under section D-2 (later section 7) and was directed to improve fire and control systems for anti-aircraft guns.[29] However, the interdisciplinary collaboration that was envisaged failed to materialize and increasingly mathematicians were placed in the role of providing

[26] William Aspray, "The Emergence of Princeton as a World Center for Mathematical Research, 1896-1939," in History and Philosophy of Modern Mathematics, ed. William Aspray and Philip Kitcher, Minnesota Studies in the Philosophy of Science (Minneapolis: University of Minnesota Press, 1988), pp. 349-53.

[27] R.G.D. Richardson, "Applied Mathematics and the Present Crisis," American Mathematical Monthly v. 50 (1943).

[28] Owens, "Mathematicians at War: Warren Weaver and the Applied Mathematics Panel, 1942-1945," p. 287.

[29] Akera, Calculating a Natural World, p. 187.

computational services to scientists and engineers instead of directing or evaluating their research activities.[30]

In 1942, in response to the tensions within the NDRC, Bush, who now headed the OSRD itself, created a separate entity, the Applied Mathematics Panel (AMP), to coordinate the services of mathematicians and to serve as a clearinghouse for mathematical information useful to the war effort. Headed by the mathematician Warren Weaver, the AMP employed almost three hundred people over the next two and a half years, including such eminent mathematicians as John von Neumann, Richard Courant, Jerzy Neyman, Garrett Birkhoff, and Oswald Veblen. The panel supported work in applied mathematics, especially in the development of statistics, numerical analysis and computation, the theory of shock waves, and operations research. More importantly, the AMP actively promoted the institutionalization of applied mathematics by supporting programs at Brown, Berkeley, Columbia, and NYU.[31]

In retrospect, the accomplishments of the AMP were uneven. While the panel was quite successful in establishing applied mathematics research at institutes across the country, their efforts to coordinate war-time mathematics foundered, in part due to mathematicians themselves. For example, Weaver enlisted the gifted MIT mathematician Norbert Wiener, but got little in return. According to one coworker, Wiener was completely uninterested in problem-solving,

> “simply on the basis of utility, particularly if it lacks the qualities suggestive of an elegant, general, formal solution -- as do so many of the problems confronting the nuts-and-bolts realist. He simply will not hammer out an inelegant though adequate solution with the cheap-and-nasty tools of the everyday applied mathematician.”[32]

[30] Ibid., p. 190.
[31] Owens, "Mathematicians at War," pp. 287-88.
[32] Quoted in Owens., p. 291.

The perceived resistance of some mathematicians to the abandonment of their disciplinary adherence to purity led to some public criticism during the war. In 1943, Time Magazine reviewed a recent book by the popular science writer George Gray in which he argued that the contributions of mathematicians to ballistics, aerodynamics, optics, acoustics, and electronics were made *in spite of* a tendency of powerful mathematicians to denigrate applications as unworthy of their attention. This failure of top-flight mathematicians to sully their hands in applied work was, according to Gray, hampering war efforts.[33]

Such public criticisms stimulated a fierce rejoinder from the mathematical community, but it also rekindled an earlier debate between the American Mathematical Society (AMS) and the OSRD as to what kind of mathematician was suited to the applied nature of war work. In 1942, leaders within the AMS had drafted a memo entitled "Mathematics in War" in which they suggested that the mathematical talents of the nation could best be mobilized by appointing a suitably qualified mathematician to evaluate defense programs and then select competent colleagues to assist in the work. To the AMS, such a qualified mathematician would be, above all, one engaged in "pure" work. Noting that mathematics had been instrumental in the discovery of natural laws and the mastery of nature, the AMS report went further, stating that these gains had been achieved "only through the skillful application of pure mathematics developed without reference to the immediate needs of physics or engineering."[34]

In practice, the leaders of the OSRD, (siding with the engineers and physicists), had a hard time understanding how practical and pressing problems could be solved effectively by mathematicians working in such rarefied air. Thus, when it came time to

---

[33] Ibid., p. 293.
[34] Quoted in Owens., p. 294.

select members for the Applied Mathematics Panel, Weaver had to decide whether the war effort would best be served by a mathematician who was an unselfish team player, comfortable with military personnel, and current with weapons development, or as he so eloquently put it, a "prima donna, a-social genius," who was convinced that their ideas were "transmitted to him by Almighty God."[35]

Weaver himself believed that pure and applied mathematicians were a breed apart and that it was not easy for one to be converted to the other. While he had great respect for the arguments of the AMS to select a pure mathematician, he felt the applied mathematician's training and qualities of character were more useful to the current war effort. The training of the applied mathematician, Weaver mused, instilled in them an attitude for service that was absent from the pure mathematician's training or disposition. For the AMS, this decision was a blow to their hopes for an elevated status in the conduct of science and technology both during and after the war. As the leaders of the AMS realized, the failure of the American mathematical profession to train applied mathematicians left it up to disciplines like physics and engineering, as well as European mathematicians forced out by the Nazis, to provide the mathematical skills for war work.[36] Rebuffed by Weaver, the AMS feared that the increased emphasis on applied mathematics would lead to a reorientation of mathematical training after the war to the detriment of research in pure mathematics.

If heightened prestige and influence for the mathematical community failed to materialize during and immediately after World War II, at least there was increased respect for the contributions that applied mathematics could make to the development of new technologies. The dawn of the nuclear age and the ascendancy of atomic physicists

[35] Ibid., pp. 294-96.
[36] Ibid., pp. 298-300.

in the new scientific world order suggested that there would be plenty of opportunities for applied mathematicians to contribute to scientific and engineering research.[37] Most significantly, the war years produced a technology that seemed to herald new possibilities for collaboration between mathematicians (both pure and applied) and scientists – that of the electronic digital computer.

Computers provided the obvious and key technology around which applied mathematicians could become involved in multi-disciplinary work. By its very nature, computing attracted specialists from a variety of disciplines. Logicians and control engineers were interested in aspects of computer architecture; physicists and electrical engineers contributed to component and hardware design; mathematicians became interested in software and numerical methods; and computer theory drew adherents from math, psychology, and even various biomedical disciplines.[38] However, in the early days of the electronic digital computer, few people recognized its inherent complexity or the extent to which computing would require the mobilization of all these different skills.

Given the wide variety of skills needed to build, operate, and apply computers, it might not be surprising that extremely fast engines of computation were initially built to assist researchers working in interdisciplinary settings. The Manhattan Project, for example, had immense computational requirements that were beyond the scope of what traditional hand computing techniques could reasonably manage, even when conducted by human "computer" teams.[39] Although the first electronic digital computer, the ENIAC, was not completed until near the end of the war, its usefulness to researchers in the military was manifest from the moment it was used to establish the feasibility of a

---

[37] See Daniel J. Kevles, The Physicists: The History of a Scientific Community in Modern America (New York: Alfred A. Knopf, 1978). especially chapters 20-22
[38] William Aspray, John Von Neumann and the Origins of Modern Computing, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge: The MIT Press, 1990), p. 3.
[39] Jennifer Light, "When Computers Were Women," Technology and Culture 40, no. 3 (1999).

thermonuclear device in late 1945. As Stan Ulam, one of the scientists at Los Alamos who ran preliminary simulations of the hydrogen bomb on the ENIAC recounted:

> “One could hardly exaggerate the psychological importance of this work and the influence of these results on Teller himself and on the people in the Los Alamos laboratory in general . . . I well remember the spirit of exploration and of belief in the possibility of getting trustworthy answers in the future. This partly because of the existence of computing machines which could perform much more detailed analysis and modeling of physical problems.”[40]

Ulam, in his excited recounting, touched on a (perhaps *the*) critical paradox of scientific computing and later computational science as it would develop: high-speed computing, from its inception, was driven as much by being a solution to a problem as it was *a solution in search of a problem*. John von Neumann, perhaps the most influential person associated with the early years of computing, seemed attuned to this duality when he stated that “the performance of the computer is to be judged by the contribution which it will make in solving problems of new types and in developing new methods,” and suggested that "a computing machine of extremely high speed will probably be used to treat problems *which nobody considered computing problems before*."[41] (my italics)

For applied mathematicians surveying new professional opportunities after the war, the computer provided obvious opportunities for research in interdisciplinary settings. How to institutionalize such an arrangement, though, remained unclear.

The Social Organization of Computing at Argonne

Following on the success of the ENIAC, a number of computing projects were initiated in the late 1940s geared toward improving the speed, reliability, and

[40] Quoted in Aspray, John Von Neumann, p. 47.
[41] Ibid., p. 55 and p. 61.

functionality of electronic digital computers.[42] The newly formed Atomic Energy Commission had a pressing need for computers to further their mission to develop atomic weapons and power reactors; unsure as to whether a commercial market to develop computers would emerge, the AEC instead decided to build their own machines in-house. At Argonne National Laboratory, a Computer Group was established in 1949 to build a computer to assist researchers at the Lab in science and engineering problems related to the atomic sciences and the design of nuclear reactors. The AVIDAC (Argonne's Version of the Institute's Digital Automatic Computer) was finished in 1953 and was immediately applied to a wide variety of such problems. Support for mathematics, however, was not seen as especially important to people outside of the computer project.[43] It seemed as if lab management saw the real challenge as building the machine, while its programming was an afterthought.

In one respect, however, the construction of the AVIDAC had a direct effect on the organization of mathematicians at Argonne; its completion initiated a process of consolidation among the two computing groups at Argonne as a more cohesive organization was required to maintain, program, and apply the computer to the mission of the lab.[44] By early 1956, the newly unified computing group had over fifty staff members and was organized into four sections including mathematical analysis, digital

---

[42] See Chapter 1, "Building Big Iron" or Akera, Calculating a Natural World; Aspray, John Von Neumann, Martin Campbell-Kelly and William Aspray, Computer: A History of the Information Machine, The Sloan Technology Series (New York: Basic Books, 1996); Paul E. Ceruzzi, A History of Modern Computing (Cambridge: The MIT Press, 1999); J. H. Curtiss. The Institute for Numerical Analysis of the National Bureau of Standards. Washington, D.C.: Office of Naval Research, Department of the Navy, 1951. box 3, COHC #196, Series 2, Subseries C, American History Museum, Smithsonian Institute; Kenneth Flamm, Creating the Computer: Government, Industry, and High Technology (Washington, D.C.: The Brookings Institution, 1988); Herman H. Goldstine, The Computer from Pascal to Von Neumann (Princeton: Princeton University Press, 1972); David E. Lundstrom, A Few Good Men from Univac.

[43] Argonne National Laboratory Applied Mathematics Division: A Synopsis of Long Range Plans. Argonne National Laboratory, 1975. Folder 2, Box 53, AMD Reports 1965-1975, Records of the Argonne University Association, University of Illinois, Urbana-Champaign Special Collections.

[44] The two computing sections had been assigned to the physics and reactor engineering divisions respectively.

programming, digital machine operations, and analog problem analysis, programming, and machine operation.[45]

Donald Flanders, affectionately called "Moll," had been the driving force behind the development of the AVIDAC and served as the first director of the computing unit. The son of a prominent liberal senator from Vermont, Flanders is an interesting character. After receiving a Ph.D. in mathematics at the University of Pennsylvania, he spent the 1930s as an assistant professor of mathematics on the then fairly undistinguished faculty of New York University. Although he was an ardent researcher in his field of topology (considered applied mathematics), he had made little progress in his work. Having spent time as a national Research Fellow at Princeton, a frustrated Flanders inquired of the eminent Princeton mathematician Oswald Veblen as to the possibility of attracting better mathematicians to the NYU department.

The deteriorating situation in Germany proved to be a boon for both Flanders and NYU. The highly-talented mathematician Richard Courant had recently been removed by the Nazi regime from his post at Göttingen where he had worked for years alongside Hilbert. The timing couldn't have been better for all concerned and on the suggestion of Veblen; Flanders was able to pull enough strings to attract Courant to the NYU faculty. It was a coup. In one stroke, Flanders succeeded in bringing to the United State a mathematician of high caliber and one whose interests tended towards applied mathematics.[46]

---

[45] Applied Mathematics Division Summary Report, Nov., 1956- June, 1958. By way of comparison, the Argonne scientific and engineering staff in 1956 numbered 621.
[46] Reid, Hilbert-Courant. During the war Warren Weaver's Applied Mathematics Panel awarded several research contracts to Courant's group at NYU to do research on underwater explosions. Furthermore, the close relationship Flanders developed with Courant paid dividends as Courant later served as one of the primary outside mathematical consultants to the AMD.

Flanders' administrative and mathematical skills gained enough recognition that he was recruited by Hans Bethe, through Richard Courant, to organize a computing section at Los Alamos to assist scientists working on the Manhattan Project.[47] While in New Mexico he gained enormous experience both in terms of how a computational operation might be organized and in how scientists and applied mathematicians could work together. As if anticipating the future, he also spent his spare time developing methods by which binary logic might be used by digital computing devices to perform arithmetic.[48] After the war, Flanders brought these insights to his work at Argonne and sought to apply them through the creation in 1956 of a mathematical laboratory organized around digital computation.

That the computing program would eventually have to be spun off as its own division was evident almost as soon as the AVIDAC was completed in 1953. Housed within Physics Building 203, it was clear that the computer and attendant staff would quickly tax the available space. The AVIDAC itself was installed in the basement while the computing staff -- including the mathematicians, programmers, and administrators -- was forced to share offices. The impending arrival of the IBM 704 in 1957, plus the construction of the Lab's experimental computer GEORGE, further strained the capacity of building 203, making it clear that further expansion of the computing program would require a separate facility.[49]

---

[47] Aspray, John Von Neumann, p. 29.. Also see Donald Flanders' testimony before the House Un-American Activities Committee for the Alger Hiss case.

[48] See Henry Tropp. Interview with Margaret Butler, Jim Butler, Dave Jacobsohn, Charles Harrison, Claire Kilty, Burt Garbow, Stan Zawadzki, Bob Kroupa, Franz Morehouse, and Wallace Givens. 1972. box 1, COHC #196, Series 1: Transcripts, subseries B: Research Transcripts, American History Museum, Smithsonian Institute.

[49] Budgetary constraints at Argonne, however, delayed the planning of a new building to house the AMD until mid 1961. W. B. Harrell, to Kenneth A. Dunbar. "New Building for AMD," May 1, 1962. ANL Construction Projects, 1961-64, Box B93-00150: ANL Administration/Appointments, MCS Archives, Argonne National Laboratory.

While the lack of space for the computer operation provided one impetus for the creation of a separate AMD, its widening scope of activities at the lab provided the more immediate pressure for reorganization. By 1956, scientists from every division at Argonne were requesting computer services and the computing section was having a hard time meeting the demand.

Although it is difficult to document, sometime in 1954-1955, Donald Flanders began to agitate for the creation of a distinct Applied Mathematics Division. The lack of a clear paper trail also makes it difficult to say what his precise arguments to Argonne administrators were other than that such a move would improve service.[50] Whether it was made explicit or not, based on his experience at Los Alamos, it is reasonable to assume that Flanders envisioned the creation of a separate division focused on computation as an opportunity for mathematicians to become more involved in scientific research. Indeed, the preface to the first AMD summary report, cited above, reveals a particular vision of computing as an integrative technology around which collaborative interdisciplinary projects could be organized. It foregrounds mathematics and mathematical research and suggests that mathematicians could play an increasingly central role -- from problem formulation to its final solution -- in the scientific mission of the laboratory.[51]

Moll Flanders' desire to create a separate Applied Mathematics Division met a receptive audience both at Argonne and at its parent agency, the Atomic Energy Commission. In the most general terms, the inclusion of a strong applied mathematics program, alongside its physics program, was seen as a major contribution to the

---

[50] Donald A. Flanders, to Erwin H. Bareiss. Nov. 1, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

[51] Applied Mathematics Division Summary Report, Nov., 1956- June, 1958.

"scientific tone" of the laboratory. Thus, physics and applied mathematics produced certain synergies; a point not lost on the Argonne Review Committee which noted "strength in divisions are necessary if the Laboratory is to have enough coupling with the scientific community of the nation to be able to have a significant place in it."[52]

Within the ANL community, administrators were eager to improve the computational service operations and were inclined to support a reorganization that would further the mission of the laboratory.[53] Further bolstering Flanders' initiative was a new appreciation for the essential role of mathematics in the development and deployment of complex technological systems such as atomic weapons, particle accelerators, and reactors.   Flanders's own success in running the computing section at Los Alamos was a testament to the contribution that applied mathematicians could make to large interdisciplinary scientific projects. Now, in the context of the Cold War, mathematical expertise was going to be mobilized for a new purpose -- to improve the productivity and efficiency of America's scientists and engineers.[54]

Locked in a Cold War that was increasingly hot, American scientists were perceived to be in a battle against their Soviet counterparts with the fate of the world hanging in the balance. The implications for science were staggering. As historian Bill Leslie notes, Cold War politics blended with a prevalent belief that high technology conferred competitive advantages to its possessor. The pursuit of such technological

---

[52] H. H. Barschall, to R. W. Harrison. "Review Committee for Physics and Applied Mathematics Divisions," Nov. 10, 1960. B93-00147, ANL Review Committee for AMD, Argonne National Laboratory. That the national laboratories had been perceived as providing cutting edge research opportunities, technical equipment too expensive for most universities to acquire, and support services second to none were chief tenets of the entire AEC laboratory program. See Robert W. Seidel, "The Cold War and National Laboratories of the Atomic Energy Commission," in Second Conference on Laboratory History, ed. Catherine Westfall (Jefferson Laboratory, Newport News, Virginia: U.S. Department of Energy, Thomas Jefferson National Accelerator Facility, 2001), pp. 9-13.
[53] Interview with Margaret Butler by Author, 5-15-2002, Argonne National Laboratory.
[54] See chapter 2, Paul N. Edwards, The Closed World: Computers and the Politics of Discourse in Cold War America (Cambridge: The MIT Press, 1996).

superiority, he argues, resulted in a redefinition of the scientific landscape, especially in terms of its funding. After WWII, the Department of Defense quickly became the largest single supporter of American scientists, to the tune of some $5.5 billion a year by 1960.[55] In addition to this, by the mid 1960s the Atomic Energy Commission had spent another $4 billion since the war on research at the national lab system, with comparable sums invested in facilities and equipment. Other agencies, such as the National Science Foundation, the National Aeronautics and Space Administration, and the National Institute of Health also contributed hundreds of millions of dollars to American scientists during this period.

Insuring the highest possible rate of return per scientific man-year was of paramount importance to these funding agencies, not only because equipment such as computers and particle accelerators were expensive, but for ideological and national security purposes, too. A series of surveys conducted by the National Research Council (in the late 1940s and early 1950s) indicated that the United States was producing scientists and engineers at a slower rate than the Soviet Union. The unexpected launch of Sputnik, the first man-made satellite, by the Soviet Union on October 4, 1957 reinforced the perception that Americans were falling behind their communist counterparts in science and technology. In this environment of ideological and technological competition, scientists, engineers, and even mathematicians were considered crucial to the very survival of the Western world.

In looking at the "militarization" of science and scientists in the Cold War, most historians have focused on prominent groups such as physicists. What is less well known

[55] The DOD mainly supported scientists in the physical sciences and engineering, but they were also willing to fund programs in the natural and social sciences. See Stuart W. Leslie, The Cold War and American Science: The Military-Industrial-Academic Complex at MIT and Stanford (New York: Columbia University Press, 1993), pp.1-2.

is the extent to which groups of mathematicians were also organized as a ready reserve for military purposes. The creation of the Institute for Numerical Analysis, part of the National Bureau of Standard's National Applied Mathematics Laboratory (NAML) is one such example. Supported primarily by the Office of Naval Research, the mathematical research of the group was justified, in part, by its potential contribution to national defense. Written into the Prospectus of the organization was the statement that the NAML:

> "Should undertake to maintain a reservoir of personnel trained in applied mathematics which can be drawn on in case of a national emergency, and should at the same time develop disciplines and tools to facilitate the conversion of the nation's peace-time scientific manpower to emergency uses."[56]

It was with this backdrop that discussions about the role of the National Laboratory system took place.[57] The idea of having a reservoir of scientific talent for national emergencies was a key area of consideration in discussions of the role National Laboratories were to play on the national scientific scene. Such arguments were adopted and repeated many times, especially by those who clearly benefited from the labs. Argonne's Policy Advisory Board was no exception, justifying the lab's existence in terms of its contributions to national security:

> It is the importance of the existence of such going laboratories as a reserve force to be used in any appropriate way in the event of a grave national emergency. The existence of such staffs and facilities would be of obvious usefulness. It seems, furthermore, that the existence of close relationships with the scientific workers in the other laboratories of the region might be just as important in providing a point for rapid and efficient mobilization of scientific manpower and other facilities.[58]

---

[56] Curtiss. The Institute for Numerical Analysis of the National Bureau of Standards.

[57] Peter J. Westwick, The National Labs: Science in an American System, 1947-1974 (Cambridge: Harvard University Press, 2003).

[58] Louis A. Turner. The Role of the National Laboratories During the Next Decade. 1959. Policy Advisory Board Meeting, Oct. 21, 1959, Box 2, RG 326: Policy Advisory Board Meetings, General July 1957-Aug. 1960, GLFRCNARA.

If Sputnik was viewed as a demonstration of Soviet technical superiority, then the United States had to meet this challenge by improving its scientific and technical output. As research shows, increasingly computers were seen as a way to augment the productivity of the scientific workforce.

Keying on this sentiment, the Atomic Energy Commission's Computer Advisory Group (CAG) in 1961 suggested increased funding for mathematics and computer science research at the National Labs, noting that "in the struggle for scientific superiority a country producing scientists at a slower rate must ensure optimal productivity per scientific man-year by automation or any other means not leading to deterioration in the quality of research."[59] While the application of computers to science and engineering seemed to be one of the most promising avenues to pursue, effective use of computers required a tremendous amount of research, most especially in the development of mathematical techniques suited to these new engines of computation. This research, the CAG report suggested, would depend in no small measure on the contributions of applied mathematicians, working in conjunction with their counterparts from other scientific and engineering disciplines.

Flanders' efforts to sell the AMD to lab directors coincided with a sea change in the AEC's attitude toward computers in general. In 1955 the AEC had disbanded the original Computer Council, which included among others, John von Neumann, Edward Teller, and Nicholas Metropolis, concluding that its mission to advise the Commission on matters pertaining to the effective use of UNIVAC #4 (which was installed at NYU under the auspices of Richard Courant) had been completed. With its goal accomplished, T.H.

[59] John R. Pasta. Minutes of the September 28, 1961 Meeting of the Computer Advisory Group of the Atomic Energy Commission. Washington, D.C.: Atomic Energy Commission, 1961. AEC 553/17. 12, 1334, RG 326: Records of the Atomic Energy Commission, collection #20, NARA.

Johnson, the Director of the AEC's Division of Research informed Edward Teller at Livermore, "there no longer appears to be a need for an AEC Computer Council."[60]

With the dissolution of the first Computer Advisory Group, responsibility for formulating the AEC's computer policy was assigned to one person, John Pasta. Within four years Pasta was overwhelmed and he pleaded with the directors of the Division of Research to reconvene a new Computer Advisory Group as developments within the computing field were moving too rapidly for one person to manage.[61] Citing the "great increase in the use of high speed electronic computers in the Atomic Energy Program" and particularly "in the work sponsored by the Division of Research," the AEC reestablished the Advisory Board. The new panel was empowered to advise the AEC Director "on matters pertaining to the use of computers in research, the program of computer research, and associated applied mathematics studies."[62]

The specific inclusion of applied mathematics research indicates awareness at the funding level that the mere availability of a computer at a lab did not guarantee the gains in scientific and engineering productivity that had been promised.[63] Furthermore, the strong representation of applied mathematicians on the reconstituted Advisory Group further testifies to the recognition that improved mathematical techniques were crucial to unlocking the full potential of computers. Of its ten members, four were applied

---

[60] T. R. Johnson, to Edward Teller. "Dissolution of the AEC Computer Council," June 15, 1955. 11: Computer Council, 2229, von Neuman, J. AEC, DOE Archives, Germantown, MD. The other members of the original AEC Computer Council include: Robert Richtmyer, professor at the Courant Institute of Mathematical Sciences at NYU, Eleazer Bromberg, also a professor of Applied Mathematics at the Courant Institute; A.S. Householder head of the Mathematics Panel at Oak Ridge National Laboratory, Henry Hurwitz, Jr. (KAPL), and Bernard Spinrad, director of the physics division at Argonne National Laboratory.

[61] AEC 553/8 Establishment of a Computer Council to Advise the Division of Research. Washington, D.C.: Atomic Energy Commission, 1959. AEC 553/8. 12, 1334, 20, NARA.

[62] R.E. Hollingsworth, to Clinton P. Anderson. Jan. 21, 1960. 12, 1334, RG: 326 Records of the Atomic Energy Commission, # 20, NARA II.

[63] Dr. Mina Rees of the Office of Naval Research made this same point in a 1950 article in Science. See Mina Rees, "The Federal Computing Machine Program," Science 112, no. Dec. (1950): p.736.

mathematicians, two were physicists who used numerical techniques extensively in their research, three were engineers engaged in the design of computers, and one was a philosopher-turned-computer scientist.[64]

Given the composition of the committee, it is not surprising that their reports consistently emphasized that computers were a key to improving the productivity of scientists and engineers and that basic mathematical research was integral to its realization. One of the first reports by the Computer Advisory Group provided an inventory of the principle areas of computer utilization in the AEC research and development program. It is worth examining this document in some detail as the themes introduced above are well represented. In general, the CAG's 1960 report proceeds along two fronts: first it notes the near indispensability of computers to the work of AEC scientists and engineers in certain research areas, and second makes the corollary arguments that computers can improve the productivity of their users and allow for a more efficient use of capital equipment.

The report begins by noting that because of its unique program, the AEC spends more money for scientific computation than any other organization in the country. This expenditure, the Committee remarked, was justified "because it has contributed markedly to the speeding up of the transition of a theoretical advance into a technological one." In the case of weapons design, the 1960 report points out that the computation of multi-dimensional hydrodynamics problems involving neutronics were already taxing the limits of current computer technologies, but more significantly "the problems of simulating new

[64] Hollingsworth, to The new members were: Nicholas Metropolis (director of the Institute for Computer Research, U. Chicago); R.D. Richtmyer (Courant Institute, NYU); A. S. Householder (Head of Mathematics Panel at ORNL); Arthur Burks, (philosopher, computer scientist U. Michigan); J.H. Bigelow (Chief designer of the IAS computer, Institute for Advanced Study); W.H. Ware (engineer, Rand Corporation); Milton Rose (mathematician, Brookhaven National Laboratory); Martin Graham (engineer, Rice University); Maria Mayer (theoretical physicists, U. Chicago).

weapons designs and estimating their effects become especially important during a test ban since technical progress in these fields must rest much more heavily on theoretical studies." Thus, as I am arguing, computers had both technological and political dimensions.

The use of computer in reactor research and design, Argonne's primary mission, was second in size only to the AEC's weapons program. As with weapons design, problems involving three-dimensional calculations concerned with depletion studies were also pushing the current limits of machine computation. Keenly aware that budget-conscious AEC administrators might balk at the escalating cost of their computer budgets, the Advisory Group emphasized that expenditures on computer equipment had to be evaluated in a new light. The fact that it was cheaper to run a computer simulation than explode a bomb or build a reactor imbued simulations with an economic value that was difficult to translate directly into dollars and cents. "The cost of mathematical and simulation experiments that could be carried out has been trivial," they noted "when compared to the alternative of measurement with live models."

If simulations provided a somewhat intangible cost savings, the potential of computers to improve the efficiency of scientists was more concrete:

> "It is quite clear that automatic recording of experimental data on media that can communicate directly with a computer must save many hours for the scientist. The recording, checking and processing can be done promptly and the accumulated results can be available during the actual experiment. Such practice can often lead to more efficient use of very large accelerators, where at present very liberal amounts of data are collected in each experiment to preclude setting up the equipment again."

Careful not to oversell computers, the Advisory Group stressed that the nature of this technology was neither fully understood nor exploited fully. Moreover, they warned the AEC directors not to rely on industry to develop the next generation of computational tools and techniques. Rather, they argue, "progress…is most likely to take place in an environment where people using computers to their fullest capacity are in contact with people who are capable of understanding the present limitations of the machines, and are also capable of providing ideas and equipment for removing these limitations.”[65] Such an environment, they suggest, is to be found only within the National Laboratory system, as these labs can supply the problems, the talent, and most importantly can afford large, complex, and very expensive computer installations.

Back at Argonne, Flanders’ push to create a separate Applied Mathematics Division in 1954-55 appealed to these same sentiments: cost savings and improved scientific productivity. In the context of the Cold War and in light of the tremendous economic investment in computing by the AEC, Flanders’ proposal resonated at both the Lab Director’s level and at the AEC.

It should be noted that the idea for creating a separate applied mathematics unit focused on digital computers did not emerge from a vacuum. As early as the 1920s, automobile, aircraft, chemical and electrical equipment manufacturers generally employed a few applied mathematicians to assist engineers with the framing of difficult problems and convert them to a form suitable for processing on punched-card equipment. By the 1950s, industrial adopters of computers, such as Convair and General Electric, also established separate Computations Laboratories.[66] However, these computation

---

[65] Meeting of the Mathematics and Computer Sciences Research Advisory Committee with the GAC. Washington, D.C.: Atomic Energy Commission, 1964. AEC 553/24. 14, 1334, 20, NARA. The 1960 report by the Computer Advisory Group discussed here is attached as an appendix to a later, 1964 report by the Mathematics and Computer Sciences Research Advisory Group, the successor of the second CAG. Thus, the beginning and very end of the appendix includes some comments that reflect the 1964 perspective. In FY 1962, the AEC spent $15.2 million on computer equipment and a total of $37 million on operating expenses. The breakdowns for FY 1963 are $17.4 million and $43 million respectively, and for FY 1964 they are $21.7 million and $49 million respectively. Although I don’t have the numbers for 1960, I feel it is safe to make the claim that the AEC’s expenditures on computers and related activities for scientific research still outstripped other organizations in the country.

[66] H.R. J. Grosch, "The Computer Laboratory in Industry," in The Computing Laboratory in the University, ed. Preston C. Hammer (Madison: University of Wisconsin Press, 1957); H.S. Wolanski, "Applications of

departments operated exclusively as service bureaus to other divisions, meaning that the mathematicians on the staff were charged with finding "quick and dirty" solutions to problems rather than doing fundamental research in applied mathematics.

A closer approximation to the AMD model emerged from the Office of Naval Research (ONR) efforts to support work in numerical analysis related to von Neumann's computer project at the Institute for Advanced Study. Dr. Mina Rees, Director of the Mathematical Sciences Division of the ONR, strongly believed that the tremendous effort being put into the development of digital computers was not being adequately paralleled by theoretical investigation aimed at finding how best to use them.[67] With the IAS project underway, Rees used her position at the ONR to push for the creation of the aforementioned National Bureau of Standards' Institute for Numerical Analysis in 1947. Situated on the campus of the University of California at Los Angeles, it was one of four branches of the National Applied Mathematics Laboratories, the other three being separate laboratories for Computation, Statistical Engineering, and Machine Development, all located in Washington D.C.[68]

In terms of a model for the organization of the AMD at Argonne, the Institute had much to recommend. As delineated in the Institute's Prospectus in 1947, its mission was to "plan and conduct a program of research in pure and applied mathematics aimed

---

Computing in the Aircraft Industry," in The Computing Laboratory in the University, ed. Preston C. Hammer (Madison: University of Wisconsin Press, 1957).

[67] Curtiss. The Institute for Numerical Analysis of the National Bureau of Standards.

[68] Rexmond C. Cochrane, Measures for Progress: A History of the National Bureau of Standards (Washington: U.S. Department of Commerce, 1966), p. 461. Other National Laboratories also set up mathematics sections to assist in their various AEC missions. For example, in 1947 Alvin Weinberg, the director of Oak Ridge National Laboratory, created a Mathematics and Computing Section within the Physics Division under the direction of Alston Householder, who was a trained mathematical biophysicist. In 1948, Householder converted this section into an independent Mathematics Panel in charge of the laboratories equipment. While it is reasonable to assume that the Oak Ridge Math Panel operated in a similar manner as the AMD, this research has not yet been done. See Leland Johnson and Daniel Schaffer, Oak Ridge National Laboratory: The First Fifty Years (Knoxville: The University of Tennessee Press, 1994), 70-1.

primarily at developing methods of analysis which will permit the most efficient and general use of high-speed automatic digital computing machinery." As a more minor activity, the Institute was also available to review, analyze and as necessary help in the formulation of particularly difficult problems in applied mathematics arising in outside laboratories. In addition, the Institute contained a Mathematical Services section which operated the NBS Western Automatic Computer (SWAC) and provided a computing service for local industries, educational institutions, and government agencies.[69]

In practice, though, the role of mathematicians within the Institute's Research section was quite different from what Flanders hoped for at Argonne. Whereas Flanders sought to *integrate* mathematicians directly into the production of science, the Institute's mathematicians were purposefully insulated from this kind of work. First and foremost, the Institute was engaged in fundamental research using the computer to address mathematical problems that, once solved, would find applications in various scientific disciplines. To John Curtiss, Chief of the Institute, mathematicians worked best when removed from the day-to-day, nitty-gritty work of the traditional applied mathematician.[70] Echoing the rarified sentiments expressed by the leaders of the American Mathematical Society during World War II, Curtiss' mathematicians at the Institute might deign to help their engineering or science colleagues on certain problems, but not at the expense of their own work or intellectual autonomy.

Repeating the old saw that nothing new had been discovered in numerical analysis since Gauss, Curtiss suggested that this was because "professional mathematicians were not interested in numerical analysis and the field was left to amateur mathematicians to

---

[69] Curtiss. The Institute for Numerical Analysis of the National Bureau of Standards.
[70] J.H. Curtiss, "The National Applied Mathematics Laboratories of the National Bureau of Standards: A Progress Report Covering the First Five Years of Its Existence," IEEE Annals of the History of Computing 11, no. 2 (1989).

develop."[71] Consequently, Curtiss made it a policy to staff his Research section with "competent professional mathematicians". In a nod to the notion that these mathematicians *might* contribute directly to scientific work, he "found [it] desirable, however, to have at least one theoretical physicist and one expert in classical applied mathematics on the staff, so that advice can be readily obtained as to profitable directions in which to work."

The insulation of mathematicians from applied work (but not applied mathematics) extended to efforts to provide attractive working conditions. Senior staff members of the Research section enjoyed private offices, had experienced typists on hand, desk computing machines readily available, and a "conscious effort was made to insulate the scientists from administrative red tape."[72] In one sense, the INA created for mathematicians the kind of rarefied work environment that had been commonplace for pure mathematicians.

Thus, the Institute's work differed from Flanders' proposal in two key areas: where mathematical problems originated and the working relationship between mathematicians and other scientists. For the Institute, problems still came primarily from the mathematicians; at the AMD problems were to arise primarily from applied fields. While the AMD might borrow some of the organization and goals of the Institute, overall Flanders and subsequent directors were looking for something different.

Although Flanders' exact arguments to management are unknown, the preceding discussion points to certain larger forces in contemporary scientific culture -- political, scientific, and economic -- to which he surely appealed. In any case, in early November,

---

[71] Curtiss. The Institute for Numerical Analysis of the National Bureau of Standards. It is important to remember that the "amateurs" to whom he is referring were mathematical physicists, engineers, and other scientists.
[72] Ibid.

1956 Norman Hilberry, the deputy director of Argonne, formally announced the formation of the Applied Mathematics Division and appointed Donald Flanders its first director. The Applied Mathematics Division did not issue its first annual report for almost two years, during which time Donald Flanders committed suicide.

The people who knew Moll Flanders are generally unwilling to discuss his death, but it is clear that he had suffered from depression for some time and was under tremendous stress. A close personal friend of Alger Hiss, Flanders had been called on to testify before the House Un-American Activities Committee about his contact with the accused traitor in the months preceding his move to Los Alamos, as well as his own family’s leftist leanings. In 1956-57, Flanders was absent from work for extended periods, at which time William Miller acted as director in his stead.

Despite the tragedy, the consolidation of the computing services continued, and the first annual report in 1958 revealed a bare-bones organizational structure, with sections for mathematical analysis, digital programming, digital machine operations, and analog computing. A fifth section listed outside consultants; one of whom was Richard Courant from NYU and another Nicholas Metropolis, who also served on the AEC Computer Advisory Committee. In total, the AMD staff numbered forty-nine.

However, the complete consolidation of the two computing sections into one Division was not complete until sometime in 1959, at which time six sections were now listed: mathematical consultation and research, programming research and development, applied programming (further divided into groups specializing in a particular type of problems such as physics or chemistry), computer engineering, digital machine operations, and analog operations. Impressively, the young division had also grown to 115 members. In addition, the quality of the outside consultants continued to be excellent, and their institutional affiliations reflect a general elevation in the status of applied mathematicians within the mathematical community at large. Joining Courant and Metropolis (from U. Chicago) as AMD consultants where Garrett Birkhoff, of Harvard, K. O. Friedrichs, of the Courant Institute and a former student of Hilbert in Göttingen, and Bernard Friedman, an applied mathematician who did his post-doctoral work under Courant and was a professor at Berkeley.[73]

With Flanders' demise William Miller, a member of the mathematical analysis section, became acting director. Miller brought with him vision, energy, and most importantly, the talents to sell his ideas to administrators at the laboratory and federal level, as well as to other scientists. As Alex Roland has shown in his study of DARPA's Strategic Computing Program of the 1980s, entire technological trajectories can be shaped by the goals and methods of particularly strong individuals. In the same way that Robert Kahn and Robert Cooper gave vision and voice to Strategic Computing, Bill Miller was the personality that indelibly shaped the AMD's first decade.[74]

Miller first came to Argonne in 1953 as a summer intern while pursuing his Ph.D. in physics at Purdue. In his dissertation research, he had made extensive use of numerical techniques and this put him in contact with Al Perlis, a legendary pioneer of computer science, who was also at Purdue. Although Miller's dissertation computations were handled satisfactorily by a Card Programmed Computer (CPC), both his advisor and Perlis thought that he should learn more about the emerging field of digital computers and suggested he work on the AVIDAC project at Argonne over the summer.

---

[73] Applied Mathematics Division Summary Report. Argonne: Argonne National Laboratory, 1960. ANL-6195. MCS archives.
[74] Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993.

During 1953 and 1954, Miller spent summers at Argonne developing mathematical subroutine libraries and input-output routines for the AVIDAC.[75] He quickly distinguished himself, demonstrating, in particular, superior administrative talent. Yet it was another quality that Flanders emphasized when offering him a permanent position on the computing staff. "The excellence of his work," Flanders wrote:

> "and his *striking ability to communicate with scientists from other Divisions on their problem* lead me to offer him a permanent position on the staff, with the particular assignment of serving as 'contact man' for problems coming to us. This function he has fulfilled with exceptional competence. In numerous instances his ability to penetrate to the heart of the physical problems, combined with his understanding of computing machine capabilities and limitations, has resulted in much clearer and more effective mathematical formulation of problems. In this work not only his scientific competence but also his personality has played an important role, since he is a sympathetic listener and demonstrates an evidently sincere desire to co-operate."[76]

Miller considered himself a "liaison" between the AMD and other scientists at Argonne, and his main duties consisted of helping other scientists formulate mathematical models of their problems and then expediting their programming.[77] While this seems fairly straightforward, in truth the success of the AMD depended to a large extent on the ability of mathematicians to understand both the physical problem under investigation and the capabilities of the computing equipment. In this sense, Miller displayed the blended skill-set that computational scientists of the 1980s would claim differentiated them from computer scientists and traditional researchers.[78]

Given his visible role at the Laboratory and his excellent administrative skills, Miller was a natural choice to be interim director after Flanders. After his appointment, though, lab management showed little initiative in finding a permanent replacement for

---

[75] Interview with Dr. William Miller by author, Nov. 11, 2003.
[76] Flanders, Donald, to N. Hilberry. Nov. 7, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.
[77] W.F. Miller, to G.A. Erskine. December 7, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.
[78] Wilson, "Grand Challenges to Computational Science."

Flanders. It was Richard Courant, during one of his several yearly visits to the AMD who suggested to Argonne administrators that Miller be given the Director's job.[79] Almost immediately Miller was offered the job and given twenty-four hours to accept, which he did.

While acting director, Miller had been asked by lab management to prepare a master plan for the activities of the Applied Mathematics Division to be submitted to the Atomic Energy Commission. The strategic plan was intended to address current areas of application for computers as well as suggest promising areas for future research in computer applications. Miller, displaying political, technical, and scientific acumen, drafted a report that became the blueprint for AMD activities for the next decade. His subsequent elevation to the position of Director ensured that he was in a position to oversee the implementation of his comprehensive vision.

Because Miller is so central to the organization of the AMD and the way it interacted with scientists at Argonne, it is worthwhile to spend some time analyzing his vision. The centerpiece of his report is the assertion that applied mathematicians are crucial to the continued quantification of science and technology and that the main role for computers is to accelerate this trend.[80] Released on May 5, 1961 <u>Mathematics and Computer Research at Argonne National Laboratory</u> is a manifesto for the direct integration of mathematical expertise in the conduct of science. To be used most effectively, Miller argued that research in applied mathematics and computer sciences should be carried out in close contact with the quantitative sciences. Unfortunately,

---

[79] Interview with Dr. William Miller by author, Nov. 11, 2003

[80] W.F. Miller. <u>Mathematics and Computer Research at the Argonne National Laboratory</u>. 1961. Box 11, Laboratory Director's Reading File, ANL, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA. The parallels between Miller's language and that of the 1961 CAG report reveals both the tightness of the computer community in the early days, and the extent to which they held a common view about the importance of applied mathematics research in advancing computer applications.

neither industry nor academia appeared interested in this kind of work.[81] Mathematicians within university math departments, he asserted, "are interested principally in the mathematical problems which have implications on the structure and foundations of mathematics itself," while industry, in general "is interested in exploiting the practical applications without encouraging research on the truly mathematical problems which arise in applications."[82] If this situation is allowed to continue, Miller warned that it would be a tragedy both for the quantitative sciences, which rely on applied mathematics, and for the field of mathematics, which he felt would lose one of its prime motivating forces.

Rather than perpetuate this arrangement, Miller proposed an alternative vision whereby applied mathematicians and computers would operate in the "hybrid area between pure mathematics and the quantitative sciences" to the mutual benefits of both fields. Miller's deliberate use of the term "hybrid" was a rhetorical device that points to his multilayered conception of computational science. At the macro level, he defined a new kind of scientific space that blended old and new styles of scientific research and was closely aligned with the emerging characteristics of post-war Big Science. The hybrid area was inherently team-based, unified by mathematics, made possible by high-speed computers, and was costly both in terms of equipment and manpower. In addition, the hybrid area was an intellectual crossroads where experiment met theory and physical or biological phenomena were translated into a mathematical language suitable for machine computation. The potential benefits of creating such a space, Miller argued, were enormous. Already, computers were practically indispensable in such areas as the design of complex engineering systems, large model calculations for atomic, molecular,

[81] IBM is one notable exception.
[82] Miller. Mathematics and Computer Research at the Argonne National Laboratory. Succeeding quotes are from the same source.

and nuclear models, the calculation of macroscopic properties of materials, calculating the response of various physical and chemical systems to the effects of nuclear radiation; and in the analysis of experimental data.

Miller also envisioned a larger role in science for mathematicians and computers. If a collaborative arrangement could be implemented correctly, it might “provide one of the substantial unifying forces for bringing together many of the diverse areas of research in the nuclear sciences.” In a sense, the hybrid area would encourage the blending of different research traditions. Furthermore, because the hybrid area shared a common language of mathematics and the common tool of the computer, new points of contact between different disciplinary agendas might be made manifest.[83] The key to this fertile zone was mathematics. Regardless of the field of inquiry -- biology, physics, or chemistry -- using a computer required the creation of a mathematical model to describe the phenomena in question. Thus, computational tools or methodologies developed for one discipline could be applied effectively in another. For example, calculations simulating neutron diffusion through various materials were as useful to engineers as they were to biologists, chemists, and physicists. Thus, *general-purpose* mathematical and computing tools applicable to a wide range of scientific and engineering disciplines might serve as a *basis for collaboration* for future research and potentially allow insights in one area to be transferred to or adopted by another.

Although he was not the only person to hold such a view, the import of what Miller was calling for is significant. In fact, his emphasis on locating the common methodological threads running through different disciplines later became a major focus

[83] On disciplinary agendas, see Mahoney, "Computer Science."

around which computer scientists tried to define their field.[84] Although this process of finding common methodological threads across different disciplines would be uneven at first, Miller suggested that this process would accelerate as more powerful and more general computer applications were developed. The wide use of Monte Carlo calculations, a statistical sampling technique that found applications in almost every scientific discipline, was just one example that Miller cited as a computer technique that was widely applicable in different fields and thus built bridges between researchers with disparate interests.[85]

Miller also proposed that future progress in scientific computing, and especially the mathematics of machine computation, required a new kind of expert to work in his proposed intellectual and disciplinary space. Heretofore, physicists and engineers had done most of the work required to apply computers to physical problems. These efforts to develop practical mathematical methods were important as all such work could not be relegated to mathematicians. However, Miller argued that their limited mathematical training meant that they were ill-equipped to carry this work beyond the elementary stages. More detrimental to the vision of computational science, they were not in a position to determine the implications of many of the features of the mathematical foundation of a theory or the basis for an analytical or computational procedure.[86] Miller claimed that the lack of clarity in the development of quantum field theory was a result of physicists pushing the field without working with mathematicians. As a result, the

[84] Seymour V. Pollack, "The Development of Computer Science," in Studies in Computer Science, ed. Seymour V. Pollack (The Mathematical Association of America, 1982).
[85] As Peter Galison has noted, regardless of disciplinary training researchers from a variety of fields could openly discuss Monte Carlo experiments even if they had nothing else in common. By the mid-50s the Monte Carlo had emerged as a *lingua franca* for researchers in the nuclear sciences and engineering. For a detailed discussion of the development of the Monte Carlo technique, see Peter Galison, Image and Logic: A Material Culture of Microphysics (Chicago: University of Chicago Press, 1997)., especially Chapter 8.
[86] Miller. Mathematics and Computer Research at the Argonne National Laboratory.

development of the entire field was hindered.[87] For complex mathematical theories to continue to develop, the gap between physical theory and mathematics would have to be bridged by the combined efforts of physicists and mathematicians. It was within the hybrid area where this kind of work could be pursued.

It wasn't so much Miller's contention that mathematicians could contribute to science that was new. More important were his efforts to define the intellectual space in which collaboration could be fostered. At another level, however, the qualities of the hybrid area mirrored the qualities that Miller saw in applied mathematicians. Put another way, if Miller's hybrid area was an intellectual space that blended theoretical and experimental science with computational mathematics, then Miller's actors within the space were themselves hybrids -- mathematician/computer expert/scientist. In short, Miller sought to define a new kind of mathematical researcher, adept at working in the hybrid zone. With expertise in mathematics and a strong understanding of a particular scientific area, Miller believed that applied mathematicians were in a unique position to develop the needed tools and techniques to extend the range of science and engineering problems that could be addressed using high-speed computers.

If this seemed grandiose for the time, perhaps more far-reaching was Miller's attempt to define the nature of mathematics research that would occur in the hybrid zone. In particular, he argued in this report that applied mathematicians had to be allowed to carry out their own research on mathematical problems that they encountered during their service duties to other divisions, but which could not receive the proper attention while trying to solve the problem at hand. This point is crucial for Miller's vision for the AMD. Because applied mathematicians worked under deadlines when they helped other

[87] Ibid.

researchers solve problems, it meant that the solution they produced was not logically complete by the standards of pure mathematicians. Quite often the applied mathematician used a mixed approach where analytical methods, physical intuition, numerical computation, and even empirical reasoning were used to arrive at an approximate solution.[88] The bricolage character of such an approach meant that in many cases, it was not possible to provide a strict proof of the validity of the method used. While this kind of approach is the rule for applied mathematicians, it is anathema to pure mathematicians (and in some respects, herein lies a central issues as to why pure mathematicians distance themselves from applied math). Thus, Miller's argument that applied mathematicians be allowed to do the research necessary to establish the mathematical foundations of their practical mathematical work had two goals. First, within the hybrid zone it would establish applied mathematicians as equal partners in collaborative work, rather than being handmaidens to other scientists. And second, establishing such a research agenda would seem to bridge the gap between pure and applied mathematics. No longer would the applied mathematician be cast as an "algorithmiker," a slight which implied that he or she blindly followed a set rule. Hybridity for Miller's mathematicians meant that their problems would emerge by helping other scientists, but they were also well-suited to investigate the implications of these problems on the foundations of mathematics itself.

Finally, it seems that Miller's efforts to stake out the somewhat amorphous hybrid area between pure mathematics and applications, in which mathematical experts and computers could thrive, reflected a general level of uncertainty about how to categorize this new collection of intellectual and disciplinary activities which would engage his

[88] Richard Courant, "Methods of Applied Mathematics," in Recent Advances in Science: Physics and Applied Mathematics, ed. Morris H. Shamos and George M. Murphy (New York: New York University Press, 1956).

applied mathematicians. Whether conscious or not, Miller's attempt to delineate a new kind of mathematically-oriented disciplinary space which integrated elements of theory and experiment in close ties with the physical, biological, and mathematical sciences, mirrored a similar, and concurrent, epistemological debate as to where computer simulations themselves fit into scientific practice. As Galison has pointed out, Monte Carlos formed a "*tertium quid*" or "third thing" that was difficult to categorize. They were:

> "a simulated reality that borrowed from both experimental and theoretical domains, fused these borrowings together, and used the resulting amalgam to stake out a netherland at once nowhere and everywhere on the usual methodological map."[89]

Because simulations reside in computer memories rather than in any physical space that could be measured, researchers who employed them found it difficult to place their research within the neat categories of either theory or experiment. On the one hand, simulations, like theoretical physics, were independent both of position and scale, could be applied to phenomena in phase-space and were essentially symbolic in that the computer *program* (like an algebraic formula) could handle an infinite number of calculations. On the other hand, the error analysis involved in crafting the computer program more nearly approximated the techniques of experimental science, as simulators constantly sought to refine the resolution of their software program in order to obtain better correlations to theory.[90]

Thus, computer simulations were themselves hybrids, with implications that threatened to destabilize established demarcations between theory and practice in different disciplines. The development of computers had created the potential for a new

---

[89] Galison, Image and Logic, p. 691.
[90] Ibid., pp. 730-31.

kind of disciplinary space whose research characteristics were still indeterminate. Miller wanted to stake out a claim to this space for his applied mathematicians and computer scientists. They would be theorists *and* experimentalists, a kind of scientist that did not fit neatly in traditional disciplinary categories. While some activities were obvious, for the most part the intellectual boundaries of the hybrid area would emerge from the interdisciplinary collaboration fostered by the organization of the Applied Mathematics Division. It would be real-world problems and the need to reformulate them for the computer which would help to define the disciplinary agenda of the applied mathematicians. In short, the field would emerge in real-time.

A final way in which Miller's report can be analyzed is as an exercise in self-fashioning. As stated previously, computers provided an opportunity to establish applied mathematicians as legitimate scientists in their own right. Consequently, Miller's report subtly emphasized that the discipline of applied mathematics incorporated the same trappings as other scientific and engineering disciplines. AMD scientists conducted research toward improving computing machines and deriving new methods for their use[91], they valued precision, measurement, and accuracy in their work[92], their work had both short- and long-term payoffs, they were attuned to the bottom line and thus sought the most efficient procedures given a set of physical and economic constraints, they spoke their own language, and had the inclination to expand their range of activities. On this last point, Miller was most insistent:

> "Although the role of the Applied Mathematics Division is to enhance the work of other scientific divisions, it is implicitly assumed in the following discussion that the Division will play an active role as opposed to a passive one. That is, the motivation for research in AMD should not come entirely from the other divisions but rather AMD should actively engage in bringing to the other

[91] ANL Public Relations, "Press Release: Objectives of Argonne's Applied Mathematics Division," (ANL: Argonne National Laboratory, 1963), p. 3.
[92] Designing metrics for evaluating system designs are a key component to computer science research.

> division suggestions for enhancement of their work through applications of mathematics, computers, and automation."[93]

Self-fashioning was not just rhetorical – it also had very practical motivations. Miller was acutely aware that in order to attract high quality mathematicians to applied work -- especially to a government laboratory like Argonne -- it was necessary to provide opportunities for the mathematician to distinguish themselves within their own profession.[94] While the prospect of a joint appointment in a mathematics department at Chicago, Northwestern or Purdue was almost certainly required to attract a senior mathematician to the AMD, these positions were difficult to find. Throughout the 1960s, despite repeated admonitions from the AMD Review Committee and due diligence by Miller and his successor Wallace Givens, it was rare that they were able to offer a joint appointment to prospective mathematicians.[95]

Compounding the problem of joint appointments was the fact that surrounding universities were pursuing mathematicians within the AMD as vigorously as the AMD was pursuing theirs. Between 1958-1963 five Midwestern universities, Northwestern, IIT, Chicago, Notre Dame, and Purdue either initiated or were looking for people to initiate programs in computer science. Over the same period, Illinois, Wisconsin, Iowa

---

[93] Argonne National Laboratory Preliminary Program Budget Fy 1963 through Fy 1967. Argonne National Laboratory, 1962. Director's Subject Files, B93-00125, Argonne National Laboratory Archives.
[94] Miller. Applied Mathematics Division Long Range Plan.
[95] Report of the Review Committee Applied Mathematics Division. ANL, 1962. ANL Review Committee for AMD, 1958-1980, B93-00147, MCS Archives, Barschall, to "Review Committee for Physics and Applied Mathematics Divisions," The Review Committees at Argonne were first suggested by the University of Chicago's Policy Advisory Board for Argonne in 1958. The Review Committee for the AMD was established on April 21, 1960 and had the purpose of periodically reviewing their work and to make recommendations for strengthening the staff and program. The candidate list for the first review committee is illuminating because it demonstrates how tightly knit this first group of applied mathematicians/computer scientists were. The names include: Richard Courant, Abraham Taub (U. Illinois), Alan Perlis (Carnegie Institute of Tech), George Forsyth (Stanford), N. Metropolis (Chicago), and Bernard Friedman (U. California). Note that a number of these men also served on the AEC Computer Advisory Board. See W. B. Harrell, to Nicolas Metropolis. "Proposal for Review Committee," April 21, 1960. Policy Advisory Board, 1957-1967, 11, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

State, the University of Iowa and Washington University were seeking considerable expansions of their programs. Each of these institutions was looking for four or five applied mathematicians in the areas of numerical analysis, programming, logic, and information theory and had actively sought to hire people from the Research and Consulting section.[96] Even more detrimental, the AMD was unable to offer a salary competitive to what a top flight mathematician could command either in industry or from a university.[97]

Given these financial and (perceived/actual) professional disadvantages, Miller wanted to attract quality mathematicians by offering considerable freedom in research and hoped that this might compensate for the lower salary and lack of university affiliation.[98] No doubt, the opportunity to work closely with scientists and engineers engaged in the nuclear sciences, the occasional Nobel laureate, plus access to some of the most advanced computing environments in existence was appealing to prospective mathematicians. The success of Miller's approach is evident in the 1963 recruitment to the Research and Consulting section of J. Wallace Givens of Northwestern, one of the world's leading mathematicians in the area of matrix algebra computations.[99] In addition, from 1958-1963 the AMD not only withstood the assault on its mathematicians, but managed to acquire several highly qualified people to the division.[100]

---

[96] W.F. Miller, to J. R. Gilbreath. "Remarks on the Report of the Review Committee for the Applied Mathematics Division, Argonne National Laboratory.," Jan. 10, 1963. Policy Advisory Board, 1957-1967, 11, Records of the Argonne University Association Policy Advisory Board, University of Illinois, Urbana-Champaign, Special Collections.
[97] Barschall, to R.W. Harrison, "Review Committee for Physics and Applied Mathematics Divisions,"
[98] Miller says this explicitly – "if a good example is set to encourage the development of these scientists (mathematicians), one can anticipate a stronger interest in applied mathematics in the future." Miller. Mathematics and Computer Research at the Argonne National Laboratory.
[99] Previously Wallace Givens had served on the AUA Review committee for the AMD.
[100] Miller, to J.R. Gilbreath, "Remarks on the Report of the Review Committee AMD," Jan. 10, 1963, Policy Advisory Board 1957-1967, box 11, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

A final reason why Miller pushed for applied mathematicians to be able to pursue fundamental research emerged in an interview I had with him. He felt it was important for AMD mathematicians to push the frontiers of their field in order to prepare them to be receivers of the latest advances in computer techniques. If the division focused solely on their service component rather than doing fundamental research in systems design, numerical methods, and mathematical modeling, he believed that it would not be in a strong position to evaluate and assimilate the new computational techniques that were being developed by industry and university researchers.[101]

Miller believed that the Applied Mathematics Division, tightly integrated with Argonne's science and engineering divisions, could produce the kind of mathematical tools and techniques necessary to expand the use of computers into nearly every area of research. In particular, he identified four areas of mathematical investigation that seemed most promising for the AMD: the mathematical formulation and foundations of modern physical, chemical, and biological theories; the development of analytical and computational procedures for model calculations and the problems of modern science; the development of non-numerical computer methods for such problems as pattern recognition and artificial intelligence; and the development of formal [analytical and numerical] procedures for the analysis of complex engineering systems. Beyond these fields, research in mathematics would be conducted in conjunction with research and development of computers and computer devices. Since computers were the tool for expediting the development and application of quantitative theories, it was important that they be developed in close contact with applied mathematicians so that it would be

[101] Interview with Miller. Miller's vision for how to integrate mathematicians and scientists together, while not unique to him, was certainly in the vanguard of such efforts. As he pointed out in his report, aside from the applied mathematicians working with Courant at NYU, Miller says "there does not exist a single center wherein several mathematicians with mutual interests in applied mathematical problems of modern science work together." Miller. Mathematics and Computer Research at the Argonne National Laboratory.

possible to study the implications of mathematical techniques on system organization and design logic of the computer.

Miller had spelled out an ambitious, wide-ranging plan in the 1961 report. In the early days of mainframe computing, his ideas were fresh and seemed as likely to succeed as any other plan. True, Miller was calling for the creation of a new kind of scientist (to populate an already crowded and competitive scientific landscape) and the immediate elevation of that researcher to a position of some prominence within the scientific community. Yet, if his plan proved feasible, the potential gains for science and engineering would be tremendous. If all went well, mathematical expertise would suddenly be highly desired by researchers in all fields. Miller's "hybrids," working in the area between pure mathematics and the applied sciences would energize interdisciplinary collaboration, further the mathematization of the sciences, and help make computational science a reality.

In practice, the situating of applied mathematicians was not so straightforward; to a large extent the entire enterprise hinged on the collaborative nature of computing itself. By the mid to late 1960s, Miller's grand vision began to run afoul of several interacting forces. In terms of the technology, computers became smaller and more affordable, making it possible for the various divisions at Argonne purchase their own machines to handle certain computing needs in-house. At the same time, the proliferation of these "minicomputers" increased the amount of raw data that was collected from experiments, which in turn quickly saturated the powerful central computers of the AMD which were used to process much of this data. As turn-around time for work submitted to the AMD suffered, scientists and engineers began to feel that the research activities of the division were siphoning away resources that could be devoted to providing faster computational services.

Further undermining Miller's vision was the concurrent emergence of computer science as a recognized scientific discipline which provided applied mathematicians and computer scientists with a professional umbrella under which to work, and not, surprisingly, less inducement to subsume their research interests in the name of service to others. But even before professional considerations entered the picture, applied mathematicians were not being called upon by other scientists to help in problem formulation. The culprit, again, was the advancements in computer technology. As more software tools were developed to handle mathematical problems, previously difficult computational procedures became routine.

And herein lay one of the peculiarities of computer science and applied mathematics -- namely its inherent invisibility. A major objective in the implementation of any piece of software is to obscure the inner workings of the algorithms and code so that the user only has to focus on the output of the system. For example, in the course of writing a FORTRAN program, a scientist might include a small subroutine that performs a particular mathematical function. The scientist does not question how the subroutine works or whether it will return the correct answer -- that had already been worked out by mathematicians and programmers in the process of creating the software routine. However, once instantiated in software the creativity and intellectual contribution of the mathematicians and programmers becomes invisible to the user. This peculiarity tended to work against computer specialists. In terms of garnering support from computer users for the research activities of the AMD, the invisibility of the work involved in creating computational tools obscured the extent to which fundamental mathematical research was required to produce them. As scientists and engineers relied more on prepared subroutines and applied programmers to handle their computational work, it seemed less and less imperative to support the research activities of mathematicians and computer scientists.

As the AMD was buffeted by these different forces, a split began to develop between the producers and consumers of computational tools. The result was that computational science continued to develop among scientists, but without the close collaboration of applied mathematicians and computer scientists. The evolution of computational science discussed above can be seen in the activities and organization of the AMD from 1957-1974.

The Drive for Disciplinary Independence

While the stated objective of the AMD -- to provide mathematical assistance to other scientists at the lab -- seemed to suggest a arrangement, in truth during the first few years of the Division's existence it was nearly impossible for scientists or engineers from other divisions to gain access to the AVIDAC, or later the IBM 704, without first having to work with a mathematician from the Consulting and Research section. No doubt a strong consideration for this move was technical; the tiny memories of both the AVIDAC and IBM 704 left no room for program bloat and thus necessitated that problems be simplified mathematically as much as possible before they reached the programming stage. Beyond this, it was often necessary to code new mathematical subroutines to solve specific portions of complex problems, and the consulting mathematician would be able to identify what work needed to be done. Finally, there was the simple fact that scientists and engineers were generally unfamiliar with the inner workings of computers -- how to express problems using Boolean arithmetic, how to divide a problem so that it could be computed efficiently, or how to determine what kind of output a given problem might generate.

By at least one metric, this early arrangement was quite successful in terms of integrating mathematicians into interdisciplinary work, as every project listed in the first Applied Mathematics Division Summary report, covering the period from November 1956 to June 1958, included a mathematical analyst from the Research and Consulting section. Problems originated in every division at Argonne, including numerous investigations from the Biological and Medical Research Division and Radiological physics.[102]

However, over the next three years, as more scientists demanded time on the Laboratory's computers, and as the number of computational tasks that were routine (or reached "production" status) increased, it became increasingly common for scientists and engineers to work directly with programmers in the division rather than members of the Research and Consulting staff. This led to some suggestions that programmers might be moved to the divisions where the work originated rather than remaining within the AMD.

Another impediment to collaboration was that scientists and engineers were learning how to program in FORTRAN, which had quickly become a *lingua franca* for computing after its release for the IBM 704 in 1957.[103] The desire of scientists to do their own programming created a need for Miller, now director of the AMD, to rethink both the service and research components of the division. On the one hand, he wanted to encourage the use of computers by scientists which suggested that they be allowed to do some of their own programming. On the other hand, the need for efficient programming,

---

[102] Applied Mathematics Division Summary Report, Nov., 1956- June, 1958.
[103] John Backus, "The History of Fortran I, II, and III," IEEE Annals of the History of Computing 20, no. 4 (1998).

quality mathematical analysis, scheduling of computer runs, and the professional development of his staff suggested a more codified approach to collaboration between mathematicians and scientists.

By 1961 Miller was advocating that computing operations at Argonne be centralized, but not go so far as to advocate a “closed shop.” In a “closed shop” operation, the requestor for computer services never gets a chance to interact directly with the machine and must work entirely through programmers. On the other hand, giving scientists free reign with computers in a completely “open shop” would lead to chaos in programming and scheduling, while at the same time eliminating the interdisciplinary collaboration that he hoped would emerge from the computing operation.

Unwilling to pursue either course, Miller instead chose a middle ground whereby the AMD increased its educational services that taught scientists how to use computers and do some programming. To assist scientists in learning how to program, the AMD also built a FORTRAN preprocessor that ran on the lab’s IBM 1401. The preprocessor, amusingly dubbed “DDT”, checked a scientist’s FORTRAN statements to detect and list errors (or bugs!). In this way, scientists could do their own programming and have it de-bugged prior to submitting it to the 704’s processor queue. Second, in order to forestall efforts to take programmers out of the AMD, Miller reorganized the division. Programmers were now assigned on a long-term basis to work on problems arising in specific divisions. This arrangement, he felt would have certain advantages for everyone. By working continuously with scientists from one division, the programmer would become well-acquainted with the problems of that division and communication between problem proposer and programmer would be on a good basis.[104]

Behind these initiatives to keep programmers centralized within the AMD was Miller's conviction that such an arrangement held the most promise for diffusing computational tools and techniques to the entire laboratory. First, by being members of the AMD, these programmers would be under a management which was prepared to look after their professional well-being by keeping them abreast of new developments in their field. Second, as members of the AMD these programmers would be in a better position to obtain mathematical assistance from the Research and Consulting section if they encountered a problem that was beyond their scope. Finally, Miller argued that new mathematical, numerical, and programming techniques would filter down to the programmers in a centrally managed group better than to programmers assigned to groups whose primary interests were not mathematical.[105]

That Miller had to defend the organization and activities of the AMD as early as 1961 suggests the difficulties inherent in trying to establish the autonomy of a new scientific division within the established Argonne community. In general, scientists and engineers at Argonne were interested in the computer services of the AMD and were much less interested in promoting the professionalization of programmers or supporting the open-ended research activities of mathematicians. Just a few years earlier the Lab's two computing groups had generally operated under the radar and existed primarily to support Reactor Engineering and the Physic division. Despite the granting of Divisional status, it was difficult for the AMD to escape previously formed perceptions that they were there to be specialized technical assistants.

---

[104] Here he notes that the communication problem is often quite serious.
[105] Miller. Applied Mathematics Division Long Range Plan.

Further compounding this problem was that the AMD retained its funding structure from the old computing group days. In particular, the activities of the AMD were supported by charging other divisions for computing and programming time, with an overhead charge tacked on to support the mathematical research efforts. Every year, members of the AMD, usually Margaret Butler, would go from division to division selling the services of the AMD.[106] This approach was especially difficult in the early days because it required the division desiring computational service to estimate their usage for the entire year, even if they didn't exactly know how they might use the computers. It was Butler's job to suggest types of computational problems compatible with each Division's activities that members of the AMD could address. In a very real sense, then, from the beginning computational science was a solution in search of a problem. If, as often happened in the early days, computer usage exceeded what was forecast by a particular division, grumbling ensued regarding the funds going to mathematical research at the expense of more computer time.[107]

Since there was no alternative to charging users for computer and programming time, Miller sought to quell scientists' complaints by securing a separate research budget for mathematics from the AEC. If successful, he would remove a point of contention with other divisions about funding mathematicians, provide the support needed for his mathematicians to pursue their own fundamental research, and further solidify the autonomy of the AMD within Argonne.

The need for direct financing of research in mathematics had long been recognized and was first recommended in the Long Range Plan for Argonne in 1959. It was then specifically requested in Argonne's budget to the AEC for fiscal year 1962, but

[106] Interview with William Miller by author.
[107] Interviews with Margaret Butler, Joe Cook, William Miller by author.

again money was not forthcoming. At the federal level, in 1960 the Computer Advisory Committee also strongly recommended that the AEC provide direct funding for mathematics and computer science research in areas pertinent to its mission. However, over the next three years the situation was largely ignored. Not surprisingly, the failure of the AEC to support mathematics research meant that the programs that did exist within the Lab system were fairly unproductive. Successive reports from the CAG commented on this point, noting that "the growth of mathematics and programming activities at the National Laboratories . . . in general has been disturbingly slow" and "pure mathematics is about completely nonexistent in the programs of the [AEC's] Division of Research.[108] Even the five million dollars slated for math and computer science research in the AEC's 1964 budget was considered "woefully inadequate" by the committee.[109]

It seems that the main difficulty in securing a separate research budget for mathematics research lay in the structure of the budgetary system of the Research Division of the Atomic Energy Commission which funded computer research. Apparently, any request for direct funding of mathematical research would appear as a new program for the Research Division.[110] Within the decentralized structure of the AEC overseeing several multiprogram laboratories, the creation of any new program was politically difficult, even if there was no net change in the total costs to the AEC.[111]

With research money unavailable in 1961-62, the AMD continued to fund its operations through service fees from other divisions. Members of the Research and Consulting section, while conducting their own research on the experimental computer

---

[108] Pasta. Minutes of CAG, AEC Sept. 28, 1961.
[109] Meeting of the Mathematics and Computer Sciences Research Advisory Committee with the GAC.
[110] N. Hilberry, to George W. Beadle. "Report on Review Committee for the Applied Mathematics Division," July 17, 1961. Box 11, RG 326: Records of the Atomic Energy Commission, Policy Advisory Board 1957-1967, GLFRCNARA.
[111] For a full discussion of the funding morass of the AEC and its contribution to the "Laboratory Problem" see Westwick, The National Labs: Science in an American System, 1947-1974, pp. 88-110.

GEORGE, still devoted a tremendous amount of time to interdisciplinary work. From July 1961 to June 1962, the eleven full-time members of the R&C group participated in twenty-three on-going collaborative projects with scientists and engineers from every division at Argonne. Projects ranged from the development of mathematics to model neutron diffusion, to pattern recognition programs, to the development of the mathematics necessary to enable real-time analysis of experimental data.[112]

As demands for computer services at Argonne continued to expand dramatically, AMD directors launched a concurrent effort to define the activities of the division more specifically. I argue that these initiatives in the early 1960s reflect broader attempts to define computer science as a distinct discipline. Part of the growing pains of any new scientific discipline is deciding what it includes and what it excludes. Subtle changes to the wording appear in the preface to the AMD's 1961-62 Annual Report which point to the evolving nature of computing and what I contend was a slow bifurcation of computing into research and service components. In contrast to the 1959 Report, which listed the Division's objectives as "providing mathematical assistance to other scientists in the laboratory" by "conducting research in numerical analysis and other branches of mathematics", the 1962 Report expands these activities to "conducting research in applied mathematics, theory and practice of computation, and design of computers and information processing machines."[113]

If the change in language used to describe the activities of the AMD and especially the Consulting and Research section demonstrates a more concerted effort to stake out a disciplinary space for applied mathematical research, it also reflects the

---

[112] Applied Mathematics Division Summary Report. Argonne: Argonne National Laboratory, 1962. ANL-6641. MCS archives.

[113] Applied Mathematics Division Summary Report, July 1, 1958 through June 30, 1959. , Applied Mathematics Division Summary Report, July 1, 1960 through June 30, 1961. AEC Research and Development Report, 1961. ANL-6453 Mathematics and Computers, (TID-4500), 16th Ed. MCS archives.

nascent beginnings of the activities that would eventually coalesce into computer science. The Division's research efforts into the theory and practice of computation were a significant departure from their initial agenda, which focused primarily on the speed and efficiency of computing. As computers were applied to an ever-widening number of applications such as non-numerical computation (pattern recognition), the creation of automated programming tools, and the collection and analysis of experimental data, there was increasing interest in studying abstract models of computation as a dynamic process.[114] Miller was aware that this kind of research generally promised long-term results but his willingness to put this new activity front and center in the objectives of the AMD was a forthright articulation of his position that mathematicians and computer specialists were not technical assistants at the beck and call of other divisions.

The revised Preface also reiterates Miller's belief that applied mathematicians could and should have a more central role in the framing of scientific and engineering problems at the laboratory, but makes the process more explicit. Instead of simply offering generic mathematical assistance to Argonne researchers as described in the 1959 Annual Report, in 1962 members of the C&R section were available to "assist laboratory personnel in mathematical consultation, problem formulation, selection of appropriate mathematical and numerical techniques, and… carry out analyses of problems."[115]

Efforts to redefine the role of applied mathematicians within the division were part of a broader restructuring of the entire operation that occurred from 1963-1968. If the new programmatic activities evident in the Preface can be considered a conceptual shift towards greater independence for mathematical research within the Argonne community, then the construction in 1964 of a separate building to house the AMD was a

---

[114] Mahoney, "Computer Science: The Search for Mathematical Theory," p. 617.
[115] Applied Mathematics Division Summary Report, July 1, 1958 through June 30, 1959. , Applied Mathematics Division Summary Report, July 1, 1960 through June 30, 1961.

more concrete testament to the division's autonomy. The rapid expansion of computing services at the lab had been accompanied by a tremendous increase in the number of people employed by the AMD. By 1964, the division had grown to 165 full-time staff members with another 36 temporary staff serving primarily as summer researchers. In addition, the impending arrival of a new state-of-the-art computer, the CDC 3600, to augment the IBM 704 (which was to be phased out), as well as the division's experimental machines (GEORGE, and later FLIP) necessitated a separate facility to house the computing activity.

The promise of a new building and advanced computer facility, together with the outstanding research already produced by members of the C&R section, had enticed several high-caliber people to join the AMD. Foremost of these was J. Wallace Givens, one of the leading mathematicians in the world in the area of matrix algebra computations. Givens had previously served on the AEC Review Committee that yearly evaluated the AMD and was familiar with Miller's goals and vision for the division.[116] In 1963, he took a leave of absence from his position as a professor of mathematics at Northwestern and joined the AMD as Associate Director and Senior Mathematician.

Givens' arrival at the AMD coincided with the beginning of dramatic changes that would challenge the notion of computational science as a collaborative and interdisciplinary endeavor. As the field of computing matured during the 1960s it began to separate along lines drawn between what I have identified as the producers of computational tools and techniques and the consumers of those tools. On the production side, increasingly, mathematicians and computer specialists within the AMD began to identify themselves with the emerging discipline of computer science. While its

[116] Miller, to J.R. Gilbreath, "Remarks on the Report of the Review Committee AMD."

definition and field of study were open to debate, nonetheless by 1968 computer science was widely recognized as a distinct discipline with it own research agendas, funding sources, and professional identity.

As important for what it was, computer science was also important for what it was *not* -- namely it was not about serving other disciplines. While useful applications might emerge from research in computer science and applied mathematics, it was not the task of either computer scientists or mathematicians to find these connections. I suggest that the abrupt reorganization of the AMD during the mid-1960s reflected attempts within the larger computer science community to separate the research and service components of computing.

At the AMD, these organizational maneuvers had surprising and possibly counterintuitive effects. Since World War II, computers had been seen by computer specialists as an *integrative* technology -- one that could help scientists to transcend disciplinary boundaries and facilitate multifarious approaches to doing science. However, as a distinct professional identity began to emerge around computing, and computing technology continued to change, *new barriers to collaboration were created*.

As will be seen, what emerged by the beginning of the 1970s were two threads of computing that remained somewhat independent of each other. On the one hand were the computer scientists and mathematicians who developed tools and techniques to maximize the effectiveness of computers, and on the other were computational scientists who used these tools and techniques to investigate physical and biological phenomena. Yet this cross-fertilization was not the same kind of interdisciplinary collaboration that Miller had envisioned, nor were mathematicians considered central to computational science. This shift in the epistemological location of computational science -- from producers to consumers and from the AMD to other scientific divisions -- ultimately slowed down the development of the field. In order to understand how these barriers were erected, it is necessary to look at how computing changed for both consumers and producers.

Briefly, on the consumption side, as computers became smaller and less expensive, other divisions at Argonne began to acquire their own machines. In addition, by late in the decade, scientists were routinely doing their own programming or had hired programmers to work directly within their division. Having access to robust mathematical software tools and high-level programming languages, there was no need to rely on the Consulting and Research section for anything but the most complex and novel scientific problems. In this light, the centralized computing facilities of the AMD seemed anachronistic unless one had a computational procedure that required a huge memory and a fast processor. As the need for mathematical expertise became less acute consumers of computer services became even more critical of AMD activities and sought measures to reorient the division towards providing more services at the expense of mathematical and computer science research. We will see the effect of this pressure on the AMD later in this chapter.

On the production side, a critical turning point in the development of the hybrid area as the centerpiece for computational science came in 1964 when, thanks in large part to the efforts of Miller, the Comptroller of the AEC agreed to provide direct support for the mathematical research activities of the AMD.[117] The infusion of funds meant that the applied mathematicians of the Consulting and Research section were financially independent. It was no longer necessary for them to spend time working with scientists or engineers from other divisions in order to justify their presence at the lab. Free to

[117] Dr. A. V. Crewe, to George W. Beadle. Jan. 15, 1963. Box 11, RG 326: Policy Advisory Board 1957-67, GLFRCNARA.

pursue their own research, the majority of applied mathematicians did just that.[118] The effect of independent funding can be seen in several ways in the Annual Reports. Whereas the 1963-64 Report lists nineteen different projects in which mathematicians from the Consulting and Research section collaborated with other divisions, beginning in 1964-65, there are zero projects listed. No doubt some mathematicians were still integrated into Laboratory programs, but their research activities were no longer listed alongside applied work. The mathematical projects they worked on certainly had application areas. The difference is that the problems on which these mathematicians worked did not emerge from doing work for other divisions. Instead, it was the mathematicians who set their own research agenda -- provided it fit within the AEC mission. In this aspect, their work more closely resembled the freedom and autonomy prized by pure mathematicians.

A more telling shift in the role of mathematicians within the AMD appears in the evolution of the division's organization charts from 1965 through 1967. In general, organization charts tend to be reactive, not proactive. An organization is created and then evolves to the point where there is confusion as to how different units work together. At times it is no longer clear to people either within or outside the organization who is responsible for what. Not surprisingly, the AMD organization shows similar characteristics.

For the first five years of Annual reports, covering the period 1958-1963, the different sections of the AMD and their members were simply listed in the opening pages of the report. Moreover, there were no attempts to show the relationships between these sections, although it seems that some hierarchy in the Division's activities was suggested

[118] There were some exceptions to this, namely Joe Cook, Gary Leaf, and James Butler.

in the order of the listing: Mathematical Consultation and Research, Programming Research and Development, Applied Programming, Computer Engineering, and Digital Machine Operations. Yet the lack of a formal organization chart is also suggestive of the interconnected nature of early computing. Both Flanders and Miller emphasized to computer users at Argonne that scientific problems were approached and solved through the combined efforts of mathematicians, programmers, engineers, and machine operators. Given the tremendous amount of crossover in the work of each section, lines corresponding to functional roles were not easy to define. However, in 1964 the AMD pursued a new course, including in the Annual Report a traditional organizational chart with lines showing linkages between sections. Examining this, and subsequent "org" charts provides insights into two different aspects of how computing changes on the producer's side. First, the org charts highlight the changing relationship of the AMD's activities vís-a-vís other scientists at Argonne. And second, I suggest that the org charts reveal efforts to institutionalize computer science as a distinct discipline.

On the one hand, the appearance of a traditional organization chart can be seen as a response to demands for more computing services by scientists at Argonne. For many of the computer users at Argonne, the inner dynamics of the AMD were essentially opaque. Scientists and engineers were overwhelmingly concerned with improved service from the AMD and were uninterested in anything that detracted from greater computational throughput. In this sense then, the org chart was an attempt to articulate visually the different activities that comprised scientific computing. By laying out the different sections of the divisions, it was possible to show that a large percentage of the AMD's activities were *not* directed solely to the service function. Thus, in the 1963-1964 chart (see chart, page 167), along the left half of the page stands the Consultation and

Research Section with lines connecting it to Programming Research and Development (further subdivided into Numerical Methods, Systems Programming, and Languages and Logic), and Computer Engineering. On the right half of the page are listed the Applied Programming section (subdivided into sections corresponding to the different user divisions at Argonne), Digital Operations (the group that runs the computers) and the Analog computing group.[119] Tellingly, there are no functional lines connecting the sections on the left and right other than they both report to the Director's office.

The 1964-1965 org chart remains the same (see chart page 168)[120], but suddenly in 1965-1966 there is a profound reorganization of the different sections that points to much larger trends in the field of computing (see chart page 169). In this new schematic, the Consultation and Research Section is now the identifying Section and is listed on the right half of the page. Contained under the rubric of C&R are now sections for mathematical algorithms, mathematical analysis, computer engineering and programming development (further subdivided into logical methods, numerical methods, and partial differential equations), and engineering remote input stations (RADS).[121] On the left half of the page is the central heading "Digital Computing Center." Flowing out of this main heading are lines connected to "Digital Machines Operations", "Systems Programming", and "Applied Programming."

The stark division of the AMD between its research and service components evident in the 1965-1966 organization chart corresponds quite closely to much wider

---

[119] Applied Mathematics Division Summary Report. Argonne: Argonne National Laboratory, 1964. ANL-6952. MCS archives. The divisional breakdown is as follows: Reactors, Physics, Chemistry, Biological and Metallurgical, High Energy Physics, and Management.
[120] The only exception to this is the replacement of "Management Applications" with "Information Processing Applications."
[121] Applied Mathematics Division Summary Report, July 1, 1965 through June 30, 1966. AEC Research and Development Report, 1966. ANL-7280. MCS Archives.

efforts among computer specialists to define computer science as a distinct discipline.[122] A discussion of the early days of computer science will help to illuminate how it connects to the activities of the AMD.

During the previous fifteen years or so, as experience with the use of computers grew and as procedures and services became more complicated, computer specialists realized that there was a sizeable body of knowledge closely related to, but distinct from computer applications. Problems associated with language and complier design, systems configurations, optimization, and numerical methods were far from trivial and grew increasingly important as computers became larger and more complex. Initially, there was little consensus among these practitioners as to whether this body of knowledge constituted its own discipline or whether its proper home was subsumed under, among others, mathematics, electrical engineering, linguistics, or business. Unable to ascribe a specific identity to this field, advocacy for the establishment of a distinct discipline remained sparse in the late 1950s and early 1960s.

As universities began to offer classes in computer-related studies, not surprisingly there was little agreement in terms of what courses to offer and in what department to locate these courses.[123] One approach was to establish interdisciplinary programs in computing. Thus, in 1961 the Carnegie Institute of Technology created a doctoral program in computer systems and communications, while Stanford University placed the computer science division within mathematics and the University of Wisconsin created an independent applied mathematics program. This lack of agreement as to where

---

[122] The following discussion of the development of computer science relies primarily on Pollack, "The Development of Computer Science."

[123] The NSF, which funded many of these initial university computer facilities, established a program for Computer and Computing Science within Mathematical Sciences in 1962, mainly because computing was seen first and foremost as a mathematical tool. See William Aspray, "Arming American Scientists: NSF and the Provision of Scientific Computing Facilities for Universities, 1950-1973," IEEE Annals of the History of Computing vol. 16, no. no. 4 (1994): p. 62.

computing activities belonged was one impediment to the crystallization of an academic framework for computer science.

At the same time, the absence of an identifiable "science" to accompany this rapidly growing field further impeded its recognition as a unique field. Although it was possible to identify collections of useful techniques or approaches to designing or improving computer systems, there was no unifying framework around which this large body of empirical knowledge could be organized. Furthermore, computer science lacked a natural focal point such as the atom or a crystal lattice to serve as a source of observation. Instead, the object of inquiry was a machine slightly over a decade old whose internal functioning could be altered at will. Lacking a tradition of its own, computer science proponents at universities sought academic respect by aligning their programs with the strengths of their particular institutions. In a university strong in engineering the computer science curriculum would usually emphasize numerical analysis; if an institution focused on computer design and construction, computer science courses featured Boolean algebra, switching theory, and mathematical logic instead of numerical analysis. This approach made it possible to provide the rather nebulous field of computing with some semblance of "tradition".

This is not to suggest that computing was developing without any "science" whatsoever. By the early to mid 1960s, the complexity of computers had advanced enough to require operating systems and multi-programming techniques that would allow several unrelated programs to run concurrently. Proper management of scarce computational resources called for highly sophisticated systems organizations. In response to these pressures, theories began to emerge for ways to characterize system dynamics, predict their behavior when parameters were changed, and analyze overall system performance. Beyond the study of operating systems, which was unique to computer science, was the concurrent realization that some degree of systemization could be applied to the development and testing of programs. At Eindhoven Technological University, E. W. Dijkstra produced groundbreaking computer science work by identifying a set of coherent principles corresponding to quality program structures. His work inspired the development of formal testing procedures to verify a program's correctness and spurred additional research into new computer languages designed to implement his principles of structured programming. Other areas pertinent to computer science began to coalesce in similar fashion; operational problems, new computing technologies, and increasingly sophisticated applications raised new issues for computer practitioners and pushed them to develop more comprehensive models of computing to replace the *ad hoc* methodologies that had previously sufficed.

Other groups outside of academia were also interested in promoting the emergence of computer science as a coherent discipline. One such organization was the Association for Computing Machinery (ACM) created in 1947. The ACM took a lead role in trying to establish computer science as a distinct discipline and in 1963 it tasked its Curriculum Committee to determine what areas were included in computer science and which should be taught in universities. Its initial draft, released in 1965, reaffirmed the idea that computer science was a distinct area of study, but the committee was still quite tentative in suggesting what courses and activities comprised a solid computer science education. Over the next three years, however, intense activity throughout the country in curriculum development led to the ACM's second and much more significant report in 1968.

Curriculum ’68, as it is known, serves as a milestone in computer science education and its conclusions provide some insight into the dramatic reorganization of the Applied Mathematics Division in 1966. While Curriculum ’68 ended the philosophical debate regarding computer science as a distinct discipline, its significance lies not only in what it includes, but also in what it excludes from this new discipline. First and foremost, Curriculum ‘68 clearly places the “occupational” components of computing, such as computer operations, coding, and data preparation outside the realm of computer science. In a fundamental reorientation of its 1965 position, Curriculum ’68 further maintains that the major foci of computer science -- the representation, structure, and transformation of information -- is independent of specific computers or applications. In other words, computer science is defined primarily as an abstract (and mathematical) science.

Yet the non-applied nature of computer science does not mean that it is separated from the computer as a physical device. On the contrary, where previously the Committee had recommended that hardware and software be considered separate areas, its new set of guidelines opposes this distinction, contending that they be considered in a single framework, i.e. systems capable of transforming information.[124] In this respect, Curriculum ’68 reflected what had already been occurring in practice, namely the unification of hardware and software design necessitated by ever more complex computer equipment and software systems. As practitioners came to realize, rather than superimposing one on top of another, it was necessary to integrate the design of hardware and software in order to make the most efficient use of computers.

[124] Pollack, "The Development of Computer Science," p. 38.

Finally, Curriculum ’68 identified a third major component of computer science as the search for common methodological threads running through computer applications. Computer science thus would focus on the development of methodologies to suit the common processing needs of users irrespective of the relationship between the users’ disciplines. For example, the area of image recognition had applications in high-energy physics for analyzing spark chamber photographs, in biology for matching chromosome pairs, and in fingerprint recognition software. In essence, the computer science curriculum as defined by the committee suggested that the discipline could indeed build tools that would enable interdisciplinary collaboration. A scientist interested in image processing in biology would be able to draw on computational techniques developed for a physicist.

It is important to remember that Curriculum ’68 was informed, above all, by the accumulated experience of practitioners. As such, it reflected trends already manifest in academic computer science programs as well as in computing centers across the nation. To members of the Applied Mathematics Division at Argonne, the findings of Curriculum ’68 provided some intellectual justifications for the profound organizational changes that had occurred in their division in 1966 -67.

Since its creation in 1956, a succession of Directors at the AMD had done their best to establish that members of the division (and especially mathematicians) conducted their own independent research. As has been discussed, though, computer users had little interest in supporting long-term mathematical research that bore little relationship (as they saw it) to getting immediate problems solved quickly. Beholden to user divisions for research money, many members of the Research and Consulting section struggled to establish some sort of professional identity that would transcend their service duties.

These were people with PhDs in mathematics or a physical science, who were being treated as support staff by other scientists.

The establishment of an independent research budget for AMD mathematicians in 1964 can be read as an official validation of the professional status of these scientists and as the green light for them to pursue their own interests in mathematics and computer science. The subsequent reorganization of the AMD evident in the Summary Report of 1966 was a visual articulation of this change in professional status. Moreover, that Mathematical Algorithms, Mathematical Analysis, Programming Development, and Computer Engineering were now all subsumed under the heading “Research and Consulting” was recognition that the computer sciences had tremendous breadth. Most prominently, the Consulting and Research Section (which could easily have been called the Computer Sciences section) was visually and conceptually separated from the service side of the AMD (operations and applied programming).

In almost every way, the reorganization of the AMD along these lines reflects the coalescence of a distinct disciplinary identity among practitioners of the computer sciences, an attempt to define the areas encompassed by this discipline, and its relationship to other researchers at the laboratory. At the AMD, this disciplinary agenda emerged organically, buffeted by budgetary concerns, pressure from other disciplines, the technology of computing, and its own internal dynamic. But clearly the AMD was also responding to changes in the professional status of computer science itself.

While the newly reorganized Research and Consulting section encompassed many of the key elements later suggested by Curriculum ’68 to be the proper purview of computer science, the changes made to the Applied Programming section in 1967 show similar prescience. (See chart page 170) Whereas since 1961 Applied Programming was organized along divisional lines (i.e. groups of programmers devoted to physics or chemistry applications), beginning in 1967 these programmers were now arranged according to the common methodological *needs* of the disciplines. As if anticipating the findings of the ACM curriculum committee, Applied Programming was now divided into Experimental Science, Engineering and Applied Systems, Theoretical Physics and Chemistry, and Reactor Theory and Development. This regrouping was significant because it emphasized a new way to conceptualize science. Rather than view each science as having unique needs, the reorganized Applied programming section sought to develop computational tools that would address similar problems across disciplines.

Finally, corresponding to the dramatic restructuring of the AMD is the evolution of the Prefaces to the yearly Summary Reports. The changing language evident here is perhaps the clearest statement of the effort to institutionalize computer science as a distinct discipline. Up until 1967, every previous version of the Preface had stressed that members of the Research and Consulting section were "available to assist Laboratory personnel by mathematical consultation, in problem formulation, and in the selection of appropriate mathematical and numerical techniques, and to carry out analyses of problems." In 1966 this description is expanded to include the statement "member of the Section also carry out their own independent research in various aspects of mathematics and programming." In terms of textual hierarchy the activities of the applied programming section are listed next, followed by those of the system Programming section, the mathematical methods group, and finally the Digital Operations section which actually ran the computers.

In 1967 the Preface is entirely revised and foregrounds the mathematics and computer science research activities of the division. The Preface begins with the statement "The Applied Mathematics Division has two objectives" and continues:

> 1) to conduct research in applied mathematics, numerical analysis, theory and practice of computation, and design of computer and information processing equipment;
> 2) to provide mathematical support for the research and development programs of the Laboratory.
>
> Members of the Consultation and Research Section carry out their own independent research in various aspects of mathematics and programming. They are also available to assist laboratory personnel by mathematical consultation, in problem formulation, and in selection of appropriate mathematical and numerical techniques, and to carry out analysis of problems.[125]

Beneath this statement is a brief description of the Division's Computer Engineering activities, noting that members of this section design and develop computers and information processing systems having special relevance to the nuclear sciences.

More significantly, the activities of the Digital Computing Center -- the primary service component of the Division -- are relegated to the last paragraph of the Preface. In this revised description of AMD work, it is members of the Computing Center's Applied Programming section, *not the Consulting and Research section* that work with scientists from other divisions to formulate and define problems for solutions on digital computers.

In slightly over ten years, the entire notion that computational science would be a collaborative venture, built around the work of mathematicians, programmers, and scientists, had been turned on its ear. The hybrid area proposed by Miller was not populated by applied mathematicians, but rather by programmers. Mathematicians, whose professional and intellectual interests were often at odds with the Division's service mandate, were quick to identify themselves with computer science since it provided a recognized disciplinary umbrella under which they could work. Although not

[125] Applied Mathematics Division Summary Report, July 1, 1966 through June 30, 1967. AEC Research and Development Report, 1967. ANL-7418 Mathematics and Computers (TID-4500). MCS Archives.

disinclined to work with researchers from other divisions, the vast majority of mathematical analysis and model development was now relegated to the more blue-collar arena of programming. These trends are evident in the evolving organizational charts and annual reports and are thus valuable as a way to trace the institutionalization of computer science and applied mathematics research. Yet we still need to look more closely at the relationship between the AMD and other scientists in the division in order to form a more complete understanding of the motivations for some of these changes.

In 1964, in the midst of these changes at the AMD, William Miller abruptly resigned as division director in order to take a position in Stanford's newly created Computer Science department. Wallace Givens was quickly promoted to director, whereupon he strove to fulfill as much of Miller's vision for the division as he could under rapidly changing circumstances. Whereas Miller had been a physicist by training and personified the kind of applied mathematician who would work in the hybrid zone, Givens was a mathematician and tended to support efforts to promote the disciplinary independence of mathematicians more forcefully.

In the face of attacks by other scientists at Argonne to improve computational services and cut research activities, Givens dug in. By 1968, such complaints had reached the attention of the Review Committee and Givens was forced to address the issue directly. For the most part, the Review Committee was sympathetic since its members tended to be applied mathematicians themselves. In his response to their evaluation, Givens reiterated the Committee's recommendation that "it is more essential than ever that the increasing burden of computational work on the AMD be paralleled by the acquisition of deeply trained mathematicians…" but then added "the nation is not well served by every more use of computers unsupported by adequate research and study *independent of the immediate pressures from users*."[126] (my italics). Echoes of Hilbert and the Göttingen program seem implicit here -- applications might emerge from the work of applied mathematicians and computer scientists, but it is not necessarily their job to find them.

The dramatic reorientation during 1963-1968 of how the mission of the Applied Mathematics Division was interpreted by its members can be ascribed only in part to the emergence of a distinct professional identity for computer specialists. A second impetus behind these changes was the incredible rate of technological innovation within computing. In general, these innovations had a more profound influence on consumers of computational tools. At Argonne, as elsewhere, the development of smaller, less expensive computers meant that computing became increasingly decentralized. The installation of the CDC 3600 in September 1963 ended the monopoly that the AMD had on computers at Argonne. While the central processor of the 3600 was situated within the AMD, two CDC-160A computer systems were installed in the High-Energy Physics Division Building and the Reactor Engineering Building primarily to handle input-output tasks for the 3600. These "minicomputers" were capable of handling many of the routine computing tasks of their respective divisions, and communicated with the central unit at the AMD only for compute-intensive data reduction work or for large computer runs that required a larger memory than available in the 160A.[127] In addition, when the AMD moved to its own building in 1964, the IBM 704 and an IBM 1401 were left in the

---

[126] J.W. Givens, to Winston Manning. "Comments on the 1967 Report of the Review Committee of the Applied Mathematics Division," March 13, 1968. folder 1, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.
[127] Dr. A. V. Crewe, to George W. Beadle. "Report on Review Committee for Applied Mathematics," April 16, 1964. Box 11: Review Committee for Applied Mathematics, RG 326: Policy Advisory Board, 1957-67, GLFRCNARA.

Physics building, and continued to service an IBM 1620 computer system that had been installed in the Chemistry Division.[128]

The importance of the end of the AMD's monopoly on computers should not be underestimated, as the acquisition of computers by other divisions signaled a distinct shift in the relationship between the AMD and scientists at the Laboratory. For the first time, physicists, reactor engineers, and chemists enjoyed the relative independence that came with having access to their own computer system. For many routine computing tasks, it was no longer necessary to submit a job to the central facility at the AMD and then wait several hours (or days) for the answers to come back, or as often happened, a note indicating that the program failed to run correctly. Instead, they could submit the job through the satellite systems located in their own Division. At the same time, the continued development of the FORTRAN programming language, together with vastly expanded libraries of mathematical subroutines meant that scientists felt more confident trying to do their own programming, thereby cutting out the AMD entirely.

The decentralization of computer systems had several other unforeseen consequences that struck directly at the autonomy and organization of the Applied Mathematics Division. First, as computer systems became less expensive throughout the mid-to-late 1960s, it became possible for other divisions at Argonne to purchase their own computers (especially the Digital Equipment Corporation's PDP line) without consulting with the AMD at all. However, once the computer was installed, the Division purchasing the system insisted that people within Applied Math maintain the machine, as well as produce the necessary software to enable the new machine to use Laboratory

---

[128] Applied Mathematics Division, Argonne National Laboratory. 1964. Folder 4: Press Releases, Box 2, Computer Oral History Collection #196, Series 2: Supplemental Docs, subseries C: Henry Tropp Files, American History Museum, Smithsonian Institute. In addition, by 1968 the biological and medical research division had a GEL 225, the Solid State Science Division had an IBM 1130, and the Accelerator Division had an SEL 810A.

programs and interface with its mainframe computers. Pressure on AMD directors to support various machines that they had no say in acquiring severely strained the limited resources of the Division and was perceived by its members as an abuse of the Division personnel. Initially, the AMD attempted to reassert its control over computing equipment at the lab by acting as advisors to other divisions wishing to acquire computers, but this service lacked official sanction within the Laboratory.

The extent to which the proliferation of incompatible systems was taxing the operations of the AMD went unrecognized by Laboratory management and the division's Review Committee. As pressure continued to mount, Wallace Givens felt it necessary to address this issue directly in a response to the 1965 Review Committee report. Noting that advising other divisions on the acquisition of computers had become a substantial activity of the AMD he wrote, "The Review Committee may have considered this [advising] function to be occasional but in fact it requires a large amount of effort. Although the scientific component is less evident here, the wise and prudent management of expenditures for acquisition and operation is in fact the core activity of the Division and must be given due emphasis in management and planning for the Division."[129]

Givens, while lamenting the troubles caused by the acquisition of computers by other divisions, was sympathetic to the desire of scientists and engineers to gain access to a computer without having to navigate the AMD's bureaucracy. Thus, in 1964-65, building on the growing interest in interactive computing among scientists, he directed a group of computer engineers, systems designers, mathematicians, and programmers to begin work on a small time-sharing system that would allow scientists to submit their programs to the main computing facility by using remote terminals distributed across the

[129] J.W. Givens, to W. M. Manning. "Comments on the 1965 Report of the Review Committee of AMD," Sept. 21, 1965. Box 11: Review Committee AMD, RG 326: Records of the Policy Advisory Board, 1957-67, GLFRCNARA.

laboratory. “Time-sharing” was a technique by which users had the illusion that they had an entire computer with attendant software at their disposal. This included any languages, data, or subroutines that a scientist needed to complete their work. What made this concept feasible was the difference between the speed at which humans work and think and the speed at which computers could fetch and execute hundreds of simple instructions. In the few millisecond between keystrokes, or minutes while a user was thinking, the computer’s processor could handle all the chores required by one user and still handle those of another. To each person using a time-sharing system, it would appear that the computer was theirs alone. In terms of implementation, the difficulty lay in having the computer’s processor keep track of different jobs and different instructions simultaneously -- a task which threatened to overwhelm the computer’s capabilities. Yet the desire to develop time-sharing systems was widely held in the early to mid-1960s, finally culminating in the funding in 1963 of Project MAC (“Man and Computer” or “Machine-Aided Cognition”) at MIT by the Defense Advanced Research Projects Agency.[130]

Givens saw the development and deployment of a time sharing system as serving a dual purpose. First and foremost, he hoped that it might ease the demands being placed on the AMD for faster turnaround time. But a second goal was to create a cutting-edge research project that would engage his mathematicians, computer scientists, and computer engineers. At the same time, the Remote Access Data System (RADS) was a

[130] For a detailed description of Project MAC, see Arthur L. Norberg and Judy E. O'Neill, Transforming Computer Technology: Information Processing for the Pentagon, 1962-1986, ed. Merritt Roe Smith, Johns Hopkins Studies in the History of Technology (Baltimore: The Johns Hopkins University Press, 1996), pp.68-118.

long term solution and one that would not reach the early stages of production until 1968.[131]

What Givens probably didn't expect was the way that the promise of time-sharing systems changed the way that scientists perceived the computing activity. As one scientist reported to Argonne's Computer Needs Committee:

> "…time sharing represents the beginning of a new era in the impact of the computer on the scientific community. It essentially is the 'Henry Ford' of computing and brings the potential of interactive sophisticated computing to the 'common man' who heretofore felt that it was too difficult or inconvenient to really concern himself with opinions on the architecture of computing services."[132]

As scientists gained more direct access to computers, they demanded greater say in the selection and implementation of the Lab's computer systems. By 1968 scientists regularly complained that the AMD failed to circulate comparison and evaluation information on the various time-sharing systems available commercially to other divisions. As a representative for computer users at the lab made clear in a letter to the AMD's Computer Needs Committee, it was necessary to develop some form of communication between "the scientist and AMD… so that the desires and needs of the research scientist not only influence system design but are anticipated."[133] The implications of these complaints are clear. First, scientists now felt empowered to make decisions about computer equipment and second, they believed that it was the AMD's responsibility to provide objective information about different systems and then allow users to determine which system best suited their needs. Thus, time-sharing served to undercut some of the authority previously held by the AMD in terms of computer system selection -- an outcome that Givens could not have anticipated.

---

131 Semiannual Report of Accomplishments in the Use and Management of ADP. Argonne National Laboratory, 1968. Computer Needs Committee, Jan 1970, Box 17: Computer Information, RG 326: Lab Director's Project Files, 1955-1970, GLFRCNARA.

132 A.C. Wahl, to Richard Adams. Oct. 8, 1968. Computer Needs Committee, August 1968-Dec. 1968, Box 17, RG 326: Lab Director's Project Files, 1955-1970, GLFRCNARA.

133 Ibid.

In the meantime, scientist's demands for improved computational "service" quickly evolved into direct competition with AMD activities. Beginning in the mid to late 1960s, several scientific divisions at Argonne began to hire their own programmers, thereby removing the need to use the applied programming section at the AMD. In general, scientists and engineers within the various divisions viewed this as an enlightened initiative on their part, intended to improve the turnaround time of computations. Although AMD management had tried to work out a solution to scientists' complaints, at the heart of the issue were profoundly different perceptions of how the computation service should be run. These perceptions, in turn, were informed by the way computing services were funded at the lab.

Every year the AMD planned its budget, including the number of programmers it would retain, based on the projected use of the computer facilities by other divisions. Since the AMD was reimbursed for applied programming services through direct charges to the other Divisions using computer services, there was no additional money to "stockpile" programming talent in case of an emergency. If, as often happened, demand for application programming from one division lagged, programmers would be reassigned to projects in other divisions. While this flexibility allowed the AMD to put people where they were needed, it also meant that the division was less able to meet unexpected and urgent demands from other divisions. Invariably, this situation led to user dissatisfaction and demands that programmers be reassigned to work exclusively within a particular division at the lab.[134]

Not surprisingly, this suggestion met with immediate resistance from AMD directors who were faced with an initiative that would seriously handicap their ability to

[134] Givens, to Winston Manning, "Comments on the 1967 Report of the Review Committee of the Applied Mathematics Division," March 13, 1968, folder 1, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections

do independent research. From an economic standpoint, the AMD had to be able to pay for the huge, expensive computer systems that it maintained and on which computer scientists did their research. It was standard at the AMD, and at almost any other large computer installation (business or scientific), to charge users for computer time and programming services. This was the only way to afford the incredibly high cost of computer equipment. Thus, if scientists in other divisions reduced the amount of programming and computer time they used, it would create a severe financial hardship for the AMD.

Moreover, echoing Miller from 1961, Givens insisted that the professional development of programmers was best handled by a division set up to look after their needs, especially in terms of advanced training. As Givens noted, during budgetary crises, other divisions would be much more likely to let the Applied programmers go and do without their services than lay off members of their own scientific staff. He warned that such actions would be highly detrimental to the AMD's mission to provide computational support to the entire lab.[135]

For the most part, Givens arguments went unheeded by users, and increasingly after 1967 Argonne divisions proceeded to hire their own applied programmers. By 1969, issues connected to the applied programming services of the AMD had reached a critical point. In response to the maneuverings by other divisions, Givens proposed three changes to the operations of the computer services functions in order to salvage what he could of the AMD's programming duties. First, beginning in 1969, programming services would be provided for large scale and long term projects, but smaller jobs would be accepted only as feasible. Givens also proposed that the AMD keep on hand several

[135] Ibid.

more personnel than the average projected needs from user divisions in order to meet fluctuating demand. When these programmers were not assigned to jobs arising from users, they would devote their time to over-all system improvement. Second, as recruitment permitted, the AMD would expand its consulting services in areas of systems problems, data management, and numerical methods. Where needed, Givens indicated that AMD consulting personnel could “get directly involved in the effective solution of the problems posed by user divisions.” Finally, if divisions persisted in hiring their own applied programmers, it would be their responsibility to bear “the full and considerable responsibility for recruiting, hiring, evaluating, and sustaining” these people.[136]

The response from consumers was in general negative, and the attitude of the members of the Solid State Science division was fairly representative of other divisions.[137] On the first point, Solid State was already doing the bulk of their applied programming in-house. Thus, they were concerned that the AMD would pay for the cost of the proposed extra programmers by increasing the cost of computer time. Furthermore, the nebulous task of “over-all system improvement” was seen as make-work that was not really needed. Instead of applied programmers, scientists wanted more systems programmers who could help them automate or process experimental data, and insisted that the cost of these systems people be separated from the cost of applied programming. Better still, they wanted more effort put into user-friendly software systems, time-sharing, and quality instruction manuals. If this could be accomplished,

[136] Committee on ANL Computer Needs, to Laboratory Director's Office. "Recommendations for Future Computer Needs at ANL," Feb. 20, 1969. Computer Needs Committee Jan. 1969-Dec. 1969, Box 7: Computer Information, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

[137] Givens’ proposal was also not well received by the Budget Manager of Argonne, who concluded that the recommendations would inject considerable instability into the support of the Applied Programming Section. See E. C. Weber, to R. M. Adams. "Comments Regarding Draft -- Computer Needs Committee Recommendations Concerning Applied Programming Services," March 28, 1969. Computer Needs Committee Jan. 1969-Dec. 1969, Box 7: Computer Information, RG 326: Records of the Atomic Energy Commission, Argonne National Laboratory, GLFRCNARA.

“scientists could be encouraged and helped to do their own programming” and the need for applied programmers “could be markedly reduced.”[138] At its most basic level, scientists wanted the skills and the tools to pursue computational science on their own, with little, if any, collaboration with the mathematicians, computer scientists, or programmers of the AMD.

Likewise, Givens’ proposal to expand the consulting services needed to be scrutinized closely. Users at the Solid State Division felt that the consultant’s value lay in being able to “answer *specific* questions and give advice on *specific* problems as they arise.” While this was still seen as a valuable and needed service, scientists balked at the suggestion that members of the AMD get “directly involved in the effective solution of problems posed by ‘user’ divisions.” Citing a tendency for members of the AMD to make a simple problem appear “unnecessarily complicated”, one scientist of the Solid State Division wrote:

> “On the basis of past experience, I don’t trust the applied programmers (except for specific questions and programming tips) unless I understand the numerical and program organization problems as well as they do, and then I find that it is more efficient to either do the programming myself or find a collaborator within the division to do the programming.”[139]

As for Givens’ third recommendation that the AMD not support applied programmers within the divisions, the Solid State division saw this as “a reluctant concession to the pressure for the divisions to maintain their own services, phrased in a manner to discourage such attempts.” Drawing on personal experience, this scientist went so far as to compare the applied programming group of the AMD to a research organization doing contractual research and commented that this kind of environment was “inefficient,

[138] T.L. Gilbert, to O.C. Simpson. "Memo of 3/18/69 of R.M. Adams to Distribution from Computer Needs Committee to Laboratory Director on Recommendations Concerning Applied Programming Services," March 19, 1969. Computer Needs Committee Jan. 1969-Dec. 1969, Box 7: Computer Information, RG: 326 Records of the Atomic Energy Commission, Argonne National Laboratory, GLFRCNARA.
[139] Ibid.

expensive, and demoralizing." While acknowledging that there were considerable problems involved in finding quality programmers to work in each division, "when the problems of divided authority, poor communications, and lack of close continuing contact between scientist and programmer are taken into account… it is better to have the divisions responsible for providing their own programming requirements."[140]

The conflict between producers and consumers did not go unnoticed by the Review Committee for the AMD, and its 1969 report focused specifically on several of these issues. However, the Committee felt that the root of the problem was essentially a communication issue, not a breakdown in the desire for collaboration between scientists and the AMD. Thus, they recommended somewhat superficial modifications to the management structure of the Division such as creating more middle managers to facilitate communication throughout the organization. With over 170 people involved in diverse research, development, and service activities within the AMD, the Committee believed better management of this unwieldy group would translate into better relations with other scientists. Most importantly, the Committee believed that better communication would sooth users by making them feel that they had a greater say in the acquisition of new computer systems.[141]

Clearly the Review Committee was seeking some middle ground between scientists and the AMD. The demand by scientists for greater control and access to computing resources was recast by the committee as dissatisfaction arising from poor communication. But the response by Givens to the report also points to the deeper issues at work here.

[140] Ibid.
[141] Ibid.

Quite directly Givens disagreed with the Committee's characterization of the problem, arguing that dissatisfaction with AMD activities was an outgrowth of opinions formed by scientists in the days when computing was considered simply a service activity. While he was willing to consider some shifts of administrative duties and even a substantial realignment of the AMD to improve communication, Givens reminded the reviewers that "the establishment of the Digital Computing Center within the AMD was always regarded as an expedient." To computer specialists, the service duties of the division made sense in the early days of computing at Argonne, but as computer science matured into its own discipline and more jobs could be handled routinely by scientists and programmers within other divisions, computers scientists and mathematicians no longer wanted to be bound by service duties.

Givens also tried to normalize the feelings of his computer specialists in regards to the complaints of other scientists. While the Committee had recognized that user satisfaction at Argonne was better than most centers around the country, Givens used this to point out that this demonstrates that in a national context, "very substantial dissatisfaction with programming and computer support is normally encountered and that we must expect some level of controversy over services provided."

To illustrate the basic kind of problem extant between producers of computational tools and their consumers, Givens pointed to two places in the report where the Committee praised John Gabriel, a computer scientists, for his efforts to improve the work of the Applied Programming section. While the Committee found his work to be of high-quality, Gabriel had been the focal point for criticism from scientists for the "short shrift [he] has given to some users to whom it appears he has made it clear that he thinks they don't know what they are doing in their specification and use of programming assistance.” While Givens acknowledges that it important for the AMD to improve public relations with its “customers”, he forcefully argued that at the same time the “sharpest and most driving minds” must be given responsibility and encouragement.

Diplomatically, Givens was making the argument that the real issue was not one of communication, but rather a contest between realms of expertise:

> “A great deal is now beginning to be known about computing and programming, although documentation is recognized as drastically behind the state of the art. In such a situation experts are likely to be abrupt and impatient with those who do not acknowledge the existence of highly specialized knowledge. The problem has no easy solution.”[142]

Computer users, Givens suggests, failed to recognize or validate the computer sciences as a legitimate discipline with its own technical language, methods, and research agenda. Instead, they were solely concerned with service, which was only a small part of what the division did. While he continued to insist that programmers would be best served by remaining centralized in the AMD, Givens clearly felt that these duties were ancillary (and distracting) to the work of mathematicians and computer scientists within the division. If another means could be found to pay for and support the system maintenance of the Laboratory’s main computers, Givens would be happy to jettison the entire Applied Programming section.

As the decade of the 1960s wound down, pressures external to the national laboratory system of the AEC began to force changes on the programs and mission of the individual labs. The AEC faced intense budget cuts primarily as a result of President Lyndon Johnson’s simultaneous pursuit of the war in Vietnam and the Great Society, and the energy crisis of the early 1970s forced another round of critical funding cuts for the national labs, forcing them to make hard choices in their pursuit of programs. In

[142] J.W. Givens, to W. M. Manning. "Response to the Report of the ANL Review Committee for the Applied Mathematics Division," Feb. 13, 1969, 1969. Folder 1, Box 153: AMD Reports, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

addition, a distinct antiscientific backlash fueled by the counterculture and directed primarily at the development and use of atomic weapons and reactors made national labs targets for protest. Most importantly though, the main justification for creating the national labs -- to develop nuclear weapons and reactors and to provide extremely expensive equipment for researchers -- had lost its impetus.[143] In order to remain relevant, the national labs were forced to develop new missions and new programs in the 1970s.

At Argonne, budget cuts fell heavily on the Applied Mathematics Division, and affected the division in several important ways. Most notably, reduced funding led to the complete cancellation of the AMD computer engineering program in 1972. Henceforth, the division would no longer build custom made computer systems but instead focus on applying commercially produced equipment to the needs of ANL research programs. Almost a decade of research and development in the area of image processing and automatic film scanning for biology and physics applications was cut off, and with it one of the key areas for interdisciplinary research between computer scientists and other researchers.

At a more basic level, lack of funding meant that the Computer Center was unable to afford a new central computer to meet the demands of user divisions. Although the division had been able to secure an IBM System/360 Model 50 and 75 in 1966 to augment the aging CDC 3600, these systems were sorely overtaxed by 1969. With less than 5% of the total ANL budget going to the AMD and less than 2/3 of this amount allocated to the Computing Center, the Review Committee noted in their 1970 report that "it was doubtful that even a second class computing facility could be supplied for the

---

[143] Westwick, The National Labs: Science in an American System, 1947-1974, pp. 269-98.

entire ANL mission."[144] A lack of support for the development of time-shared remote terminals and process-support computers followed, leading to further dissatisfaction with the computer service by users. In response, other divisions continued to acquire their own computers (especially the Sigma-7 and PDP-10) further reducing the amount of money the Computing Center received by charging users for computer time.

Ironically, the activities of the Applied Programming section were partially responsible for their own problems. "In some sense" the Committee observed, "these people have worked themselves out of a job by giving training to other divisions and developing computer 'expertise', of a sort, in the user's group. Now, with budgetary problems, the divisions retain their 'own' people and cut of the Scientific Applications people of the AMD." The ability of individual divisions to handle their computing needs in-house further reduced the amount of money available to the AMD since they supported the Computer Center through charges to users. Shockingly, the Review Committee recommended that 2/3 of the present Scientific Applications staff be moved to the divisions they had historically served, a move that the AMD had long resisted. Left behind in the AMD would be a small core of about ten programmers charged with the mission of "making itself an alter-ego for the ANL user community and acting in that capacity to 'smooth' the interface between a centralized facility and the users." In essence, the Committee was suggesting that the AMD do organizationally what Computer Scientists had done professionally -- shed their service duties.

[144] Quotes in the following paragraphs come from Herbert B. Keller, to Edward H. Levi. "Review Committee Report, Applied Mathematics Division, Argonne National Laboratory," July 8, 1970. ANL Review Committee for the AMD, B93-00147, AMD Archives, Argonne National Laboratory.

Conclusion

Crippled by budget cuts, the situation at the AMD disintegrated tremendously in 1970. Tired of fighting administrative battles and wanting to devote himself to research, Givens resigned as Director of the AMD. In a moment of understatement, the Review Committee noted that Givens' resignation, compounded by budget cuts and the termination of about 10% of the division left the AMD "somewhat unsettled". The selection of a new director was the top priority, but the Committee cautioned that the desired attributes of the new director depended upon the mission and the role of the AMD within ANL:

> "These should be carefully reviewed at the highest level bearing the long range goals in mind, before a choice is made. For example it now appears that the Computing Center has become the salient feature of AMD, clearly shifting its historical mission. This is unfortunate in our opinion. A conscious move in this direction, or preferably away from it, has much relevance to the choice of a new director."

Sensing a "general malaise" about the long range goals of the ANL and the, hence the AMD, the committee recognized that no matter the new missions developed for the lab, large scale scientific computing and data handling would continue to be a basic requirement. In this context the committee warned sternly 'it would be a serious error to allow AMD to become mainly a service computing center." The ability of the AMD to contribute to the scientific mission of ANL required continued research in the full range of the computer sciences. Choosing a Director to guide the AMD through these difficult was thus of paramount importance.

The person tabbed by Argonne administrators to replace Givens says much about the growing power of computer users to shape the direction of the AMD. Richard Royston, a high-energy physicist who had been the head of the HEP Applied Programming section of the AMD in 1964-65 was selected as the new Director. Royston represented the new archetypical scientific computer user -- an able programmer with little need for either Applied Programmers or the mathematical consultants of the AMD, but doing research that required the high-performance computing equipment of that division. Focused on increased service, for the first time in AMD's history, they had a director disinclined to support, at least initially, the expansion of the computer sciences.

Between the ascension of Royston to the helm of the AMD and the budget difficulties, the research activities of computer specialists within the division were either sharply curtailed or reoriented away from long range goals in favor of short-term applied projects. This trend was noticed by the Review Committee in 1972 which admonished the AMD for allowing fiscal concerns to "dominate the direction of the work to too great an extent."[145]

Morale continued to suffer in 1973 and 1974, as layoffs hit the division particularly hard. Simultaneously there was increased pressure for the AMD to streamline its mathematical and computer science research initiatives so that they could show clear contributions to the research programs of the AEC. Nonetheless, the Review Committee for 1973 felt that the program was still lacking a coherent vision for mathematical research. While praising the division's work in the production of general purpose mathematical software, the Committee charged AMD directors to find a relevant field in which to apply the Division's mathematical expertise.

Ironically, one of the roots of trouble for the AMD provided a new research domain for its mathematicians and computer scientists. The acquisition of minicomputers by other divisions at Argonne had driven a wedge between the historical mission of the AMD to provide computing services in an interdisciplinary environment

---

[145] William B. Cannon, to Robert Duffield. "Report of Review Committee Applied Mathematics Division," Jan. 14, 1972. ANL Review Committee for AMD, 1958-1980, B93-00147, MCS Archives.

and its user community. Although the AMD had tried in 1968 to reassert some control over the purchasing of computers by other divisions, their responsibility extended only to the assessment of digital computers for their "feasibility and compatibility."[146] They had no veto power over the acquiring division, which usually went ahead and bought the equipment anyway.

However, the very proliferation of these minicomputers opened a new and somewhat unexpected research area for computer specialists. The development of ARPANET in the late 1960s represented the first steps in creating a widespread computer network and focused increasing attention on the creation of smaller, local networks that could then be connected to ARPANET.[147] Argonne, which was scheduled to become an ARPANET node in 1974, was presented with an opportunity to kill two birds with one stone. Noting that Argonne, " as a typical laboratory, must be suffering from the standard proliferation of minicomputers as experiment controllers and special purpose computers", the division's Review Committee suggested that stepping up support, both in program preparation and in data collection facilities, would be an answer to the "increasing and ubiquitous mini-computer population of the laboratory."[148]

More importantly, the Committee pointed out that the proliferation of minicomputers provided "a new dimension to a number of activities of the AMD" because of the possibility for *networking* these computers together to take advantage of the advanced computing resources available within the AMD. Such computer networking would provide a new field of research for the mathematicians and computer

---

[146] Applied Mathematics Division Summary Report, July 1, 1967 - June 30, 1968. Argonne: Argonne National Laboratory, 1968. ANL-7521 Mathematics and Computers. MCS Archives.
[147] Janet Abbate, Inventing the Internet (Cambridge: the MIT Press, 1999).
[148] W. R. Sutherland. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1973. folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign Special Collections.

scientists within the division, while offering the potential for revitalizing cooperation between the AMD and scientists in other divisions.[149]

In addition, the new emphasis on computer networking was one answer to the prohibitive cost of purchasing new state-of-the-art computer equipment. By 1973 networks were being considered as a technical solution to the economics of computing. If local networks of minicomputers could be linked to increasingly powerful computational resources whose physical locations were irrelevant, it would provide scientists with computing power far in excess of what could be afforded by a single institution. Furthermore, computer networks could enable researchers with similar projects to share data and collaborate in ways previously untenable due to financial, technical, or physical constraints.[150]

Computer networking also dovetailed nicely with the recently streamlined research strengths of computer scientists and mathematicians within the AMD. In particular, the Division had developed a world-class effort in the development and implementation of numerical software which will be discussed in some detail in the next chapter.

A second research project emerging from the shift to networked computers was a mathematical and computer science investigation of distributed computing. In this case, the AMD proposed that computer networks might allow researchers to selectively utilize various components of the overall network in order to carry out a specific application. If, for example, one computer on the network was well-suited at carrying out matrix manipulation and another was highly effective at handling graphical data structures, then a researcher could make use of the two machines independently. If the details of

---

[149] Ibid.
[150] Argonne National Laboratory Applied Mathematics Division: A Synopsis of Long Range Plans.

implementation could be worked out, in a network environment it would be possible to select components of different computers based on their particular virtues. The hope was that this research would lead to a more efficient use of computational resources while greatly enhancing the overall computational power available to scientists and engineers.[151]

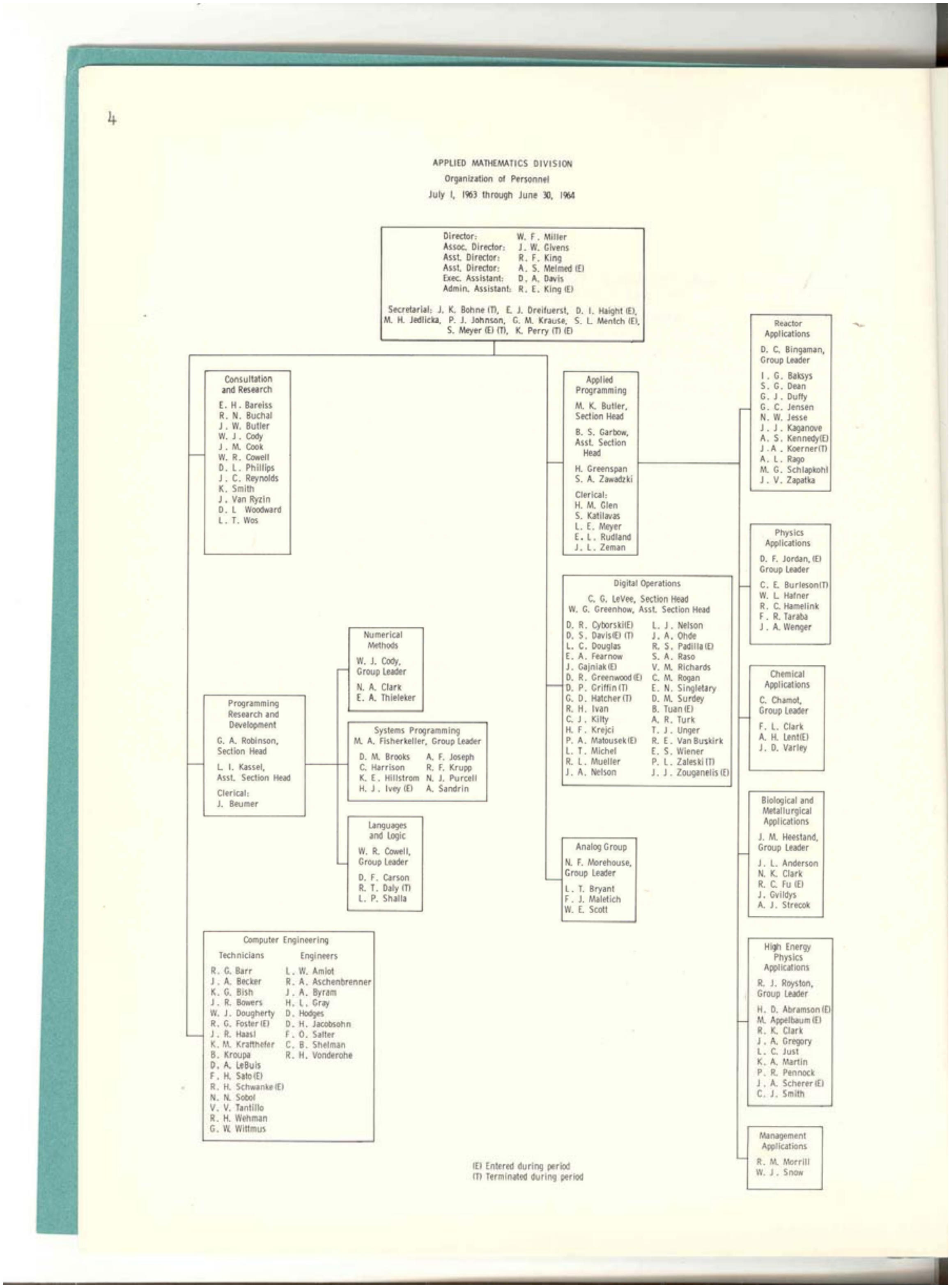
The cutting-edge nature of the research associated with networks proved to be an intellectual and professional salve for many of the mathematicians and computer scientists within the AMD. While work still continued in the established areas such as linear and non-linear analysis, Monte Carlo simulations, partial differential equations, matrix calculations, and automated reasoning, the open realm of networks provided opportunities for computer specialists to follow their own research interests. More importantly, networks provided a new opportunity for some interdisciplinary collaboration between producers and users of computational tools. Yet the character of this collaboration was different from what Moll Flanders and Bill Miller had originally imagined. Their vision had been informed by an understanding of current computing technologies and their recognition that fundamental mathematical research was crucial to the effective use of computers. Miller's proposal of the hybrid area was an attempt to make applied mathematicians central to an interdisciplinary, collaborative, and computational approach to scientific and engineering research. The existence of the hybrid area was conceived against a backdrop of longer historical trends towards the increased mathematization of all the sciences, and its fruition was seen in the development of electronic digital computers that could crunch numbers at unprecedented speeds.

[151] Ibid.

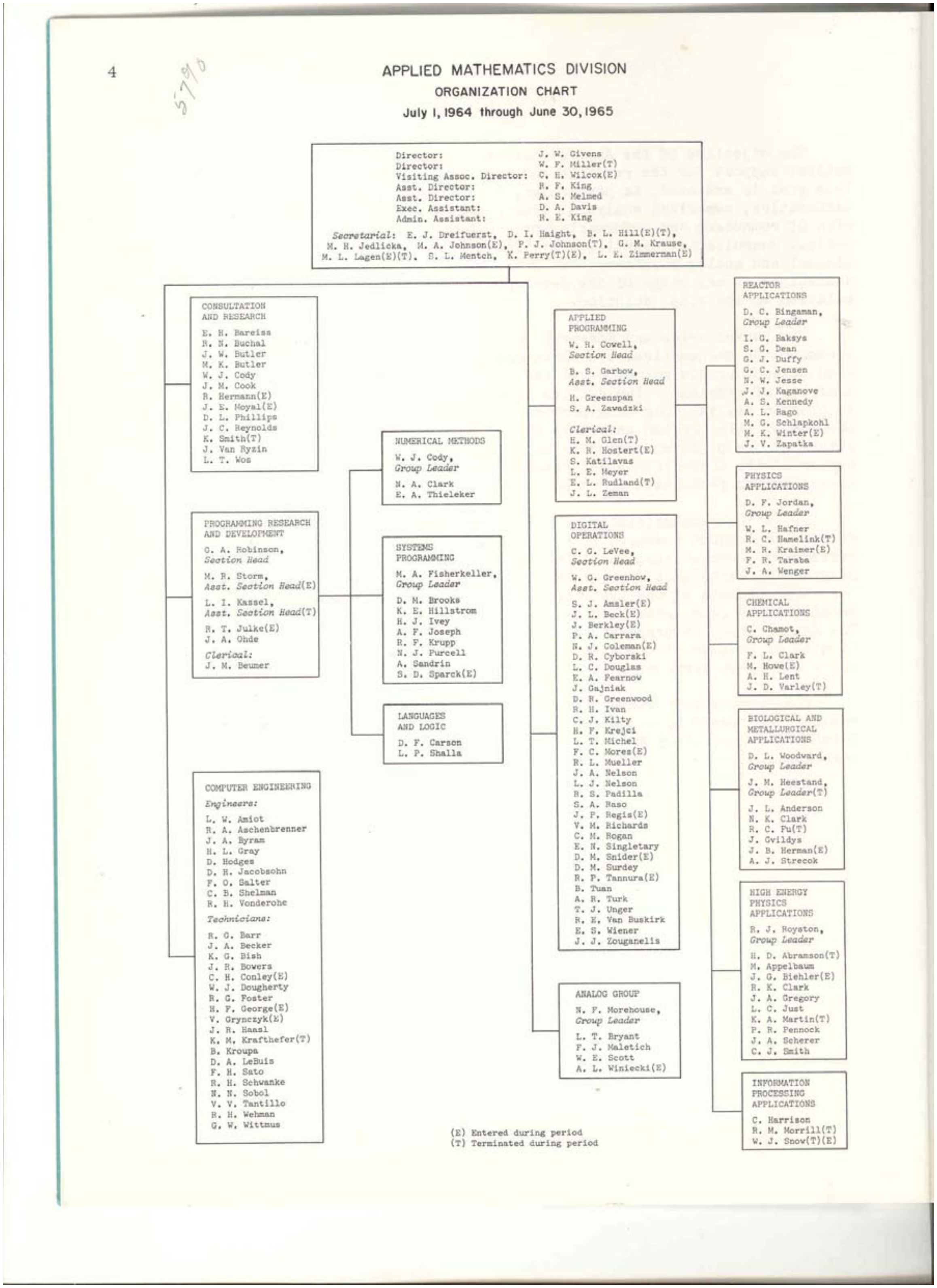
Yet Flanders and Miller could hardly have anticipated that within fifteen years, small computers with power far in excess of the room-sized mainframes of the 1950s would be available and affordable to individual divisions at Argonne. Beyond the incredible advancement in hardware technologies, the development of high-level programming languages and huge libraries of mathematical subroutines enabled scientists and engineers to bypass the collaborative structures originally built into the Applied Mathematics Division. At the same time, the emergence of computer science as a distinct discipline provided a professional status which mathematicians and computer specialists successfully leveraged in order to establish considerable autonomy in their research activities. In the process, both professionally and organizationally, computer scientists and mathematicians at the AMD effectively shed their service duties to other divisions at Argonne, which had the effect of limiting opportunities for collaboration.

Overarching the technological and professional changes and working against the construction of the hybrid zone were basic tensions between the producers of computational tools at Argonne and the computer user community. The core point of contention was a fundamental difference of opinion between producers and consumers as to the role of the AMD and the computer specialists within the division. Consumers looked at the AMD as providing a service to the research community and pushed for some control over the activities of the division under the rubric of improving its services. In contrast, the producers of these tools considered computing a research area in its own right, and thus struggled to stake out a measure of independence and autonomy separate from their mission to provide computational services. What became clear throughout the 1960s is that while both producers and consumers liked the idea of interdisciplinary collaboration, they each had specific ideas for how it would be implemented and the relative authority of the researchers within this collaborative venture. The result was a gradual split of computer science and mathematics on one side and what would become computational science on the other. As I have argued, the way these tensions played out can be seen, to a certain extent, in the evolution of the AMD's organization from 1956 to the early 1970s.

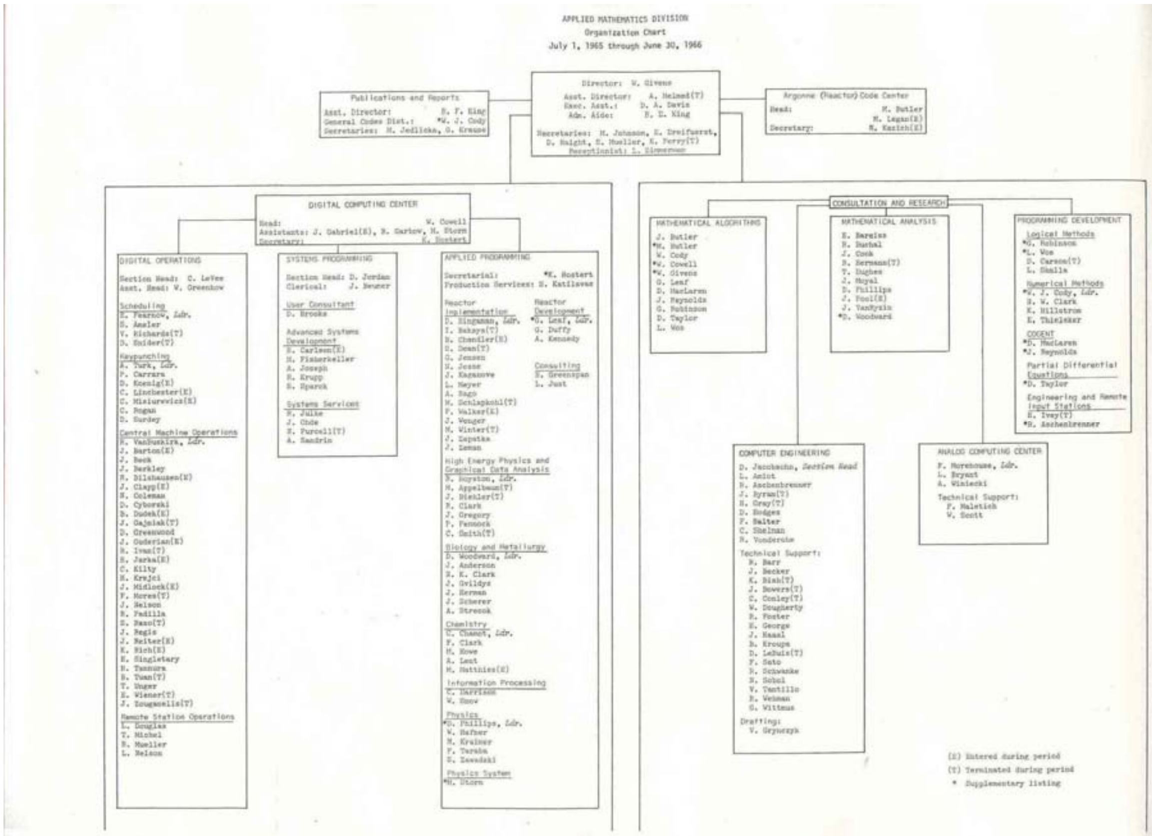
The harsh budget cuts, a change of directors, personnel layoffs, and changing national priorities at the beginning of the 1970s left the AMD with the need to reinvent itself and its mission at Argonne. With many of its original service duties, especially in terms of application programming, outsourced to various divisions, AMD mathematicians and computer scientists needed a new avenue in which to pursue cutting edge research while still being relevant to the mission of the national laboratory system. Networking was one such area that promised unique research opportunities and the potential to vastly increase the computational resources available to scientists and engineers. As will be seen in the next chapter, the development of "supercomputers" in the late 1970s and early 1980s provided new linkages between mathematicians, computer scientists, and computational scientists. By the mid 1980s, these groups saw common cause in terms of attracting the vast sums of money needed to pursue supercomputer research and applications. The creation of several high-performance computing initiatives on the national scale, culminating in the $5 billion Grand Challenges program of the 1990s, succeeded in creating the kind of interdisciplinary collaboration Miller had originally proposed in 1961.



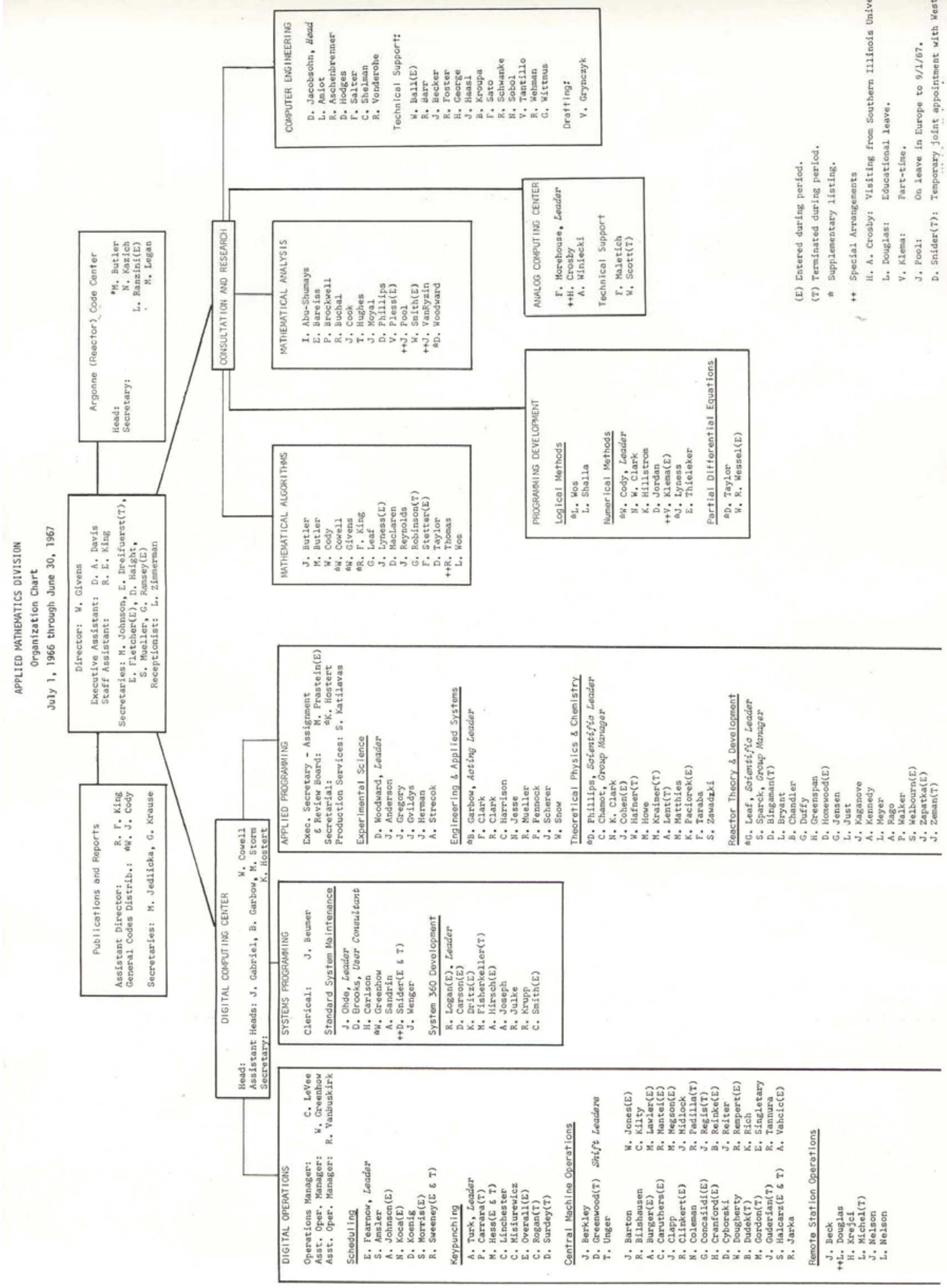
5. 1963-1964 AMD Organizational Chart



6. 1964-1965 AMD Organizational Chart



7. 1965-1966 AMD Organizational Chart



8. 1966-1967 AMD Organizational Chart

9. The Central Computing Facility of the AMD. This is the Control Data Corporation (CDC) 3600 computer system that was installed in 1963, replacing the IBM 704. As with computer facilities in industry, this one was on display and included observation windows and railings on which to lean.

10. This is the face of the AMD for most computer users at Argonne. Programs are dropped off here and then picked up later. Directly reflecting the ever present tension that existed between the AMD and its customers, the sign reads: “We are planning to be operating Thursday night (July 16) and on Friday night (July 17) and Saturday night and ALL DAY SUNDAY!! Please watch this space for further changes!!”

Chapter 3

Emergent Identities: High-Performance Computing and the Rise of Computer and Computational Science

> "The past decade has seen the emergence of a new way of doing science and engineering. This new mode of 'computational science' is poised to join theory and experiment as a third approach to solving scientific and engineering problems." Argonne High-Performance Computing Research Center, Argonne National Laboratory, April 14, 1992, p. 15.

> "Computing cycles provided by large-scale supercomputers are the raw material from which discoveries in computational science are made." Argonne High-Performance Computing Research Center, Argonne National Laboratory, April 14, 1992, p. 17.

When Argonne National Laboratory released its proposal to take part in the federal High-Performance Computing (HPC) Program in 1992, the claim that computational science was a distinct "third mode" of scientific inquiry, alongside theory and experiment, was already widely accepted by computer scientists, scientists, and politicians. In 1991, the four-year, $4.7 billion federal program High-Performance Computing (HPC) which funded collaborative projects between industry, academia, and government research laboratories was a testament to the arrival of computational science on the scientific world stage.

At the heart of the HPC were several key assumptions that guided the entire program. First and foremost was the belief that high-performance computing and communications were essential to national security as well as to the future economic strength and competitiveness of the United States. The second guiding principle was that research projects using high-performance computers would be *collaborative* ventures that included computer scientists and researchers from industry, academia, and government laboratories. Thus, the program was framed in the context of scientific and engineering “Grand Challenges.” As defined by the official Office of Science and Technology Policy’s “The Federal High Performance Computing Program”, Grand Challenges were “fundamental problem[s] in science and engineering, with potentially broad economic, political, and/or scientific impact that could be advanced by applying high-performance computing resources.”[1] Interdisciplinary collaboration in pursuit of these problems was not suggested, it was mandatory. A third principle informing the program was that the HPC would relieve the funding stress that had plagued the academic computing community for decades by providing the support and experimental equipment needed to train the next generation of computer experts.[2]

This chapter seeks to place the activities of the Applied Mathematics Division (AMD) at Argonne from 1970 to 1990 into the larger context of the emergence of computational science as the “third branch” of science and the Division’s eventual

[1] Office of Science and Technology Policy, "The Federal High Performance Computing Program," (Washington, D.C.: Executive Office of the President, 1989).
[2] National Research Council Computer Science and Telecommunications Board, "Computing the Future: A Broader Agenda for Computer Science and Engineering," (National Academy Press, 1992), p. 6.

participation in the Grand Challenges.[3] In it execution, then, this story jumps back and forth between events and research projects at the AMD and the broader cultural, social, and economic issues effecting computer science (CS) research in America. One of the main threads in this story, and one that has not been addressed in the historical literature of computing, is the extent to which the experimental component of computer science provided a platform upon which interdisciplinary collaborations could be built. As the discipline of computer science continued to mature in the 1970s and 1980s, a more nuanced conception of the field emerged that subsumed subfields such as numerical analysis, systems design, optimization, software engineering, and computer engineering into an overarching category of “computer science.” As part of this evolution, the ongoing debate as to whether CS was really a science or was mere engineering was settled to a large extent by the identification, by its practitioners, of theoretical and experimental components within the discipline. This intellectual and rhetorical move is significant for both economic and epistemological reasons. First, the goal of experimental computer science was to produce working systems that had immediate applications. And second, the demarcation of both experimental and theoretical traditions within computer science made it easier for its practitioners to draw parallels between their discipline and traditional scientific disciplines.

By recasting computer science research as inherently experimental, it encouraged federal funding agencies to support work that promised real improvements in scientific computing. Investments by the federal government in such equipment in the past had produced significant results, especially in the development of interactive computing and computer networks; but this support had dropped off considerably by the end of the

[3] Robert Pool, "The Third Branch of Science Debuts," Science 256, no. 3 (1992).

1960s as agencies like the National Science Foundation and the Atomic Energy Commission saw the development of experimental equipment more as an exercise in engineering, and thus better left in the hands of industry. As dollars were shifted away from ‘engineering experiments’ there was not a corresponding increase in the funding of theoretical computer science. The result, according to computer scientists, was a general decline in the vitality of computer science research in the United States with potentially disastrous implications for the nation’s scientific and technological lead. By resurrecting the experimental tradition of their discipline, computer scientists were able to convince funding agencies that support for their work paid dividends for all of the sciences.

The effort to identify an experimental tradition within computer science was more than an attempt to attract additional funding; it also touched on the larger epistemological debate as to whether computer science was a science or an engineering discipline. Computer scientists argued that, like physicists with their particle accelerators, they, too, needed access to cutting-edge experimental equipment to complement the theoretical component of their field. As money began to flow into experimental computer science projects, it reinforced the idea that computer scientists were just that -- scientists -- and should be supported in the same manner as experimental physicists, chemists, or biologists.

In the mid 1980s the professional aspirations of computer scientists were bolstered by the emergence of computational scientists who claimed that computer simulations represented a third mode of science alongside theory and experiment. Computational scientists came from every scientific discipline and provided a political cachet that was unavailable to the computer science community, namely spokesmen who were Nobel Laureates. While computer scientists often had a hard time being heard in the halls of Congress or at federal funding agencies, the nation's top physicists did not. Very quickly, computer and computational scientists wedded their fortunes together around the technology of supercomputers, as these machines were the primary tool of computational scientists and also the kind of experimental equipment computer scientists desired.

That computer and computational scientists found common cause in the mid 1980s owed much to events beyond their control. In late 1981 the Japanese announced their Fifth Generation Project, a joint government-industry-university program to apply supercomputers and artificial intelligence to problems of national import. This effort was a conscious attempt to create what today would be called an "information economy." The wide attention given this program by the American press effectively linked the issue of international competitiveness to computer science funding. Congress enacted the HPC in an effort to secure the economic, scientific, and military leadership of the United States in the face of the Japanese threat. As formulated, the HPC created the institutional and intellectual framework in which computer scientists and computational scientists collaborated in interdisciplinary research projects centered on the use of state-of-the-art supercomputers. As I will argue, computational science as instituted by the HPC was a new scientific methodology that reflected the values and structures generally ascribed to "Big Science": high-tech, collaborative, interdisciplinary, and very, very expensive.

Lessons learned: the Computer Science Institute and EISPACK

As the Applied Mathematics Division at Argonne entered the 1970s, it was beset by a host of organizational, technical, and economic problems. Under the directorship of the eminent mathematician Wallace Givens, the division had undergone several periods of restructuring in the mid to late 1960s intended, as I have argued, to separate the research activities of the computer scientists and mathematicians from the service activities of the computing section. Givens' weary resignation in 1970 and the subsequent elevation of the high-energy physicist Richard Royston to the position of Division Director seemed to signal a reorientation of the AMD towards more service at the expense of research.[4]

It was a testament to the strength of the math and computer science section that they were not dissolved or reintegrated into the research programs of other disciplines. Credit for this should be given to the leadership of Jim Pool, who originally joined the mathematical analysis section of the AMD in 1966.[5] Somewhat of an autodidact, Pool was born on a small farm in Kansas and spent time studying both engineering and physics at the University of Kansas before eventually choosing to pursue a degree in mathematics.[6] After graduation, Pool accepted a fellowship at Northwestern University where a physics professor arranged for him to spend time at Argonne National Lab. That summer (in the late 1950s) he was assigned to Argonne's chemistry division and spent

---

[4] Herbert B. Keller, to Edward H. Levi. "Review Committee Report, Applied Mathematics Division, Argonne National Laboratory," July 8, 1970. ANL Review Committee for the AMD, B93-00147, AMD Archives, Argonne National Laboratory.

[5] Applied Mathematics Division Summary Report, July 1, 1966 through June 30, 1967. AEC Research and Development Report, 1967. ANL-7418 Mathematics and Computers (TID-4500). MCS Archives.

[6] The following biography of James Pool is taken from an unpublished interview with him conducted by Thomas Haigh on contract with the Society for Industrial and Applied Mathematics. Interview July 14, 2004 at his office at CalTech in Pasadena, CA. Interview provided courtesy of Thomas Haigh, UW-Milwaukee.

time testing numerical software for them. After a year at Northwestern, Pool moved on to the University of Iowa to pursue his doctorate in mathematics with a strong emphasis in theoretical physics. He passed on a job offer to work at Bell Labs after graduation and instead went to work for the Ford Motor Company in their Mathematical and Theoretical Sciences Department. There he joined a group of six or seven other mathematicians and, in addition to his own research, spent about 1/3 of his time working with other scientists such as solid-state physicists.

As the Ford Motor Company became more interested in addressing issues of automobile pollution, members of his research section increasingly were transferred to other divisions in order to assist engineers in their work. Faced with the eventual dissolution of his group, Pool left Ford and accepted a one-year NSF Fellowship at Brandeis, after which he sought employment back at Argonne. After a brief stint in Europe, he joined the mathematical analysis section of the Applied Mathematics Division in 1967 and continued his research on the mathematical foundations of quantum mechanics. In August 1968 Pool left again, this time for academia and the math department at Amherst. Quickly disenchanted (to say the least), he noticed that the AMD was now looking for an Assistant Director and called Wallace Givens and asked for the job. By May of 1969, he was back at Argonne.

Pool's return to AMD was serendipitous for reasons that will be made apparent. When he arrived back at Argonne there was a well-organized effort by members of the Consulting and Research section and by the Argonne University Association (AUA) to establish an Institute for Computer Science. One component of this initiative was to produce high-quality numerical software packages for scientific computation. In the midst of deliberations about the nature of the proposed Institute, Pool was forced by economic reasons to make some decisions concerning what kind of expertise he wanted to foster within the AMD. His decision to develop a program in the creation of mathematical software was significant. While the Institute would eventually fail to attract even meager federal funding, the program to create numerical software became a resounding success. A comparison of the two efforts -- one a failure and one a success -- suggests much about the state of computer science as a discipline and its perceived position in the scientific arena. In particular, I want to suggest that the numerical software project succeeded because it was couched as an *experimental* rather than *theoretical* pursuit. The success of the mathematical software effort provides clues as to why computer scientists sought to distinguish an experimental component within the discipline over the next decade.

Sometime in 1967, members of the Applied Mathematics Division, in conjunction with several Midwestern university computer science departments, began to circulate ideas for the creation of a Computer Science Institute. It seems that the impetus for the institute was an NSF report released that year by J.T. Schwartz of the Courant Institute of Mathematical Sciences entitled "An Organizational Proposal for the Improvement of U.S. Computer Science."[7] The premise of Schwartz's paper was that the relatively dispersed nature of computer science research in America was leading to wasted and duplicative efforts. Moreover, he argued that the limited scientific resources devoted to computer science research were insufficient to provide theoretical support for large-scale

[7] J.T. Schwartz. An Organizational Proposal for the Improvement of U.S. Computer Science. Courant Institute of Mathematical Science, New York University, 1967. Personal Papers of Dr. Wayne Cowell.

computing or encourage the rapid development, documentation, and dissemination of new computing techniques.

In building his case for the establishment of a computer science institute, Schwartz embarked on a tour of the intellectual content of the discipline. In particular, he drew a distinction between mathematics and computer science. “Mathematics,” he wrote:

> “. . . stands to computer science as diamond mining to coal mining. The former is a search for gems; while such a search may involve the preliminary handling of fair masses of raw material, it finds its culmination, on the removal of dross, in an exquisite item, easily viewed once found. The latter is permanently involved with the necessity to deal efficiently with large masses of useful but relatively undistinguished material. It is necessarily a social rather than an individual effort.”

Thus, the pattern of funding dispersed individuals in math and computer science was ultimately ineffective because it failed to integrate these two different components of the discipline.

According to the Schwartz report, a central problem with computer science research in America -- both theoretical and applied -- was that it fell under two main branches: industrial and academic. In the industrial setting, which included application and systems programming in service bureaus and by computer manufacturers, the goals were often too narrow. The tendency in this work was to tailor the research to specific machines rather than look for generalizable techniques. While industrial software products were meant to service many people and thus tended to have a greater significance in the long run than individual efforts, they were hampered by having to be produced to fulfill contractual obligations. As Schwartz noted, a high-pressure, market-driven environment like this was ill-suited for the creation of “generalized techniques and applications on which sustained and improved practice must necessarily be based.” In cases where particular software companies had produced significant advances in computer science by retaining a large group of talented individuals and leveraging expertise in algorithm design, they were poor agents for scientific advance because there was little incentive to share the knowledge that they had accumulated.

Academic computer science research -- which included computer-based support for scientific applications, university computer support programs for faculty research, and university and government laboratory computer science research proper -- also suffered from shortcomings. Although helping other scientists develop new scientific applications for computers provided the opportunity for computer scientists and mathematicians to develop new techniques, the exigencies of service-type duties meant that there was little time to pursue the interesting questions which arose from this work. A similar problem was found in university settings, where machine time was monopolized by service-type activities or by student programming exercises. Pedagogy and service left little time for faculty to push the envelope in their research. Finally, both environments suffered from a lack of critical mass both in terms of talent and equipment. The result was that academic and government researchers were compelled to undertake projects that might be funded with several hundred thousand dollars within industry, with only a few part-time graduate students and one ore two professional assistants. What was needed was a research environment that blended the best aspects of academia and industry: concentrated resources, intellectual freedom, and no service mandate.

Schwartz argued that a computer science research institute, amply staffed, funded, and equipped, would go a long way toward addressing these deficiencies.[8] Such an

[8] Schwartz envisioned a moderately sized group of 12-24 "first rate people" from the various subfields of computer science with another six "first rate" staff members per researcher to carry out the ideas. In

organization would be charged with developing, evaluating, implementing, and disseminating new tools and techniques in information science.

One of the most important aspects of the proposal addressed the institute's relationship to the outside world. At its core, the center eschewed the development of specific applications. While Schwartz felt it should maintain good contact with industrial and academic programming efforts, such as providing advice "in the design and execution of significant industrial computer projects," the center's main work would be determined by the research interests of the researchers.

Inspired by the Schwartz report and the dialogue with the AMD, Philip Powers, the president of the AUA, authorized the creation of an *ad hoc* committee to discuss the creation of an Institute for Computer Science at Argonne, and the committee held its first meeting at the lab on May 9, 1968.[9] In addition to members of the AMD, participants at the monthly meetings for the Ad Hoc Computer Study Group included well-known computer scientists from the Universities of Michigan, Illinois, Northwestern, Wisconsin, Purdue, Chicago, Carnegie-Mellon, and the Illinois Institute of Technology. From May until October, 1968, the committee worked out the details for a proposed institute which was then presented to the AUA Board of Trustees in December. With the wholehearted endorsement of the Board, a technical steering committee was convened, beginning in January, 1969 to craft a proposal to various federal funding agencies.[10]

---

addition, there would be a large visitor program of approximately twenty people per year, another forty students of the Master's degree level or above, and "first rate equipment." All this for an annual budget of around $12 million.

[9] Wayne R Cowell. Minutes, First Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal Papers of Dr. Wayne Cowell.

[10] Wayne R Cowell. Minutes First Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.

When the *ad hoc* committee began its deliberations, the timing seemed opportune, as there was growing interest at the federal level for the creation of regional computing institutes. The NSF and Department of Education had recently provided funds to the Washington, D.C. based Associated Universities, Inc. (AUI), which operated Brookhaven National Laboratory, to explore the feasibility of establishing such a center in the northeast.[11] The minutes of the meetings clearly show the extent to which the improved funding context and the Schwartz report informed the deliberations of the committee.

As charged by Philip Powers, the committee had to decide several key issues. The first of these addressed whether the proposed regional center would provide computational services or whether it would serve primarily as a think tank for computer science. This tended to be one of the most contentious issues, although the breadth of the participant's opinions were fairly narrow, ranging from providing very limited service to providing none at all.[12] In addition, the committee also had to show how an institute would contribute to the computing efforts at major universities.[13] While most members of the committee eventually agreed that there must be some service component, the consensus seemed to be that, as recommended in the Schwartz report, the center should focus on "problems of great complexity" whose solution "would enhance the status of computer science as a discipline."[14] The committee also decided that the center would not accept direct funding from industry, even though the contract binding Argonne and the AUA entitled them to accept funds from any source.[15] Finally, and significantly, the

[11] Cowell. Minutes, First Meeting.
[12] Cowell. Minutes, Third Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal Papers of Dr. Wayne Cowell.
[13] Cowell. Minutes, First Meeting.
[14] Wayne R Cowell. Minutes, Second Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal Papers of Dr. Wayne Cowell.
[15] Ibid.

committee felt that the Institute should be solely for computer scientists, and that people whose interests "[were] in a physical science rather than in computer science should be in an appropriate division at Argonne rather than in the center."[16]

By July, the *ad hoc* committee had hammered out a draft description for the creation of an Institute for Computer Science to be presented to the Board of Trustees of the AUA.[17] As described in the proposal, the adequately funded and equipped Institute would provide a cooperative setting in which to conduct long-range computer science research that could not or was not being done at universities or within industry. In general, the activities of the institute would focus on four key areas: system characterization and performance; evaluation and certification of numerical routines (mathematical research and numerical software development); structure of operating and communication systems (operating systems and networks); and implementation of comprehensive information systems. An additional area of inquiry included the social implications and applications of computers, whereby researchers would investigate how computers could be used to study issues in transportation, pollution, and city and urban planning.[18]

Also quite explicit in the proposal was that this center would *not* be service oriented. In terms of its interaction with researchers from the physical sciences or humanities, the Institute "could provided a base for groups of researchers in computer-

---

[16] Cowell. Minutes, Third Meeting.
[17] Interestingly, the debate over the name of the institute reveals the continued uncertainty over what computer science actually entailed. Three other names suggested were: "Institute for Applied Computer Science"; "Institute for Computer Science and Engineering"; and "Institute for Computer Research and Application". Some members of the committee also expressed discontent over using the term "applied" because they felt it implied that computer science could be easily categorized into "pure" and "applied."
[18] Ad Hoc Computer Study Group. Draft Description of a National Institute for Computer Science. AUA/Argonne National Laboratory, 1968. Personal Papers of Dr. Wayne Cowell. Although they did make a nod towards studying issues of data safety, for the most part it seems that their notion of the "social implications of computing" were much different from what a historian of technology would consider.

dependent fields to meet and exchange information on a systematic basis" although this "would not be a major effort." Furthermore, while the Institute might provide start-up support for small colleges in the AUA, provisions for large-scale numerical computational services "must be regarded as adjuncts to needed facilities of the Institute and their inclusion must be considered separately and evaluated on economic grounds."[19]

Selling the Institute for Computer Science to the AUA Board of Trustees and to Argonne directors was easy, and indeed both groups threw their support behind the initiative.[20] More difficult was selling the idea to federal funding agencies such as the AEC, the Advanced Projects Research Agency (ARPA), and the NSF.[21] While some funding would come from the AUA and Argonne, it was recognized that the federal government, and especially the NSF, was the only entity that could fund what amounted to a $47 million request over its first four years ($26 million for computer acquisition alone).[22]

Rather than ask the NSF for the entire amount, in February, 1969 President Powers drafted a letter to the agency asking them for $60,000 (later revised to only $19,000) as partial support for a two-week summer study of the issues to be addressed by the Institute.[23] In addition, in March Wallace Givens approached the Mathematics and Computer Science Research Advisory Committee of the AEC with the same proposal. By April, the committee began to get feedback. Givens reported that in discussion with

---

[19] Ibid.
[20] Philip N. Powers, to J.W. Givens. "Charge to Technical Steering Committee, Institute for Computer Science," December 24, 1968. Personal Papers of Dr. Wayne Cowell.
[21] Cowell. Minutes, Fifth Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal Papers of Dr. Wayne Cowell.
[22] Cowell. Draft Budget Summary, Institute for Computer Science. 1969. Personal Papers of Dr. Wayne Cowell.
[23] Wayne R Cowell. Minutes, Sixth Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell. Philip N. Powers, to Milton Rose. February 21, 1969. Personal Papers of Dr. Wayne Cowell.

various government officials it had been made clear that in order to attract federal financing, there needed to be a clear cut proposal demonstrating how computers could be used to address "significant problems representing national goals." One such example that was suggested was the creation of a data bank for high energy physics research.[24]

A second issue the project encountered was that of competition for funding. In addition to the AUI program mentioned previously, representatives of the Big Eight universities plus Colorado State had also begun to develop plans for a regional computing center.[25] More directly, the NSF was concerned that the overall amount requested for the Institute would necessarily require cuts in funding to other academic computer scientists who depended on them for support.[26] Carnegie-Mellon computer scientist Alan Perlis, who was a member of both the AUA Technical Steering Committee and the NSF Computer Science Advisory Board, pointed out that, by necessity, the NSF had to take a national view and was thus disinclined to support the Institute because it smacked of sectionalism.[27] The outcome was probably predictable; in July, the Office of Computing Activities of the NSF informed the AUA committee that they would not support the summer study.[28]

Perhaps a more incisive appraisal of the AUA proposal was provided by members of the newly created Computer Science and Engineering Board (CSEB) within the

---

[24] Cowell. Minutes, Sixth Meeting Technical Steering Committee of the Institute for Computer Science.
[25] Cowell. Minutes, Fifth Meeting Ad Hoc Computer Study Group.
[26] Cowell. Minutes, Ninth Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.
[27] Cowell. Minutes, Eleventh Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.
[28] Cowel to Glenn R. Ingram, Office of Computing Activities NSF. August 8, 1969. Personal Papers of Dr. Wayne Cowell.

National Academy of Sciences.[29] After the failure to obtain funding from the NSF, the Committee turned to the CSEB. The National Academy of Sciences was seen as having some influence in shaping national policy towards science so the creation of a board to assess the implications and needs of computer science research and applications seemed promising.[30] However, Perlis (a member of this panel, too) indicated that the CSEB took a "very negative view" of the creation of institutes, but that it might be helpful to invite members of the Board to visit the AUA Committee to solicit their recommendations.[31]

In October 1969, Launor Carter (chair of the CSEB and from System Development Corporation), and William Miller (member of the CSEB, former director of the AMD at Argonne, and now at Stanford) visited the fourteenth, and final, AUA Committee meeting. After hearing and discussing the proposal, they provided two concrete suggestions and then some observations about the state of computer science in general that worked against such proposals. Most prominently they suggested that for any Institute for Computer Science to succeed, it had to include strong ties to industry, which the current proposal entirely lacked. And second, the AUA Institute, with its emphasis on long-range theoretical research, failed to capture the imagination of the CSEB members, in part because it almost entirely eschewed applications and demonstrated few linkages across disciplines. As for their thoughts on the general state of computer science support in the United States, Carter and Miller noted that, as a discipline, computer science suffered from a total lack of representation in Washington.

---

[29] "Computer Science and Engineering Board Established at Academy of Sciences; Oettinger Named Chair," Communications of the ACM 11, no. 7 (1968).
[30] James P. Titus, "The New NAS Board as a Government Advisor," Communications of the ACM 11, no. 8 (1968).
[31] Cowell. Minutes, Eleventh Meeting Technical Steering Committee of the Institute for Computer Science.

Without allies within funding agencies, ambitious plans such as the Institute were doomed to failure. [32]

The utter collapse of the two year effort to create a regional Institute for Computer Science says much about the state of the discipline entering the 1970s. The proposal had included some of the best computer scientists in America as well as the support of a national laboratory and high-visibility research universities. Yet the proposal floundered, to a large extent because it was unable to demonstrate how theoretical computer science could contribute to the work of other scientists. Although the proposal did contain an initiative to create mathematical software, this was a minor activity. The framers of the proposal were in a Catch-22 situation: if the Institute had included a strong service activity it would have been no different than the various computer facilities already in existence; without it, there was no way to justify its cost.

After almost fourteen months of meetings and planning, the failure to attract even meager funding for the Institute stung considerably.[33] It was in this context that Jim Pool began to make his presence felt. In 1969 Wallace Givens had taken a leave of absence and left Pool in charge of the Consulting and Research section. As deliberations for the proposed institute continued, Pool began to take an interest in a separate initiative within the AMD to produce a library of certified mathematical software subroutines.

Over the years, the AMD had developed a good deal of expertise in the creation of dependable mathematical subroutines for scientific users, in part because of their connection to the work of the British mathematician Jim Wilkinson. Wilkinson, a fellow

[32] Cowell. Minutes, Fourteenth Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.
[33] Cowell to Glenn R. Ingram, Office of Computing Activities NSF. August 8, 1969. Personal Papers of Dr. Wayne Cowell.

of the Royal Society and colleague of computer pioneer Alan Turing, had done seminal work on error analysis of mathematical codes and spent several summers at Argonne in the early 1960s. While working at the National Physical Laboratory in England in the late 1940s, Wilkinson spent considerable time solving linear systems, eigenvalue problems, and matrix calculations on computers. In so doing, he discovered that some of the standard algorithms that were perfectly good mathematically were complete garbage on computers because they were approximations.[34]

The key issue was in the way that computers perform arithmetic; some computers rounded off numbers after a certain decimal position, while other computers simply truncate them after a certain position. In either case, when multiplied over millions of calculations, rounding and truncating computers would yield answers that diverged widely from the real solution. Wilkinson studied what happened mathematically in algorithms that worked and in those that failed. Over time, he developed the idea of a "stable algorithm" where the errors cancelled themselves out. Through a technique called "backward error analysis" Wilkinson was able to prove that the answers computed by particular algorithms were closely related to the solution of the original problem.[35] He and his colleagues then wrote a collection of codes to perform stable matrix computations in ALGOL.[36]

[34] Interview with Dr. Wayne Cowell, at Argonne National Laboratory by author, 11-30-2001. Computers are discrete machines, which means that numbers are represented internally as a combination of 1s and 0s. A computer is only as precise as the number of positions it can store after the decimal place. So pi may only be accurate to fourteen digits after the decimal, after which everything else is dropped.

[35] Office of Energy Research Scientific Computing Staff. Summaries of the fy 1989 Applied Mathematical Sciences Research Program. Washington, D.C.: U.S. Department of Energy, 1989. MCS-HPC "Federal HPC Program", Box 5, Records of the MCS, Argonne National Laboratory.

[36] Matrix calculations are at the root of computational science since almost every problem, from the solution of partial differential equations to the design of a bridge involves matrix calculations in some form.

As a frequent visitor to the AMD, Wilkinson and also Garrett Birkhoff, of Harvard, stimulated intense interest in this kind of work among several mathematicians within the division and over the years the AMD began to acquire a reputation for producing quality mathematical software routines. As more scientists came to rely on computers in their research, access to stable, efficient mathematical software became vitally important.

When it became apparent that the Institute proposal was going to fail, Pool began to push for the AMD to secure funding from the NSF and AEC for a project to develop high-quality numerical software. Although certain numerical software collections were available before this -- most notably those developed by user groups such as SHARE and the IBM Scientific Subroutine Package -- their reputations were "deservedly notorious."[37] Ideally, this new initiative would produce *certified* programs that were reliable, robust, and portable. Certification meant that these routines had been tested on different machines and had proven themselves to operate as intended. Reliability meant that the program would return an accurate result. Robustness meant that the program could handle both intentional and unintentional misuse and still produce accurate answers. Finally, portability meant that the program could be moved to any machine -- regardless of how it did arithmetic -- and return the correct answer.[38]

Although interest in mathematical subroutines for scientific computing extends back to the days of von Neumann and Princeton's program to build a computer at the Institute for Advanced Study, the term "mathematical software" was not coined until

---

[37] William J. Cody, "Observations on the Mathematical Software Effort," in Sources and Development of Mathematical Software, ed. Wayne R Cowell (Englewood Cliffs: Prentice-Hall, Inc., 1984), p. 3.
[38] William J. Cody. Arithmetic Standards: The Long Road. Argonne National Laboratory, 1992. Personal Papers of Dr. Wayne Cowell.

1969 by John Rice, a computer scientist at Purdue.[39] In the early 1960s researchers at the University of Toronto, University of Chicago, Stanford, Bell Laboratories, and Argonne were among the first to critically examine mathematical subroutines. At the time, there was no official outlet in referred journals, so these scientists presented their work in technical reports and at conferences. In 1966, the computer scientists J.F. Traub organized the Special Interest Committee on Numerical Mathematics (SICNUM) which by midyear attracted almost one thousand members. Following the creation of SICNUM, sessions on numerical software became regular features at national computer conferences such as those held by the ACM.[40]

In 1969, John Rice called for a symposium to be held at Purdue to discuss the state of the field. In the announcement, Rice defined mathematical software as "computer programs which implement widely applicable mathematical procedures," although this was later revised to the much broader "set of algorithms in the area of mathematics."[41] William Cody, a member of the AMD and one of the people who would spearhead what would become the enormously influential EISPACK project, pointed out that the two definitions offered by Rice:

> ". . . illustrate the fundamental confusion between algorithms and computer programs that plagued the early development of numerical software. The realization that an implementation is different from the underlying algorithm marks the emergence of mathematical software as a separate field of endeavor."[42]

---

[39] William Aspray, John Von Neumann and the Origins of Modern Computing, ed. I. Bernard Cohen and William Aspray, History of Computing (Cambridge: The MIT Press, 1990), pp. 68-9. Also see chapter 2, "Building Big Iron" p. 47. For the origin of the term "mathematical software" see Cody, "Observations on the Mathematical Software Effort," p. 1.

[40] Cody, "Observations on the Mathematical Software Effort," pp. 2-3.

[41] Ibid., p. 3. J.F. Traub had yet another definition: "numerical mathematics is the theory of the efficient calculation and error appraisal of approximate solutions of continuous mathematical problems." See J. F. Traub. Numerical Mathematics and Computer Science. Pittsburgh: Carnegie-Mellon University, 1972. 5, 14, Computer Oral History Collection, series 2, subseries C, Museum of American History, Smithsonian Institute.

[42] Cody, "Observations on the Mathematical Software Effort," p. 3.

In the past, it was common for a numerical analyst to consider their work completed once an algorithm had been developed. Its implementation was left up to programmers as numerical analysts only touched the keys when it was necessary for their work (and pure mathematicians never did). What was needed, then, were mathematicians who could both design algorithms and oversee their implementation. As characterized by Cody, mathematical software was the bridge "between numerical analysts who devise algorithms and computer users who need efficient, reliable numerical software."[43]

With the support of the AUA and under Pool's leadership, in March 1970, several members of the AMD along with Cleve Moler from Stanford, Y. Ibeke from the University of Texas, Austin, submitted a proposal to the NSF to develop a certified subroutine library. The project was designed to explore new ways to develop high-quality mathematical software *collaboratively* and then *test* this software on a variety of computer architectures. The goal was to be able to produce one set of mathematical software that was truly portable across computer systems that performed arithmetic differently and then distribute and maintain the software as needed.[44]

There were three stages to the program: the selection of routines for testing, field testing these routines at a variety of computer installations, and then distributing these routines to scientific computing centers. Initially, this work would involve researchers at Argonne, Austin, and Stanford because it allowed the codes to be tested on both the IBM

[43] Wayne R Cowell, "Preface," in Sources and Development of Mathematical Software, ed. Wayne R Cowell (Englewood Cliffs: Prentice-Hall, Inc., 1984), p. xi.
[44] Cowell. A Proposal to the National Science Foundation for the Development of a Certified Subroutine Library. Argonne: Argonne National Laboratory, 1970. Personal Papers of Dr. Wayne Cowell.

360 systems (Argonne and Stanford) and a CDC 6600 (Austin), but eventually the software would be sent to dozens of sites.[45]

Rather than create new subroutines (which might be perceived as too theoretical), the researchers choose two important collections of codes for the initial effort. Since the project was initiated at Argonne, one obvious choice was to take James Wilkinson's excellent ALGOL codes, rewrite them in FORTRAN IV, and then proceed with the testing.[46] This satisfied the need for a package of subroutines to solve linear equations which were used widely by scientists and engineers. The other initiative was to develop a package of special function subroutines that were also instrumental to scientific research but had to be highly tuned to individual machines.

At the NSF, this proposal was warmly received and the Foundation agreed to provide $156,000 for the two year program.[47] Over the next several years the National Activity to Test Software Project (NATS) became an unqualified success.[48] As described in the NSF proposal, the initial phase of the project focused on the several interrelated questions including: the choice of basic algorithm; the effect of round-off errors; issues of its correctness and efficiency on different machines; and the design of interfaces for users.[49] After this, the Argonne team of William Cody, Wayne Cowell, and Brian Smith

---

[45] In 1970 the IBM 360 series and the CDC 6600 were the most powerful scientific computers in existence and were widely used in academic, industrial, and government research labs. The NATS project also tested software on UNIVAC machines at the University of Wisconsin.

[46] As already noted, the matrix methods which Wilkinson had addressed were at the heart of computational science.

[47] William W. Bolton Jr., to K.A. Dunbar Manager Chicago Operations Office USAEC. January 14, 1971. Personal Papers of Dr. Wayne Cowell. The NATS project is also noteworthy in that it was able to attract funding from both the NSF and the AEC. In this, the participation of universities was the key ingredient for Argonne researchers to receive NSF funds.

[48] Wayne Cowell also mentioned that NATS stood for "National Science Foundation, Argonne, Texas, and Stanford." Interview with Dr. Wayne Cowell at Argonne National Laboratory 11-30-2001.

[49] Cowell. A Proposal to the National Science Foundation for the Development of a Certified Subroutine Library.

sent the software packages out to different installations to be "field-tested."[50] Based on the feedback received, the subroutines were then "certified" as being robust, reliable, and portable. Finally, these subroutines were made widely available to researchers around the world through Argonne's Code Library.[51]

EISPACK, as the software package became known, was incredibly influential. Along with similar projects from the commercial firm IMSL (International Mathematical and Statistical Libraries, Inc.) and the British venture NAG (Numerical Algorithms Group), for the first time the same library of numerical software was available across a variety of computing platforms. This, in turn, allowed programmers to write and distribute applications to colleagues without worrying whether the proper subroutine library was available. But beyond producing superior software, one of the most significant contributions of the NATS project was to prove that "the development and distribution of quality software [could] be achieved by the joint efforts of several different organizations."[52]

Why did the NATS project attract funding while the Institute for Computer Science could not even get $16,000 for a summer study? Clearly, the overall difference in the scale, cost (by an order of magnitude), and ambitions of the two projects were radically different. But to get at the heart of the issue, it is important to look beyond the obvious reasons.

---

[50] As of April 8, 1971, the NATS tests sites included: IBM 360 – University of Chicago, University of Michigan, Stanford, and Argonne; CDC 6000-7000 – Lawrence Radiation Laboratory, Northwestern, Purdue, and University of Texas, Austin; Honeywell – University of Kansas; UNIVAC 1108—University of Wisconsin; and PDP-10 – Yale University. See NATS Test Sites as of April 8, 1971. Argonne: Argonne National Laboratory, 1971. Personal Papers of Dr. Wayne Cowell.

[51] Cowell. A Proposal to the National Science Foundation for the Development of a Certified Subroutine Library.

[52] Cody, "Observations on the Mathematical Software Effort," pp. 3-4.

First, in putting together this proposal, the AMD at Argonne was drawing on historical expertise in the area of numerical analysis and code development. That the AMD had a critical mass of talented applied mathematicians who were also interested in producing numerical software was not an accident. In early 1970s, substantial cuts were made in the budgets of the national labs. Jim Pool, Acting Director of the Consulting and Research section at the inception of EISPACK, was faced with the unenviable task of having to reduce the staff of the AMD by twenty-five to thirty-five people. After a careful appraisal of the section's activities, Pool decided to preserve the numerical software activities. He even went so far as to honor a commitment made previously to hire the numerical analyst Brian Smith, which then required him to lay off one additional person already on the staff. Smith, it turns out, became one of the chief architects of EISPACK.[53] Pool's decision to make the development of mathematical software a salient feature of the AMD was an astute reading both of the funding environment and of the strong perception that computer science was primarily a support activity to enable other scientists to do their work.

Because of Pool's efforts, when the EISPACK proposal was submitted, the AMD could show that it had the expertise needed to carry out the project. Instead, I suggest that the NATS project was palatable to federal funding agencies because at its heart, the project was cast as *experimental* in nature rather than *theoretical*. The Institute had been pitched as a place to do long-term theoretical investigation in computer science that *might* have applications down the road. In contrast, EISPACK was pitched as an effort to repackage existing algorithms and then do the experiments necessary to make them run on computers with different architectures. Significantly, the "experiments" utilized

[53] Unpublished interview with Jim Pool by Thomas Haigh, July 14, 2004, CalTech in Pasadena, CA., p. 10

newly developed computer networks that connected researchers at universities, industrial labs, and government facilities. Equally important, EISPACK performed experiments as a way to choose algorithms that were the most *efficient* in terms of machine time. As scientists demanded more access to computers, and since the cost of computing remained high, the development of more efficient mathematical algorithms promised to deliver on the bottom line. In short, EISPACK promised more bang for your buck to computer users. At a time when the discipline of computer science was experiencing an identity crisis and it was difficult to justify theoretical research in the field, federal funding agencies felt more comfortable supporting proposals that promised deliverables. The Institute had little to offer along these lines, and thus it sunk.

As the AMD continued to suffer under harsh economic circumstance throughout the early and mid-1970s, Pool, now the Associate Director of the division and in charge of the research section, managed to preserve the mathematical software effort. Perceived as an example of where mathematics and computer science research clearly led to advances in the overall use of computers, every Annual Review from 1972-75 singles out EISPACK and its offspring as being the "world-class" accomplishment of the division.[54] In contrast, the Division's theoretical research in computer science and mathematics was questioned quite directly. For example, after reviewing the activities of two mathematicians in the AMD, the Committee wrote that their work was "academic" and

---

[54] William B. Cannon, to Robert Duffield. "Report of Review Committee Applied Mathematics Division," Jan. 14, 1972. ANL Review Committee for AMD, 1958-1980, B93-00147, MCS Archives; P.J Eberlein. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1975. folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign archives; W. R. Sutherland. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1973. Folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign archives; R.S. Varga. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1974. folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign archives.

its “direct applicability to laboratory problems is much more difficult to foresee. Presumably such work is justified by the need to carry on some very fundamental work, maintenance of general laboratory morale, and the need to have highly competent experts available as consultants.” [55] This comment is typical and captures well the feeling that open-ended theoretical computer science was less desirable and much harder to justify.

The general unwillingness of federal agencies to fund research in theoretical computer science, which is suggested by the failure of the Institute, was also widely experienced by other computer scientists in America. While the roots of the funding difficulties are multifarious, it is possible that the inability of computer scientists to attract long-term funding for research had much to do with the persistent inability of the discipline in the early 1970s to define itself and its agenda. While the Association for Computing Machinery (ACM) Curriculum ’68 had been a landmark document for defining different areas of interest that made up the academic discipline of computer science, practitioners were unsuccessful in leveraging this to achieve recognition in a competitive funding environment.[56] For example, NSF funds obligated to research in computing in fiscal year 1972-73 were $9.9 million, down from $12.5 million the year before. This was projected to decrease again to only $9.5 million in 1973-74.[57] In the midst of a recession some retrenchment could be expected, but as the president of the ACM, Anthony Ralston, noted, funding for the other scientific research programs at the NSF increased by 7.2% during the same period and was slated to go up by another 2.8%

---

[55] Cannon, to Robert Duffield "Report of Review Committee Applied Mathematics Division,"
[56] For a discussion of Curriculum ’68, see Chapter 2.
[57] Anthony Ralston, "Computer Science Research -- Storm Clouds in Washington," Communications of the ACM 16, no. 12 (1973): p. 725.

the following year.[58] Within the Atomic Energy Commission, the situation was similar. In the case of Argonne, funding for the AMD was cut from $1.16 million in 1972 to $0.9 million in 1973, much of it from the research component of the division.[59] Given the growing importance of computers to the research activities of almost every scientific field, it seemed incomprehensible to many computer scientists that they had to continue to beg for funding.

In an article in the Communications of the ACM, Ralston provided several explanations for the disparity in funding between CS and other scientific fields; his comments echo what members of the Computer Science and Engineering Board had told the AUA Institute committee. In particular, he singled out the lack of a presence in Washington, D.C. for the computer science community. While the CSEB had seemed like a step towards filling this vacuum, the National Academy of Science disbanded the board shortly after its creation. This political impotence was likely to continue, Ralston argued, unless computer scientists began to seek positions on major committees and boards that influenced federal science policy. In addition, Ralston called on the American Federation of Information Processing Societies (AFIPS) to establish a permanent presence in Washington to lobby on behalf of computer scientists. An AFIPS office would bring visibility to the discipline and also provide a unified voice when external advice was sought by the government "on matters such as the relative position of computer science in the constellation of scientific disciplines."[60]

---

[58] Ibid.
[59] R.V. Laney, to Edward H. Levi. "Response to the Report of the Applied Mathematics Division Review Committee," March 30, 1973. folder 2, Box 53, Records of the AUA, University of Illinois Urbana-Champaign.
[60] Ralston, "Computer Science Research -- Storm Clouds in Washington," p. 726.

While these were valid points, what Ralston did not mention, or possibly failed to see, was that much of the discipline's troubles lay in poor packaging, the result of their struggle to become an independent scientific field. As I argued in the previous chapter, in an effort to escape their historical role of providing mathematical services to other disciplines, during the 1960s computer scientists and applied mathematicians had gone to great lengths to demonstrate that computer science was just that -- a "science." As such, practitioners were entitled to the same freedoms allowed researchers in fields like physics such as the ability to pursue theoretical research that was independent of any set applications. From 1964-1969, the epistemological battle was reflected in the changing organizational structure of the Division itself. But success came at a price. First, in pushing the theoretical aspects of their discipline, computer scientists unwittingly obscured the places where they *did contribute* directly to getting scientific and engineering work done. Second, and more importantly, they were almost entirely unsuccessful in convincing other scientists that computer science was a discipline on par with fields like physics or chemistry.

The establishment of computer science as a science in its own right depended upon the discipline overcoming these perceptions. Here, it is best to leave the context of the AMD at Argonne to discuss the broader issues of how computer scientists began to overcome these impediments from the early 1970s to the mid 1980s. In particular, I argue that computer scientists began to emphasize the *experimental* rather than *theoretical* facets of their discipline. This approach was beneficial in several respects. First, experiments like EISPACK could be shown to lead directly to applications. Second, experiments were goal directed but still required theoretical input. This allowed computer scientists to pursue theoretical research under the guise of experiment. Third, the identification of experimentation in computer science made it seem more like established disciplines such as physics. Fourth, cutting-edge computational experiments required extremely expensive equipment, which in turn demanded increased funding for the field. Foregrounding the experimental activities also had one other benefit: because experimental computer science could be framed as part of an effort to solve problems in other disciplines, it revitalized the notion of interdisciplinary collaboration.

As I will show, these changes in how the discipline presented itself were crucial to establishing its credibility, and for raising awareness of how computer science could help solve problems. When my story returns to the AMD, the emphasis on experiment and collaboration will be explored through the establishment of the Advanced Computer Research Facility in 1985, which can be seen as a precursor to the High Performance Computing and Communications program. While the changes within the computer science discipline were *significant*, they are not *sufficient* to explain the eventual creation of the HPC in the late 1980s. Thus the eventual success of computer scientists to position themselves as legitimate players in science needs to be seen in a larger context of technological, socio-scientific, and economic change.

The Technological and Social Context for Computational Science

On the technological front, the push to create computer networks to link researchers together provided a perfect opportunity to apply theoretical CS research in an experimental setting. Research in computer networks also coincided with the emergence of true “supercomputers” in the mid 1970s, which required new kinds of software tools to make them useable. By the early 1980s, intense interest in parallel supercomputers, which promised to speed up scientific computations by several orders of magnitude, provided yet another fertile ground for computer science "experiments." Thus, technological trajectories provided the material foundation on which experiments could be conducted.

Socio-scientific changes were also important. By the mid-1980s there were a growing number of researchers from a variety of disciplines who began to identify themselves as "computational scientists." These adherents, who used computer simulations extensively in their work, argued that this computational approach was a distinct "Third Branch" of science and did not fit within the traditional categories of theory or experiment.[61] Computational scientists found common cause with computer scientists in that they both wanted access to the highest performance supercomputers in order to do their work. Significantly, the emphasis on supercomputers for simulations meant that there were opportunities for collaboration between computational and computers scientists, based on the nature of the technology. Computational scientists also brought with them political assets that their computer scientist counterparts lacked -- namely, Noble Laureate spokesmen like Cornell University physicist Kenneth Wilson. These prominent scientists (who were *not* computer scientists) had instant credibility at the highest levels of government.

While the partnership of computer and computational scientists was powerful, it was not enough to generate an initiative of the magnitude of the HPC. Scientists had been performing simulations on computers since their inception; the first problem computed by the ENIAC was a simulation to determine the feasibility of thermonuclear

[61] Pool, "The Third Branch of Science Debuts."

weapons. Yet these simulations, and the existence of high-performance computers, did not in turn create computational science. It would be almost forty years before the computational scientists made claims for methodological distinction.

What finally crystallized computational science, I suggest, was the threat of international economic competition from the Japanese, which succeeded in making support for computer and computational science a national priority. By the mid to late 1980s, computer and computational science had been permanently linked the issues of the United States' national security and future economic competitiveness.

That the discipline of computer science lacked a political presence in Washington, D.C. in the early 1970s points to deeper issues in how the field was perceived by funding agencies. In fact, the NSF, which provided most of the funds for computer science research outside of the mission-oriented agencies (DOD, AEC), did not even recognize CS as a distinct discipline. In response to the stark budget cuts to CS research, in 1974 the Computer Science and Engineering Section (CS&E) of the NSF's Office of Computing Activities (OCA) adopted the following resolution:

> "Resolved that the Computer Science and Engineering Advisory Panel of NSF affirms the distinction of Computer Science from all other science or engineering disciplines and recommends that the National Science Foundation make this manifest in its statistical and programmatic activities."[62]

The emphasis on collecting statistical data was a response to the discovery that in contrast to other scientific fields, government agencies kept no statistical measures for computer science. Instead, the field was lumped together with mathematics or 'mathematical sciences', thereby reinforcing the perception that it served a support function. As a remedy, the CS&E section requested that the OCA conduct a definitive survey of

[62] Bernard A Galler, "Distinction of Computer Science," Communications of the ACM 17, no. 6 (1974).

computer science and engineering, including, among other things, its definition, goals, and role in science. In the words of one member of the committee, the hope was that such a study would “help us clarify to ourselves and to others just who we are and what we aspire to be.”

Further complicating the situation, the lack of computer science representation on advisory bodies such as the National Science Board meant that issues related to the discipline were not even raised in policy debates. Part of the problem, as noted by the former president of the ACM Bernard Galler, was the short history of the field; when the eminent members of the NSB received their scientific education, computer science didn’t even exist as an academic discipline.[63]

While it seems clear that the uncertainty within traditional funding institutions as to the nature of computer science led to the perpetuation of funding structures that worked against computer scientists, external perceptions were surely informed by the debate within the community itself. Indeed, practitioners were having as hard a time defining their activities to themselves as they were at doing it for others. Although Curriculum ’68 had attempted to settle this issue, as late as 1989 the Communications of the ACM, the main journal for computer scientists, still carried articles indicating a continuing identity crisis.[64]

Back in 1967, the renowned scientists Allen Newell, Alan Perlis, and Herbert A. Simon from the Carnegie Institute of Technology attempted to answer other scientists’ queries (“what is computer science?”) in a letter to the editors of Science. In essence,

---

[63] Ibid.

[64] Peter Denning and others, "Computing as a Discipline," Communications of the ACM 32, no. 1 (1989).

their suggestion was that computer science was what computer scientists did.[65] "Wherever there are phenomena," the authors began, "there can be a science to describe and explain those phenomena…. There are computers. Ergo, computer science is the study of computers."[66] Although a good bit of the article is devoted to the argument that computer science was much more than simply a branch of electronics or engineering, I think its greater significance was the authors' efforts to defend their definition by comparing it to particular aspects of other recognized sciences. On the possible objection that computers were constructed and "hence obey no invariable laws, hence cannot be described or explained", the authors point out that such an objection would "rule out of science large portions of organic chemistry (substitute "silicones" for "computers", physics (substitute "superconductivity" for "computers") and even zoology (substitute "hybrid corn" for "computers")."[67] To the charge that computer science lacks a well-defined subject area, since "computer" has no fixed meaning and will change over time, the authors contend that all scientific disciplines experience a change in their phenomena of study. "Astronomy," they wrote, "did not originally include the study of interstellar gasses; physics did not include radioactivity; psychology did not include the study of animal behavior." Finally, the authors take issue with the assertion that computers are instruments, like thermometers, not phenomena, and thus belong to the realm of the sciences that use them:

> "The computer is such a novel and complex instrument that its behavior is subsumed under no other science; its study does not lead away to use sciences, but to further study of computers. Hence, the computer is not just an instrument but a phenomenon as well, requiring description and explanation."

---

[65] This definition is very similar to "physics is what physicists do." See Daniel J. Kevles, The Physicists: The History of a Scientific Community in Modern America (New York: Alfred A. Knopf, 1978).
[66] Allen Newell, Alan J. Perlis, and Herbert A. Simon, "Computer Science," Science 157, no. 3795 (1967): p. 1373.
[67] Ibid.: p. 1374.

Although this missive failed to inspire a rebuttal in the "Letters" column of Science from other scientists -- indeed it seems to be overshadowed by scientists' concern over the implications of the Vietnam War -- Simon's, Newell's, and Perlis' note can be seen as an early, tentative step towards redefining computer science. Their effort to identify particular elements of computer science and then connect them to their counterparts in other scientific disciplines became a strategy that was utilized by computer scientists during the ensuing decade.

Over time, however, the diverse nature of computer science activities worked against a direct one-to-one identification with elements in other disciplines. Possibly in recognition of this difficulty, as well as lessons learned from projects like the Institute for Computer Science and NATS, by the mid-1970s most computer science practitioners no longer pursued this approach in an effort to establish their scientific credentials. In fact, some notable computer scientists even dismissed the notion that their discipline *could* be compared to other sciences. In a 1978 panel discussion on "Computer Science in a Decade" J. Hartmanis, of Cornell's Computer Science Department, argued:

> ". . . computer science is a brand new species among all the known sciences and that it fundamentally differs from the older science . . . in large parts of computer science the classic paradigms from physical sciences or mathematics do not apply and [thus] we have to develop and understand the new paradigms for computer science research. The fundamental difference between, say, physics and computer science is that in physics, we study (to a large extent) a world which exists and the main objective is to explain the existing (and predict new observable) phenomena. Computer science, on the other hand, is primarily interested in what can exist and how to describe and analyze the possible in information processing. It is a science which has to conceptualize, to create the intellectual tools and theories to help us imagine, analyze, and build the feasible…. Computer science is indeed a different intellectual discipline than we have ever encountered before."[68]

[68] J. F. Traub, "*Quo Vadimus*: Computer Science in a Decade," Communications of the ACM 24, no. 6 (1981): p. 353.

Whether this was a commonly held position among computer scientists is difficult to tell. What is clear is that in their quest for recognition as a distinct discipline, practitioners increasingly sought to frame their work as incorporating the two established research traditions of theory *and* experiment. The importance of this reorientation should not be underestimated. By drawing on the cultural/scientific cache of experiment, computer scientists were able to position themselves as legitimate contributors to science, as contenders for funding, for representation on scientific advisory panels, and as equal partners in collaborative research projects.

Rather than address the epistemological issues surrounding experiments, which has been done, I instead want to show how experiments became the salient feature of computer science by the mid 1980s.[69] In truth, the emergence of experiment in computer science is really a matter of its re-emergence. The experimental side of computing has existed since humans began to use mechanical means to do mathematics. Although pioneering computer projects like the ENIAC, and Princeton's AIS computer project were experimental in nature and received considerable attention from scientists and the media alike, over time the construction of hardware was increasingly viewed as engineering and not experiment. While it might seem understandable that people outside of computer science held this opinion, it seems that it was often necessary to remind computer scientists themselves as to the value of experiment in their field. For example, the eminent computer scientists George Forsythe took time in his keynote address to the 1968 International Federation of Information Processing (IFIP) societies Congress to stress the importance of the experimental side, stating:

---

[69] See for example David Gooding, Trevor Pinch, and Simon Schaffer, eds., The Uses of Experiment: Studies in the Natural Sciences (Cambridge: Cambridge University Press, 1989).

> "To a modern mathematician, design seems to be a second-rate intellectual activity. But in the most mathematical of sciences, physics, the role of design is highly appreciated . . . If experimental work can win half the laurels in physics, then good experimental work in computer science must be rated very high indeed."[70]

Two points are implicit in Forsythe's statement: first, good experimental work in computer science was being done, and second, this work was not being recognized, within or outside of the discipline.

Despite his contention that experimental computer science was often overlooked even by practitioners, it wasn't until the late 1970s that the discipline as a whole began to reassess its value. In November 1978, the NSF sponsored a workshop in Washington, D.C. to investigate the status of experimental computer science in the United States. Significantly, the participants included representatives from industry, academia, and government research laboratories.[71] In addition, prior to the meeting, all doctorate-granting computer science departments in the nation were solicited for comments and suggestions on problems related to experimental computer science. The result of these deliberations was the publication of "Rejuvenating Experimental Computer Science: A Report to the National Science Foundation and Others."[72]

Although the Feldman Report, as it became known, seems to have faded into history, in many ways it can be seen as a landmark in the history of CS because it laid out a blueprint for how to "sell" computer science. The first part of the report places experimental computer science at the forefront of technological advance. As the digital

---

[70] Donald E. Knuth, "George Forsythe and the Development of Computer Science," Communications of the ACM 15, no. 8 (1972): p. 722.

[71] The final report was co-authored by Gordon Bell, Digital Equipment Corporation; Bernard A. Galler, University of Michigan; Patricia Goldberg, IBM; John Hamblen, University of Missouri; Elliot Pinson, Bell Telephone Laboratories; Ivan Sutherland, CalTech; and Jerome A. Feldman, University of Rochester.

[72] Jerome A. Feldman and William R. Sutherland, "Rejuvenating Experimental Computer Science: A Report to the National Science Foundation and Others," Communications of the ACM 22, no. 9 (1979).

convergence of microelectronics, communications, and software continued to accelerate, the authors argued that experimentation became vitally important to the intellectual, economic, and military strength of the United States. That experimental CS had already proven its worth could be seen in the university research that had led directly to commercial applications. Database systems for large businesses, for example, required scores of terminals which could access information simultaneously, the ability to update multiple files, and strong file protection mechanisms. In each case, the techniques which made these features possible had their roots in the timeshared operating systems developed at universities like MIT. Likewise, office automation systems which linked hardware and software from areas like graphics, document preparations, database manipulation, timesharing, communications technology, and networking each had their origins in university research and especially from the subfield of Artificial Intelligence.[73]

The decision of the Feldman Report authors to foreground the contributions of experimental computer science was astute because it reinforced the notion that experiments led directly to deliverable products. Despite these past accomplishments however, the Report stated that this entire branch of experimental computer research was in a state of crisis. The growing demand for skilled computer scientists by industry was leading to a drain of quality people away from academia and experiments. In many cases, top people were going to places like Xerox PARC, General Motors, and Bell Laboratories because these companies had state-of-the-art experimental computing facilities. The result was that over two hundred faculty positions in computer science went unfilled in 1979, and this, in turn, jeopardized the future of computing in the U.S.[74]

---

[73] Ibid.: pp. 497-98.
[74] Ibid.: p. 499.

What was needed, according to the Feldman Report, was a complete rejuvenation of experimental computer science within universities. And it is here in their analysis that one can see a clear break from previous efforts to do a one-to-one comparison of research elements across disciplines in favor of comparing the *equipment needs* of experimentalists in computer scientist to the *equipment needs* of experimentalists in other disciplines. This approach is also significant because it incorporates the main arguments of computer science from the 1960s that its practitioners should not be relegated to a service role. The authors argue that computing systems (hardware and software) serve two purposes. On the one hand, scientists and businesses use them as tools to manipulate data. Their second role for computer systems, and one crucial to experimental computer science, was as objects of research experimentation. "Just as the aircraft research community needs experimental aircraft," write the authors, "so the computer science research community needs experimental computers." More importantly, computer systems for experimentalists by necessity must be different from commercial production machines. While the exploration of novel architectures could lead to more innovation, it was also important that the computer not be engaged in production work, because then it would lose its value as an experimental machine.

Moreover, funding agencies needed to realize that the kind of experimental equipment needed for research was prohibitively expensive -- in fact, the Report stated, experimental computer science "almost by definition requires cutting edge facilities." While some industrial research laboratories were able to support experimental research at the level of $40,000-$60,000 per researcher, the vast bulk of experimental work in universities was funded at less than $10,000 per researcher, thereby making these efforts only marginally viable. Given this huge disparity in funding between universities and industry, it was not surprising that many of the best people were leaving academia. If this funding gap could be addressed, the authors said, it would attract not just the best computer scientists to academia, but also "imaginative workers from other disciplines."[75]

In addition to pointing out the needs and benefits of experimental computer science, the Feldman Report also provided recommendations for how to increase funding for it at universities. Because experimentalists were interested in the latest equipment it was necessary to come up with a way to finance it on a continuing basis. In their search for a solution, the Feldman Report suggested new kinds of cooperative ventures between universities, private industry, and the federal government. In the case of universities, the needed change was a reevaluation of traditional benchmarks for professional advancement. Because experimentalists rarely had access to cutting-edge computers needed for their work, they did not produce papers as quickly as theoretical computer scientists and thus were handicapped in the race for tenure. Here again, industry could be very helpful by providing money, equipment, ideas, joint research, and joint appointments to experimentalists. Although corporations usually wanted to exert control over the research if they were paying for it, the Feldman Report suggested instead that companies set up independent foundations which would support work that was more general in nature. Finally, the government needed to reassess its allocation of resources in order to support more experimental CS work and also to retain excellent people in academia who would train the next generation of researchers. In particular, the Report suggested that the computer science should be organized as a distinct discipline by the

[75] Ibid.: p. 500.

NSF and that the funding agency should transfer some of the money earmarked for theoretical research to experimental work.[76]

In their comments on the Feldman Report, the Executive Committee of the ACM applauded the committee on their suggestions and reemphasized that the failure to support experimental computer science in universities would lead to decay within the field and ultimately this would “propagate to other sectors, affecting our economic and military strength in an area where now we enjoy the strongest international position.”[77]

In summary, the Feldman Report was significant for several reasons. In calling for increased funding and disciplinary recognition, the authors subsumed theoretical research under the rubric of experimental science. This had several benefits: experiments could be shown to have direct applications while theory was amorphous; and experiments were expensive and thus required a tremendous infusion of money for the entire discipline. Support experimental computer science and you support the entire field. Furthermore, the Report directly linked computer science to issues of national competitiveness at a time when traditionally strong U.S. industries like automobiles were being soundly thrashed by the Japanese. Finally, the Report was crucial because it called on universities, industry, and the federal government to work together to support cutting edge computer research.

The decision to emphasize experimental computer science began to pay dividends quite quickly. Within a year the president of the ACM, Peter Denning, reported that changes could be seen across the board. The government seemed more willing to shift

[76] Ibid.: pp. 501-02. Interestingly, despite the suggestion of the CS&E board of the NSF in 1972 to recognize CS as a distinct discipline, it still had not happened by 1979.
[77] Daniel D. McCracken, Peter J. Denning, and David H. Brandin, "An ACM Executive Committee Position on the Crisis in Experimental Computer Science," Communications of the ACM 22, no. 9 (1979): p. 503.

resources into computer science research; the NSF had created an Industry/University Cooperative program, an experimental research center program, and had dedicated more money for researchers to use experimental machines. In addition, the NSF and the Advanced Projects Research Agency (ARPA) were cooperating to fund research, while several industries had also made either cash grants to universities or provided them with advanced equipment at deep discounts.[78]

Despite this early success, however, Denning was aware that there was still no consensus as to what constituted "experimental computer science." This uncertainty could be seen in the different ways that the same proposal was evaluated by various funding agencies. Denning dismissed practitioners who argued that computer science was "in flux -- making a transition from theoretical to experimental science -- and hence, no precise definition of 'experimental computer science' is yet available." This position was dangerous, he suggested; agencies that were redirecting money from one science into CS would expect, after a reasonable period of time, that the discipline would produce good experimental work "when judged by *traditional* standards."[79] If practitioners could not succeed against such criteria, then computer scientists would once again lose funding.

Denning contended that experimental computer science had to conform to certain tenets of the experimental tradition; namely it had to classify knowledge based on observation. The key components to this would be having an apparatus to be measured, a hypothesis to be tested, and then systematic analysis of the data. Although there was great flexibility in what constituted an apparatus -- it could be an entire computer system

---

[78] Peter J. Denning, "What Is Experimental Computer Science?" Communications of the ACM 23, no. 10 (1980).
[79] Ibid.: p. 543.

or a much smaller subsystem -- the validity of experimental computer science hinged on what knowledge was derived from the apparatus:

> "No scientific discipline can be productive in the long term if its experimenters merely build components. Building a complex apparatus in the lab is a technological effort that may require great skill. But unless the apparatus is used to obtain significant new knowledge, the research is judged not to be substantive and is soon forgotten. This is why scientists from other disciplines regard machine construction as engineering, not science."[80]

Denning also argues that acceptance of experimental computer science outside of the CS community depended upon addressing three misconceptions held by funding agencies. The first misconception was that it was not worthwhile to repeat experiments in computer science. Indeed, proposals had been rejected out of hand because reviewers contend that this work had already been done. "How untraditional!," he remarked, "It is custom in Physics, Chemistry, Biology, and Medicine for different groups to repeat an important experiment under slightly different conditions or with slightly different methods -- to see if it can be independently corroborated."[81] Computer science is no different, Denning says, and replicating experiments is necessary if the community is to accept their results.

A second misconception Denning sought to dispel was that mathematics was not experiment. He says this misconception appears in common phrases like "theory *versus* practice" and "mathematicians *versus* practitioners." He dismisses this dichotomy, stating that mathematics often provides the *ideas* that are then tested by experiments. Thus, mathematics is crucial to the scientific process because it helps researchers discover which *ideas* are important. Denning's defense of theoretical mathematical research is a crucial part of the overall reorientation of computers science towards an emphasis on experiment. As I argue, computer scientists push experiments because they

[80] Ibid.: pp. 543-44.
[81] Ibid.: p. 544.

promise tangible products to funding agencies, but at the same time by characterizing the relationship between mathematics and experiments *a la* Denning, practitioners are also able to carve out a space for theoretical work.

The third and final misconception Denning attacks is the notion that assembling computer parts "to see what happens" constitutes a science. He strongly condemns this idea, stating that, "unless it seeks to support a hypothesis, tinkering is not science…Undirected work wanders aimlessly, finding interesting results only by accident; it produces 'researchers' with spotty and erratic records." While tinkering has its place, Denning argues that without systematic testing and the perseverance that characterizes other scientific researchers, experimental computer scientists run the risk of being seen as "hackers" and this in turn will be reflected in funding decisions.

Clearly, by 1980 when Denning wrote this article, there had been a definite shift in the way that computer scientists were characterizing themselves and their discipline. This also points to a general maturing of the discipline which was not yet two decades old. At the same time, though, while emphasizing experiments had indeed attracted additional funds, computer scientists increasingly saw their discipline as being in a state of "crisis." This was due to several factors, but the most important one was that there was an acute shortage of highly trained computer scientists at the PhD level. Again, the numbers were telling; in 1979 only 250 PhDs graduated from American Universities (down from 256 in 1975) as compared to 1300 positions seeking PhDs that year. Of these, fewer than one-hundred PhDs sought academic positions while universities had over 650 positions open. An additional symptom of the "crisis" was the doubling of undergraduate enrollments in computer science since 1975 with only slight increases in lab space or faculty size. The implications for United States competitiveness were ominous. "Unless the trend reverses," began one report, "the country will soon lose its lead in computer technology because enough computer experts cannot be trained and because the basic research to ensure a continuing supply of new concepts for the long terms future cannot be conducted."[82]

The perceived crisis in computer science became the central theme of the 1980 meeting of Computer Science Department Chairmen, held biennially in Snowbird, Utah. The forum, which brought together the heads of the 83 departments in the United States and Canada that grant PhDs in computer science, regularly issued a report of their discussions. The 1981 "Snowbird Report: A Discipline in Crisis" highlighted the dearth of qualified individuals and made suggestions for how to address these problems. Although the report reiterates the hybrid nature of computer science -- it is both a theoretical and experimental science and thus is similar to the physical sciences; it is also an indispensable tool in other disciplines and thus is like the mathematical sciences -- not surprisingly, given the recent reorientation of computer science, the solution lay in increasing the funding to computing facilities "capable of sustaining *experimental* research." (my italics)[83] The idea is that for academia to attract talented PhDs away from industry and thus provide the faculty to train future computer scientists, it must be able to offer cutting-edge computing facilities. "Experimental science is expensive," the report states, "the department that wants its research to be at the frontier of Computer Science will require capital investment at a much higher level -- about $55K to $75K per researcher. The department that chooses not to emphasize the experimental side of

---

[82] Peter J. Denning and others, "The Snowbird Report: A Discipline in Crisis," Communications of the ACM 24, no. 6 (1981): p. 370.
[83] Ibid.: p. 371.

Computer Science can get by with a capital investment of about $10K to $15K per researcher."[84]

The Fifth Generation

If computer scientists had been calling for these incredible funding increases in a political vacuum, it is doubtful that they would have achieved much. However, their contention of a crisis in the discipline began to take on political import following the release of the report "Science and Engineering Education for the 1980s and Beyond," which had been requested of the NSF and Education Department by President Carter on February 8, 1980. The conclusions of this report, published by the Office of Science and Technology in the Executive Office of the President, echoed what the computer science community was claiming: that "faculty erosion" due to salary disparities between industry and academia, poor computing equipment at universities, and a lack of PhDs to fill positions threatened the United States lead in computing and engineering professions. While the report's authors hoped that market forces might remedy the situation, they also felt that this would take much too long. Thus, the report called specifically for the direct intervention of the federal government to help alleviate the problems within the computing profession.[85]

That the crisis within the discipline of computer science was receiving Presidential attention suggests a new sensitivity to the role computing played in economic competitiveness. But the attention of President Jimmy Carter, had he not lost his reelection bid to Ronald Reagan, would probably not have been enough to mobilize

[84] Ibid.: pp. 371-72.
[85] Ibid.: p. 373.

support in Congress for increased funding of computer science research. What did mobilize legislators and bring national public attention to issues of computer science was a direct challenge from the Japanese to the United States' dominance in ultra high-end computing. In October 1981 Japan's Ministry of International Trade and Industry (MITI) announced a new $500,000,000 national project to develop, within ten years, revolutionary new computer systems, some of which incorporated artificial intelligence. Dubbed the "Fifth Generation Computer Project," the initiative was designed to develop innovate computer architectures substantially different from traditional von Neumann machines, and then apply these new machines to "cultivate information itself as a new resource comparable to food and energy."[86]

The story, tellingly, was first announced to the business community in the pages of Business Week, whose editors noted the economic implications of the project stating ". . . it is no secret that the Japanese have targeted the computer industry as a critical strategic business for the rest of the century."[87] According to a Japanese researcher involved in the project, ". . . the time [had] come for Japan to change strategy, not just to follow IBM, but to do better."[88] By early 1983, major articles about the Fifth Generation Project appeared in both Newsweek and Time. One year after Time made the personal computer its "Man of the Year," the American public was being told that the future of computing might belong to the Japanese.

---

[86] David H. Brandin, "The Challenge of the Fifth Generation," Communications of the ACM 25, no. 8 (1982): p. 509. In modern computing parlance, first generation computers were those that used vacuum tubes (1940s-1950s); second generation were those that used transistors (late 1950s-1960s), and third generation computers used integrated circuits (1970s- present). Thus, the decision to choose the title "Fifth Generation" is a clear indication of the Japanese desire to leapfrog current computer technologies.
[87] Business Week, July 5 1982.
[88] Quoted in Brandin, "The Challenge of the Fifth Generation," p. 509.

For the American computer science community, the Japanese initiative became a rallying point for several reasons. First, the Fifth Generation Project had all the elements of an experimental project. The goal of the initiative was to develop new kinds of computer architectures (most likely parallel processing machines) as well as new kinds of programming techniques and tools. Hardware and artificial intelligence software would then be applied in an experimental setting to "reason" through huge amounts of knowledge and data. No longer simply a computer, these "knowledge information processing systems" or KIPS would be able "to learn, associate, make inferences, make decisions, and otherwise behave in ways usually considered the exclusive province of human reason." To achieve such a lofty goal would require a tremendous amount of theoretical research to support the experimental machines, but in the end there would be a working system.

American computer scientists applauded the experimental nature of the Fifth Generation Project and even suggested that the program's organization and funding structure was enlightened. The project was arranged as a consortium of eight firms (Fujitsu, Hitachi, Nippon Electric Corporation, Mitsubishi, Oki, Sharp, and Toshiba) and two national laboratories (the government-owned Nippon Telephone and Telegraph's Musashino Laboratories and MITI's own Electrotechnical Laboratory). These participants provided hand-picked researchers who were then relocated to a state-of-the-art facility in Tokyo called the Institute for New Generation Computer Technology (ICOT). ICOT itself was funded entirely by the government through MITI as a way to encourage the industries to provide their top researchers to the project without them also having to assume the risk entailed in such a lofty project. Each week the researchers from ICOT would return to their companies to keep them abreast of developments, and contracts were let to participating firms based on their interests and strengths.[89]

Although not all American researchers thought the Japanese plan was correct (and there was a tremendous debate as to whether they would succeed at all in their efforts to create thinking machines), the coherent vision put forth in the Fifth Generation Project was appealing. At the very least, the public became extremely interested in details of the program. Noted computer scientists Edward Feigenbaum and Pamela McCorduck quickly became best-selling authors with their 1982 book The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World. They make no bones about their position on the issue; the cover of the paperback edition of the book depicts the top half of the Statue of Liberty with a Japanese woman's face and a robotic arm holding up the torch.

In their third-person gripping narrative, the authors discuss the prospects for Artificial Intelligence in book sections titled "The New Wealth of Nations" and "It's Not Just the Second Computer Revolution, It's the Important One." Under their section on "The American Response" subheadings include titles such as "Are There Any More American Heroes?" and "Where There Is No Vision, the People Perish."

Prominent computer scientists are cast in the book as visionary heroes unable to convince either American corporations or governmental entities that the barbarians were at the gate. Playing on the recent decline in the United States economic position worldwide, Feigenbaum and McCorduck wrote:

> "We now regret our complacency in other technologies. Who in the 1960s took seriously the Japanese initiative in small cars? Who in the 1970s took seriously the Japanese

[89] Pamela McCorduck, "Introduction to the Fifth Generation," Communications of the ACM 26, no. 9 (1983): p. 629-30.

> national goal to become number one in consumer electronics in ten years? (Have you seen an American VCR that isn't Japanese on the inside?) In 1972, when the Japanese had yet to produce their first commercial microelectronic chip but announced their national plans in this vital 'made in America' technology, who would have thought that in ten years they would have half of the world's market for the most advanced memory chips? Are we about to blow it again?"[90]

What was needed, according to the authors, was a substantial, well-organized answer to the Japanese initiative. Given the limited resources available to any single body, any true response must be cooperative and include industry, academia, and governmental research organizations.

In summarizing their conclusion, the authors laid out several different courses of action. The first (and unacceptable) one was to maintain the status quo. Their second suggestion was to form an industrial consortium to meet the Japanese challenge, but this might entail rewriting federal laws that prohibited monopolies. A third suggestion was to set up a national laboratory similar to Los Alamos for the development and study of computer technologies. A "National Center for Knowledge Technology" such as this would serve as a clearing house for new innovations as well as being "an expression and institutional embodiment of national will" similar to NASA's Kennedy Spacecraft Center. A final option suggested by Feigenbaum and McCorduck was that the United States "can prepare to become the first great agrarian postindustrial society."[91]

Feigenbaum and McCorduck's book was a slam-dunk. Where previously computer scientists had struggled to be heard on scientific advisory boards, suddenly, they were being called on to testify before Congress about the threat posed by the Fifth Generation. In testimony before the House Committee on Science, Space, and

---

[90] Edward Feigenbaum and Pamela McCorduck, <u>The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World</u> (New York: Addison-Wesley Publishing, Inc., 1983), p. xvii.
[91] Ibid., pp. 265-66.

Technology, Feigenbaum effectively aroused a sense of alarm among legislators. Despite the Reagan administration's lack of concern, Congress pursued several different initiatives to answer the Japanese. In 1982 and 1983, numerous U.S. microelectronics firms created cooperative ventures along the lines of the MITI project. The largest of these ventures, the Microelectronics and Computer Technology Corporation (MCC) had twenty-one members and a budget of around $70 million by 1985. In an effort to encourage such initiatives, in 1984 Congress passed the Joint Research and Development Act which decreased the usual antitrust liability by two thirds. In addition, Congress supported other consortia in an effort to protect the domestic microchip industry and also approved a program to accelerate the development of new microelectronics for military purposes.[92]

There is little question that the Fifth Generation Project served as a lightning rod for criticisms of computer science funding in the United States as well as being a catalyst for change. Here, it seemed, Japan had fully grasped the importance of computer science -- both theory and experiment -- to the future of its economy. As can be seen by the response of Congress, there was no single answer to such a broad challenge. However, it is possible to narrow down the variety of responses to two primary efforts that indelibly shaped computer science research over the following decade.

The first initiative can be seen as a direct answer to the Japanese work in artificial intelligence. In 1983, the Defense Advanced Research Projects Agency (DARPA) launched a ten-year, $1 billion program to develop machine intelligence. Although plans for such a program had been in the making for years, DARPA had a hard time selling it

---

[92] Alex Roland and Philip Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993 (Cambridge: The MIT Press, 2002), pp. 91-2.

to Congress until the Fifth Generation. Robert Cooper, one of the chief architects of the program, admitted "We trundled out the Japanese as the arch-enemies" and used them "unabashedly" in private conversations with legislators. This approach found a receptive audience within the halls of Congress, which formally approved the Strategic Computing Initiative (SCI) in the Defense Appropriations Act of 1984.[93]

As chronicled by historian Alex Roland, the SCI was nothing less than an all-out effort to advance artificial intelligence technologies along an entire front. To the program architects, SCI required concurrent research in every subsystem, with each component feeding into the one above it in a conceptual pyramid. (see pages 283-84 for pictures of the organizational pyramid)

At the bottom of the research pyramid were efforts to develop new computer chips and microelectronics. These would then be incorporated into the middle of the pyramid that included new computers and AI software tools. Near the top were the actual applications that would emerge from the research -- in this case a series of "thinking" machines for the military.[94] The influential guiding light for the entire SC program was the notion that applications would "pull" the technology. It was thought by the directors of the program that if a set of applications could be designated, and their characteristics established in advance, then this would help to shape the kinds of tools and techniques that researchers developed.

A central organizing principle that would help the SCI achieve this goal of "technological pull" was that of "connection." Components had to be connected, so researchers from different parts of the pyramid had to be connected. Connections

[93] Ibid., p. 93.
[94] Ibid. See especially Chapter 2.

extended across researchers within academia, government laboratories, and industry, all of which received funding from DARPA. Most importantly, the components and the researchers had to be connected to the end users.[95] By forging strong connections between researchers, components, and end-users, it was hoped that this would speed up the process by which "good" technologies would be recognized and then incorporated into products.

In seeking to imbue this ethos in the SCI participants, the directors of the program created a research environment that incorporated much of what computer scientists had been calling for over the preceding decade. There was the increased funding, certainly, but beyond that was the recognition that experimental computer science was expensive and required a tremendous commitment of resources over an extended period of time. SCI was also set up to allow different approaches to the same problem; this allowed researchers to look for the underlying ideas that worked best, rather than putting all their apples in one cart. Finally, the DARPA program brought some level of prestige to its researchers who, for a little while anyway, were no longer begging for money.

Over the next ten years, SCI faced many setbacks in its quest for machine intelligence. Although the program did succeed in producing new component technologies, for the most part it failed entirely to create a "thinking" machine. The official end to the program was in 1993, when it vanished as a line item in the DARPA budget; in truth it had dissolved in the late 1980s. According to Alex Roland, however, SC did not really disappear; instead, the project jettisoned its weak parts in favor of its strengths and slowly transformed itself into High Performance Computing (HPC).[96]

---

[95] Ibid., pp. 2-3.
[96] Ibid., p. 285.

While this may be true, and certainly the HPC adopted much of the organization of SCI, it is more likely that it was the second major response to the Fifth Generation project that ultimately was more important.

While DARPA's SCI was focused on machine intelligence and the technologies needed to achieve it, other federal agencies began to focus on ways to increase access to supercomputers among scientists. Whereas Artificial Intelligence made up a very tiny fraction of all computer applications, large-scale scientific calculations were ubiquitous. The need for ever faster computers to handle larger and more sophisticated problems was endemic in computing. The Fifth Generation Project was responsible for stimulating intense interest in machine intelligence, but in actuality this was only one component of a broader movement by the Japanese. Along with the Fifth Generation, the Japanese had also launched the National Superspeed Computer Project to develop its own supercomputer industry.

The importance of supercomputing technology to the professional arch of computer scientists should not be underestimated. Although supercomputers make up a very small percentage of all scientific computing (both equipment and its usage), their centrality to advanced scientific research make them an especially significant technology. While new architectures and components led to faster machines, at least half of the speed improvements over the years were due to better programming techniques and tools. What this meant is that while the primary users of supercomputers were scientists and engineers, computer science research was absolutely critical to making these machines efficient. Thus, the timing of the Japanese supercomputer initiative could not have been more fortuitous for computer scientists; they moved quickly to emphasize that supercomputers were precisely the kind of experimental facilities they needed. New computer architectures required new research -- all of which could be lumped under the rubric of experiment. In addition, the development of high-speed networks (a triumph of experimental computer science) meant that supercomputing resources could be shared by researchers across the United States. The Japanese effort to foster a domestic supercomputer industry transformed the United States supercomputer industry into a national resource. The rising tide of money that poured into supercomputing raised all ships -- including those of computer and computational scientists.

In addition to being a national asset, supercomputers are important for another reason: they are the primary tool of computational science. By the mid 1980s, a growing number of scientists from across the disciplinary spectrum began to refer to themselves as “computational” scientists. According to these researchers, computer simulations had become so sophisticated that they were able to provide new insights into biological and physical phenomena. The use of simulations, they argued, constituted a “third” branch of science that complimented theory and experiment. One of the main spokesmen for computational science was the Nobel Prize winning physicist Kenneth G. Wilson from Cornell University. Wilson won the prize in 1982 for developing a theory of second-order, or continuous, phase transitions in matter under different environmental conditions. Not surprisingly, his research was heavily dependent upon computers as his object of study could not be observed directly, but could only be simulated.

There is some question as to when the term “computational science” was coined. Some claim that it was Wilson, himself, who coined it in 1986 in a paper entitled “Basic

Issues for Computational Science."[97] However, the term "computational" before "science" was not new; in 1966 William Miller (previously head of the AMD at Argonne and now at Stanford) started the Journal of Computational Physics, which is still published today.[98] While the origin of the term is debatable, what is not questioned is that the mid 1980's witnessed a concerted effort by computational scientists to distinguish themselves as a different kind of researcher with different kinds of needs. In a paper published in 1987, Wilson attempted to give a definition of the science and to outline the kinds of problems it faced. Significantly, Wilson made it clear that this was a new science and thus it was experiencing growing pains. In his opinion, computational science dated back to the 1930s and the use of electro-mechanical computers to do science. In contrast, the experimental and theoretical sciences were hundreds, (or thousands) of years older. But, he said, the newness of this methodological approach to science did not detract from its significance.

Wilson felt that the best way to define computational science was by contrasting its core activities with those of experimental and theoretical science. Experimentalists, he argued, ". . . are engaged in measurement and the use (if not the design) of scientific instruments to help make measurements; they are concerned with the design of controlled and reproducible experiments and with the analysis of errors in these experiments." "Theoretical scientists," on the other hand, "are concerned with relationships among experimental quantities and the principles (such as Laws of Nature or symmetry principles) which underlie these relationships, and with the mathematical concepts and

[97] Grand Challenges, High Performance Computing, and Computational Science. Washington: Scientific Computing Staff Office of Energy Research USDOE, 1990. box 5, MCS Archives.
[98] The next oldest entry I can find that include "computational" in its title is the Journal for Computational Chemistry which was first published in 1980.

techniques needed to apply the principles to specific cases."[99] The core interests of computational scientists, though, were different. They centered on ". . . the algorithms which define computational methods for solving scientific problems, the computer software needed to implement these algorithms, the design of computational experiments and the errors in these experiments, the basic laws or models for which the computations are defined, and the mathematical framework underlying the computations."[100]

Wilson acknowledges that basic algorithms are a common feature of the computational requirements of both the theoretical and experimental traditions and computer programs to execute these on standard computers are easily written. But the computational scientists did not use standard computers; instead, they used supercomputers which brought with them certain entailments. For example, to solve partial differential equations or many body computations in classical or quantum problems:

> ". . . poses major problems in algorithms and software that are as intellectually demanding as the problems of modern theoretical or experimental science. Long experience or professional training is required to be successful in computational science at the supercomputer level, making it appropriate to think of computational science as both a separate mode of scientific endeavor and a new discipline."[101]

Lest the two fields be confused, Wilson also made it clear that computational science was different from computer science. The former, he said, is interested in a specific set of applications that apply to a problem within a particular discipline, such as physics. Thus, the computational scientist is trained primarily in that field. In contrast, computer science is interested in the "generic intellectual challenges" of the computer itself.

---

[99] Kenneth G. Wilson, "Grand Challenges to Computational Science," (Cornell Center for Theory and Simulation in Science and Engineering: 1987), p. 2.
[100] Ibid., pp. 2-3.
[101] Ibid., p. 3.

As the field of computational science matured in the late 1980s, the effort to distinguish between it and computer science intensified. In 1989 a Grand Challenges to Computational Science conference co-sponsored by the Air Force Office of Scientific Research, NASA, the University of California, and the Department of Energy drew over one hundred senior scientists from universities, national laboratories, and industrial research centers. In seeking to describe their field, few participants viewed themselves as computer scientists; instead, they were specialists in various scientific and engineering disciplines who used supercomputers. Along these lines, several papers from the conference focused on placing computational and computer science activities on a continuum:

Physical system ↔ math model ↔ algorithm ↔ numerical implementation

The claim was made that the term "computational science" is more strongly related to "physical system" and "math model" and that the term "computer science" fell closer to "algorithm" and "numerical implementation."[102]

Nevertheless, despite the difference in training and emphasis, computer and computational scientists share a common demand for incredibly expensive equipment (supercomputers), networks, and human resources. Indeed, without supercomputers, there would be no computational science as Wilson envisioned it and computer scientists claimed their discipline would wither.

---

[102] Eugene Levin, "Grand Challenges to Computational Science," Communications of the ACM 32, no. 12 (1989): pp. 12-13.

Wilson also devoted a section of his paper to a defense of the supercomputer as a scientific instrument, arguing that regardless of the incredible expense, funding must be made available in order to advance the frontiers of computer technology. Whereas traditional experimental equipment such as microscopes and telescopes obtained images directly from nature, or more generally examined nature and man's additions to it, Wilson argued that the principle function of supercomputers was to overcome these limitations. Supercomputers could be used to see *into the future* -- for example in weather forecasting -- which no other instrument could do. They could also be used to reconstruct past events long before experimental observations were in place, such as in investigations of the Big Bang. Moreover, supercomputers could be used to explore objects on smaller scales and at shorter time intervals than those used in traditional experimental equipment. Supercomputers were also indispensable for evaluating models or explanations of experiments, especially if the model was too complex to be solved analytically.[103]

Wilson made this coherent statement about computational science in 1987, but this was more an effort in pulling together ideas and statements that had already been in use for years. What is important about the emergence of computational science as a distinct discipline is that its spokesmen brought with them a certain cultural and scientific cachet. Computer science had produced no Nobel Laureates; computational science had. Thus, it is here, at the leading edge of computer technology, that computer scientists and research scientists finally found common cause.

The Japanese challenge gave new impetus to their calls for increased funding of supercomputer projects, and it became clear to federal funding agencies that actively

[103] Wilson, "Grand Challenges to Computational Science," pp. 5-6.

supporting supercomputing might kill three birds with one stone; it would help protect the domestic supercomputing industry in the face of the Japanese challenge; it would provide increased access to these machines for computational scientists; and it would provide the experimental facilities needed to attract the best computer scientists to academic work and to train the next generation of computer scientists.

As historian Donald McKenzie has pointed out, the terms “supercomputer” and “high-performance computing” are relative.[104] Long before the word was coined (sometime in the late 1960s or early 1970s) the fastest computer in any given period would be considered the “supercomputer” of its day. As computer technologies have advanced, the level of performance needed to be considered a “supercomputer” has also changed, even though the criterion has not. Since the late 1950s, the benchmark for performance has been the speed at which a computer performs “floating-point” arithmetic -- the computer equivalent of scientific notation. The number of floating-point operations (flops) per second has increased enormously “from the thousand (kiloflops) in the 1950s to the millions (megaflops) in the 1960s to thousand millions (gigaflops) in the 1980s to the million million (teraflops) machines” of today.[105]

While “supercomputers” existed as custom made machines, such as IBM’s 7030, also known as STRETCH, at Los Alamos, the first machine to achieve commercial and technical success was the Control Data Corporation 6600 released in 1964 and designed by Seymour Cray.[106] The CDC 6600 quickly won business away from IBM, especially

[104] Donald MacKenzie, "Nuclear Weapons Laboratories and the Development of Supercomputing," in Knowing Machines: Essays on Technical Change, ed. Weibe E. Bijker, W. Bernard Carlson, and Trevor Pinch (Cambridge: The MIT Press, 1998).
[105] Donald MacKenzie, "Nuclear Weapons Laboratories and the Development of Supercomputing,", p. 100.
[106] The STRETCH machine was delivered to Los Alamos in 1961 and was priced at $13.5 million. However, it consistently failed to meet its targeted performance specifications which led IBM to reduce the

for scientific computing. The National Labs, particularly, were keen on CDC equipment; a 1971 survey of computing equipment installed in the Labs by the Atomic Energy Commission showed the following: Argonne (CDC 3600); Brookhaven (two CDC 6600); Los Alamos (CDC 6600, CDC 7600); Lawrence Laboratory at Berkeley and Livermore (two CDC 6600, CDC 7600); and Sandia (CDC 3600).[107]

As more and more scientists began to use computers in their work, there was a never ending demand for more computational power. Complex modeling problems unfeasible on a standard commercial machine because it might take a week of computing time could be done in hours on a supercomputer. For example, if the first CDC machine for scientific computing from the late 1950s, the CDC-1604 is given a power factor of 1, then the CDC 6600 is twenty-five times more powerful.[108]

In 1972, genius computer designer Seymour Cray left Control Data to form his own company, Cray Research, in Chippewa Falls, WI. In 1976, the company installed the first Cray-1 supercomputer at Los Alamos (the second went to the NSA) and this machine quickly became the worldwide *de facto* standard for supercomputers. At $8.8 million, the Cray-1 was capable of 160 megaflops (160 million floating point operations per second) and boasted eight megabytes of main memory.[109] By way of comparison, if the CDC 6600 has a power rating of 25, the Cray-1 is a 500. As new generations of supercomputers were introduced, the "Cray hour" became the benchmark for

---

final price to $7.78 and cancel all pending orders for the machine. Despite the lowered performance, the STRETCH was still the fastest computer in the world from 1961-1964.

[107] Jr. Morse, E.H., to Glenn T. Seaborg Chairman Atomic Energy Commission. "Multiple Procurement of Computers on Deferred Payment Plans," February 9, 1971. folder 1, 7847, RG 326 Records of the AEC, collection 9, NARA II.

[108] J.T. Pinkston, "Supercomputer Trends and Needs," in Frontiers of Supercomputing, ed. Nicolas Metropolis et al. (Berkeley: University of California Press, 1986), p. 130.

[109] http://www.cray.com/about_cray/history.html. For comparison, my laptop has a forty gigabyte main memory, which is approximately 500 times larger than the Cray-1 memory.

comparisons between different machines.[110] But as scientists were quick to point out, this was still not enough computing power; to increase the resolution by an order of magnitude of a three-dimensional, time-dependent problem required an increase of computing speed by a factor of 10^4 over a Cray 1.[111]

Throughout the 1960s and 1970s, the United States was the undisputed leader in high-performance computing. Foreign access to this technology was tightly controlled by the federal government and according to one Control Data executive, it was used "as the carrot or the stick in the U.S. government's effort to reward or punish other governments in the realm of foreign policy."[112] Supercomputers also became the primary computational workhorses in key American industries like automobiles, petroleum, and pharmaceuticals.[113] Thus, the Japanese supercomputer project of the early 1980s was seen by many, including policymakers, as a direct challenge to our technological lead, economic competitiveness, and our national security.

All of these elements were manifest at the 1983 "Frontiers of Supercomputing" conference co-sponsored by Los Alamos National Laboratory and the National Security Agency and attended by 165 representatives of academia, industry, and government. Tellingly, the keynote address was by Admiral B.R. Inman of the Office of Naval Research; the second presentation was by New Mexico Senator Jeff Bingaman. Both

---

110 In proposals seeking supercomputing time, estimates for time needed were made in CRUs where one CRU equals one Cray-1 hour. This unit of measure persisted long after the Cray-1 was obsolete. For example, in 1991 the aggregate computing power available to Argonne scientists was approximately 100,000 CRUs per year. Hans G. Kaper and Rick L. Stevens. Argonne National Laboratory High-Performance Computing and Communications Program. Argonne: Argonne National Laboratory, 1991. HPCC White Paper, box 1, MCS Archives, Argonne National Laboratory.
111 Nicolas Metropolis and others, eds., Frontiers of Supercomputing (Berkeley: University of California Press, 1986), p. xi.
112 Quoted in MacKenzie, "Nuclear Weapons Laboratories and the Development of Supercomputing," p. 101.
113 Metropolis and others, eds., Frontiers of Supercomputing, p. x.

linked supercomputing directly to national security and long-term economic competitiveness.[114] Although not explicit in Bingaman's speech, his comments make it clear that the Japanese challenge was going to have far reaching implications on science in the United States:

> "I am sure it is clear to all of us that the challenge to our predominance in computer science and technology is coming from Japan. The Ministry of International Trade and Industry (MITI) National Superspeed Computer Project and MITI's Fifth Generation Computer Project are really the reason we are gathered here today. The remarkable success that the Japanese have had in the semiconductor industry in the last decade lends credence to their ability to achieve the ambitious goals that they have set for themselves in these projects as well. Without the challenge of Japan, I doubt whether we would be considering changes in the decentralized and possibly uncoordinated way in which we have developed our information-processing industry thus far."[115]

Bingaman's speech is significant and it speaks directly to my argument about the contingency involved in the sudden emergence of computational science at this moment in history. Scientists had been performing simulations on computers since the ENIAC became available of use in 1946; the first problem run on the machine were calculations to determine the feasibility of making a thermonuclear weapon. But the existence of "supercomputers" at any given time since then did not, in turn, create the computational scientist. One possible explanation for this is that the technology was too rudimentary and that computational scientists had to wait until visualization technology (which is used to convert the incredible amounts of data generated by computer experiments into a visual image) had matured. While visualization technology is important to computational science, I argue this is not the only kind of simulation. For example, one might use a spread sheet to calculate an answer based on data entered into a given range of cells. By

---

[114] Jeff Bingaman, "Supercomputer: A Congressional Perspective," in Frontiers of Supercomputing, ed. Nicolas Metropolis et al. (Berkeley: University of California Press, 1986), B.R. Inman, "Supercomputer Leadership: A U.S. Priority," in Frontiers of Supercomputing, ed. Nicolas Metropolis et al. (Berkeley: University of California Press, 1986).
[115] Bingaman, "Supercomputer: A Congressional Perspective," p.1.

changing the numbers in one of these cells, it is possible to see how the answer also changes. Asking "what if" questions by altering the input data is analogous to performing a numerical simulation of a particular problem on a large computer. Numerical simulations were a standard use of computers throughout their history. Yet, as I have mentioned, this did not produce the computational scientist.

The Bingaman quote above explicitly recognizes that without the Japanese threat to the United States' leadership in a technology that heretofore had received little interest from Congress, there would not be nearly this amount of interest in either computational science or computer science. But it was computational scientists, especially, who were able to take advantage of the links drawn between their respective fields and the international position of the United States in order to achieve disciplinary recognition.

For American scientists interested in the computational approach, the paramount issues of supercomputing were access and cost. In 1982, there were about sixty-one supercomputers in the world: forty-two in the U.S. (and only three of these in universities); seven in England; six in Germany; four in France; and two in Japan.[116] The small number of computers meant that access was often times very difficult. But an equally significant impediment to their use was the price; even if a researcher could gain access to a supercomputer at a national lab, their academic budget bought little time on it, as fees for their use typically ran $2,000 or more per hour.[117] Since supercomputing cycles were seen as the "raw material for computational science" devising some way to increase researcher access to them while also lowering the cost of computing became central to the arguments of computer and computational scientists.

[116] Gene Dallaire, "American Universities Need Greater Access to Supercomputers," Communications of the ACM 27, no. 4 (1984): p. 292.
[117] Ibid.: p. 293.

That computer and computational scientists found common cause in supercomputing can be seen to a certain extent in the increasing number of articles published after 1983 in the Communications of the ACM that address researchers' access to these machines from the perspective of scientists in other disciplines. One such article documents the increasingly frequent practice of American scientists traveling to foreign countries -- especially Germany -- in order to get time on *American-made* supercomputers.[118] Although the United States had the most number of installed supercomputers, the problem was that they were largely concentrated in government laboratories and used for very specific purposes such as weapons or energy research. Should this situation be allowed to persist, so the argument went, it would have wide-ranging repercussions on the ability of American scientists to do advanced research.

The second area where computer and computational scientists found common cause was in the arguments they put forth for why access was necessary. First was the concern that American basic research and engineering would lag behind that of other countries which did encourage their researchers to use supercomputers. Here, it was common to cite examples of where U.S. researchers had been forced to curtail their investigations for lack of access to a high-performance machine.[119] A second common theme was that supercomputers and the simulations made possible by them were crucial to the future of American industries. Simulations allowed companies to eliminate many of the costly steps, such as building actual prototypes, which were required to bring a new product to market. Rather than actually build a concept car or airplane, it could be modeled on a computer which was both faster and cheaper. The problem, according to

[118] Ibid.: p. 292.
[119] Ibid.: p. 294.

computer and computational scientists, was that the industries were unprepared for this revolution and would not be ready to embrace it until there was a large pool of scientists trained in supercomputing. A third theme that was heavily emphasized was that providing access to supercomputers in universities would lead to more people trained in their use and subsequently a higher demand from American computer manufacturers for these machines. Creating such synergies would thus have the benefit of protecting the domestic supercomputer industry while at the same time stimulating innovation. The fourth, and final argument that computational and computer scientists held in common was that it was up to the federal government *and* industry to achieve some solution to the problem.[120]

As with DARPA's Strategic Computing Initiative, there had been some plans within NSF to address supercomputing in the early 1980s, but the Japanese challenge hastened their crystallization. In 1983, the NSF released the Bardon and Curtis Report which led to the creation of an Advanced Scientific Computing program and later the establishment of five National Computing Centers.[121] That same year, the agency created the Advisory Committee for Advanced Scientific Computing Resources, which consisted of fifteen members representing universities, industry and national laboratories. A summary of the committee's first meeting in January, 1984 reveals that many of the arguments that had been made by computer scientists during the preceding decade and computational scientists more recently were being absorbed institutionally. This is significant for two reasons: first, the opinions were not simply those of one NSF

[120] Ibid.: pp. 293-98.
[121] M Bardon and K Curtis. A National Computing Environment for Academic Research. Washington: NSF Working Group on Computers for Research, 1983.

Advisory Committee, but reflected a much broader consensus; and second, the NSF was establishing the *framework* in which support for supercomputing would occur.

As their report makes clear, by 1984 it was now accepted both within computing circles and at federal funding agencies that computational science was indeed a third methodology for doing science. Next, the Committee recognized that computational science was a scattered activity and that ultra-fast networks would be the key to providing this dispersed group of researchers with access to high-performance computing resources. Finally, and most significantly for this story, the Advisory Committee explicitly noted that "collaboration between natural scientists and computer scientists should be encouraged to overcome the human limitations for comprehending the complexities of computer programming."[122]

The Advisory Committee's encouragement of interdisciplinarity, was, I think, in recognition that supercomputing -- the technology underlying computational science -- was inherently a social activity. If we look back at Wilson's description of the "core interests" that comprise computational science, it is clear that this was a team-based approach to science. Algorithm development and their implementation into software were the provenance of researchers like those involved in EISPACK and the development of numerical software. The design of computational experiments required the combined talents of programmers and the scientist from the particular field of investigation (physics, chemistry, biology, etc.). In addition, the identification of errors in the experiments and the selection and analysis of particular mathematical models underlying the computations also required different sets of skills. Thus, if supercomputing as a social activity informed the character of computational science, from the very beginning

[122] Rosalie Steier, "NSF Takes the Initiative," Communications of the ACM 27, no. 6 (1984): p. 528.

this technology also suggested to funding agencies like the NSF a particular way to organize their support. As we will see, this argument becomes instantiated in the HPC.

At the same time that the NSF released the Bardon and Curtis Report, two other governmental bodies came to similar conclusions following their own investigations. In 1982, the Panel on Large Scale Computing in Science and Engineering (known as the Lax Panel after Peter Lax from the NYU's Courant Institute) called for a federally coordinated program in supercomputing as a way to increase access but not facilitate redundant efforts.[123] The following year, a panel from the Federal Coordinating Council for Science, Engineering, and Technology (FCCSET) recommended that in the absence of a federally coordinated program, individual agencies should create their own supercomputer centers and connect them through networks to handle their own research needs.[124] The recommendation by the FCCSET for federal agencies to "go it alone" was recognition that a centrally coordinated supercomputer program would require the imposition of guidelines on relatively independent federal agencies, each having their own management styles, bureaucratic organizations, and research agendas.

[123] Peter D. Lax. Report of the Panel on Large Scale Computing in Science and Engineering. Washington: National Science Foundation NSF 82-13, 1982. Throughout the history of computing, the fluid movement of people between industry, academia, government labs, and federal administrative positions has been crucial to its development. For example, Jim Poole, the director of the Mathematics and Computer Science section of the AMD at Argonne, moved to the Department of Energy in 1979 to work in the Math and Computer Science section of the Office of Basic Energy Science. While there, he was instrumental in helping to draft the Lax Report.

[124] Report to the Federal Coordinating Council for Science, Engineering, and Technology Supercomputer Panel on Recommended Government Actions to Retain U.S. Leadership in Supercomputers. Washington: U.S. Department of Energy Report, 1983.

High Performance Computing at Argonne: Computational Science Gains Momentum

Even without some kind of central coordination, by the middle of 1984, money was beginning to flow into supercomputer projects. The NSA, NSF, and Department of Energy (DoE)[125] all received grants to set up supercomputer centers or to buy time on existing machines, and total federal funding for advanced computing expanded from $173.4 million in 1983 to $226.9 million in 1984. Although the largest amount of this money was for defense related projects, the NSF and DoE became strong supporters of basic research in supercomputing.[126]

The improved funding environment had profound effects on both computer and computational scientists. By 1984, it was now widely accepted that computer scientists could contribute directly to the work of other scientists. Established also was the need for computer scientists and applied mathematicians to have access to the latest computer equipment in order to develop the tools and techniques that would enable computational scientists to do research. Institutional changes quickly followed; in 1984, the Department of Energy reorganized its disparate computing activities under one roof. The Applied Mathematical Sciences research activity, Advanced Computing activity, and the Energy Sciences network were now managed jointly by the Director, Scientific Computing Staff within the Office of the Director of Energy Research.[127]

---

[125] The Department of Energy was the successor agency of the Energy Research and Development Agency (ERDA) which itself was the successor of the Atomic Energy Commission. In 1974, the Energy Reorganization Act divided the responsibilities of the Atomic Energy Commission into ERDA and the Nuclear Regulatory Commission (NRC). ERDA handled research and development; the NRC dealt with regulation. Then in 1977, the Carter Administration signed the Department of Energy Organization Act which transformed ERDA into the modern-day DoE.

[126] Roland and Shiman, <u>Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993</u>, p. 289.

[127] <u>Summaries of the FY 1989 Applied Mathematical Sciences Research Program</u>. Washington: Department of Energy, 1989. DOE/ER-0422 UC-32. Federal HPC Program, box 5, MCS Archives.

Back at the Applied Mathematics Division at Argonne, the rapid ascension of computer science provided new opportunities for researchers there. In addition to increased autonomy for the Mathematics and Computer Science section within the laboratory, its members also began to reassert themselves as valuable collaborators in interdisciplinary projects. A brief examination of the activities and struggles of the AMD during the preceding decade makes it clear that there was a tremendous reevaluation of the *value* of mathematics and computer science research to the overall mission of the Laboratory. What is also clear is the extent to which efforts to couch the Division's mathematics and computer science research as *experimental* in nature placed these researchers in an excellent position to take advantage of the new funding opportunities in the mid 1980s.

Going back to the mid-1970s, the research component of the division, under the leadership of Jim Poole, had continued to emphasize its strengths in mathematics, numerical algorithms, and the creation of mathematical software such as EISPACK.[128] At the same time, in an effort to remain relevant in more areas of computer science research, at the urging of its Review Committee, members of the research and consulting section also began to do investigations using networks.[129] Although most of the networking activity was carried out through the Central Computing Facility (the service side of the AMD) for computer scientists, networks were appealing because they lent themselves to experiments. For example, the mathematicians working on EISPACK and

[128] Sutherland. Report of the Review Committee for Applied Mathematics Division.
[129] Robert G. Sachs, to Paul W. McDaniel. "Response to the Review Committee Report for the Applied Mathematics Division," February 20, 1974. folder 2, box 53, Records of the AUA, University of Illinois Urbana-Champaign.

its successor packages were able to attract additional funding by using networks both to test and distribute their software.[130]

While research that produced tangible products -- such as mathematical software and networking -- received strong support from the Review Committees into the mid 1970s, they were less certain how the other activities of the section contributed to the work of other Argonne scientists. Thus, in 1976 the Committee requested that the Laboratory Director begin a study on the impact of computation and applied mathematics on ANL. As one reviewer noted, such a study would "provide an additional basis for any Laboratory decisions on the amount of its resources put into computing and mathematical operations and development, and serve as useful guidance to [the] research program." [131]

That the Mathematics and Computer Science research program was on thin ice was apparent from the text and tenor of a report on the AMD's long range plans released at the beginning of 1975 and is worth quoting at length:

> "The research activities have emerged from a chaotic period during which the fight against inflation was competing with the fight against the energy crisis against a background of uncertain, and sometimes decreasing, funding. Those parts of the Division whose strength appeals to the sponsor have been identified and used as the basis for building our new program, while other areas have been dropped, some because they were weak, but other because they did not appeal to the sponsor. One of our major planning activities, in fact, has been to work with our sponsor in assisting in the development for a rationale for the whole AEC Mathematics and Computer Science research program. This program, unlike the Nuclear Physics or Chemistry programs, for example, did not grow out of a belief on the part of the AEC that basic research in mathematics was needed to underpin the R&D work in reactor development in the same way that basic research in these other areas was needed. It arose instead out of a fairly straightforward effort to construct computers at a time when the future path of computer development was still somewhat obscure and it was unclear whether the kinds of computer facilities which the AEC required would be available commercially or whether the AEC would have to construct them all for itself. It has since become clear that the majority of types of computer facilities which would be required can in fact be obtained commercially. The Mathematical and Computer Science research program grew out of this essentially developmental activity and has been under greater pressure to establish its programs on a

[130] G. Estrin. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1976. folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign archives.
[131] Ibid.

> zero-budgeting basis each year and to demonstrate its direct relevance to the AEC's programmatic work than have the other basic research programs. At the same time it has also been under pressure not to carry out work which is so directly of benefit to other programs that it can be characterized as subsidizing them, or carrying out their work without any element of accountability to the program because of the absence of funding support from that program."[132]

Reflecting the larger effort in the computer science community to couch basic research in terms of experiment, this report links the core research activities of the division -- applied analysis, numerical algorithms, and mathematical software -- to current experiments in the development of networks:

> "The Research Facilities Program currently addresses two problems: the remote usage of computational resources via high capacity networks and the formulation of the plans for a mathematical and computer science laboratory. The former will focus on the use of remote resources for the evaluation of numerical software, the use of networks for distributing software, and the remote usage of specialized applications systems. Thus, resource sharing will be investigated in the context of the previously discussed research ventures. The development of a laboratory for mathematical and computer science research is only in its early stages. Emphasis continually will be placed upon those areas impacting the needs of the previously described programs, that is, how do you facilitate the solving of problems by mathematicians and computer scientists through the utilization of state-of-the-art hardware and software systems. *Basic issues of computer science will be addressed only as they block the achievement of problem-solving goals*."[133] (my italics)

In 1977, as part of the transition of the Energy Research and Development Agency into the Department of Energy, the Applied Mathematics Division was again asked to explain its role within the overall activities of Argonne. The report is interesting in that it not only repeats the main arguments of the Research and Consulting section from the 1960s -- that computer scientists and mathematicians had to be freed from doing service work for other divisions -- but it also emphasizes that members of the Mathematics and Computer Science section were heavily engaged in experimental work leading to the creation of mathematical software.

---

[132] R.J. Royston. A Synopsis of Long Range Plans. Argonne: Argonne National Laboratory, 1975. folder 2, box 53, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.
[133] Ibid.

The reiteration of the mantra ‘free from service work’ was most likely an acknowledgement that the Mathematical and Computer Science Research section included “only a fraction of the mathematical and computer science research carried out at Argonne.”[134] For example, researchers in Applied Physics were doing work on Boltzman Transport Equations; research in fluid dynamics was being done in the Components Technology Division; and the Chemistry Division was doing work on automating chemistry experiments. According to this report, what made the AMD effort unique, however, was that the mathematicians and computer scientists were not wedded to doing this kind of work. This meant that they spent their time conducting *basic* research in software engineering methodologies (scientific software packages), computational mathematics, algorithm development and implementation, and some long-term theoretical investigations[135]

A comparison of the Mathematical and Computer Science research section budget for 1976 and 1977 shows the gradual reorientation of projects to emphasize experimental work. In 1976 the section’s budget was $1,025,000.[136] Of this, only 12.2% ($125K) went to work that could be considered primarily theoretical. Numerical algorithms, which included their development and evaluation and was a mix of theory and experiment, accounted for 32.2% ($330K); Mathematical Software, including systemization studies and automated programming aids, was primarily experimental, and received 33.2% ($340K); Experimental systems, which included networking, consumed 22.4% ($230K) of the budget. In 1977, the overall budget increased to $1,069,000.

[134] R.J. Royston, to Members of the Applied Mathematics Division Review Committee. "The Role of the Applied Mathematics Division at Argonne National Laboratory," October 19, 1977. folder 2, box 53, Records of the AUA, University of Illinois Urbana-Champaign Archives.
[135] Ibid.
[136] In comparison, the Central Computing Facility budget for 1976 was $3,570,000.

While funding for theoretical work increased to 13.6% ($145K), a considerable amount of money was transferred from the numerical algorithms effort which was mixed theory/experiment, to the mathematical software effort. Support for Numerical algorithms fell to only 23.1% ($147K) while Mathematical software work now accounted for 39.7% ($425K). Likewise, the Experimental Systems unit saw an increase to 23.6% ($152K). Of that, $133K was devoted to research on how to use the Lab's ARPANET connection to execute experiments in remote resource sharing.[137]

It is important to remember that this shift in funding to experimental categories did not mean that less theoretical work was going on; it means only that such work was being subsumed under experimental projects. For example, several researchers within the section were working on a project called TAMPR which was an experimental piece of software designed to automatically transform a program written for one machine so that it could be run on one that performed arithmetic differently. In their 1977 review, the Committee was critical of TAMPR, noting that while it was a "good example in the area of software portability technique . . . we feel that more effort should be devoted to research on tools and techniques that can aid in the design and production of programs."[138] In defense of TAMPR -- and of the theoretical research underpinning it -- the AMD's Director wrote:

> "We remark that the work on TAMPR addresses the question of the reliable transformation of programs and thus bears on a much broader class of problems than just portability, including (a) language extensions for abstract formulation of algorithms. (b) automated generation of several programs from a single prototype, and (c) improvement and optimization of programs produced either by hand or by machine. The far-reaching implications of reliable transformation techniques, strengthened by the fact that, even in

---

[137] Wayne R Cowell, to Richard Royston. "Information for Review Committee," October 17, 1977. folder 2, box 53, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.
[138] E.N. Pinson. Report of the Review Committee for the Applied Mathematics Division. Argonne: Argonne National Laboratory, 1977. folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

> its current interim state, TAMPR is a useful program development tool, have persuaded us that further development of TAMPR will be a major emphasis in software engineering research."[139]

The Review Committee seemed willing to accept this, even though the following year they still felt that TAMPR "was taking on a more abstract tone" and thus questioned its purpose.[140]

It should also be remembered that using the category of "experiment" to cover computer science research could be a two-edged sword. While this approach did tend to make it easier to justify the work, it also meant that the research undertaken had to provide some deliverables. If it failed to do this, the project often came under intense scrutiny. Two different research projects of the AMD make this point clear. The first project was in the field of applied analysis and was headed by the mathematician Hans Kaper. His work was understood to be highly theoretical and was directed towards a deeper understanding of certain problems in gas dynamics by analyzing the relationship between kinetic and continuous evolution equations. Remarking on his work, the Review Committee noted "although the research is highly theoretical there is some possibility of practical application to problems involving extremely rarified gases."[141] Thus, they strongly suggested that support for this work continue.

In contrast, the mathematician Larry Wos was also doing highly theoretical work in a field called automated reasoning. Automated Reasoning was a subset of artificial intelligence (AI) that sought methods that would allow computers to solve mathematical

---

[139] Robert G. Sachs, to John T. Wilson. "Comments on the Report of the Review Committee for the Applied Mathematics Division," April 4, 1978. folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.
[140] Saul Rosen. Report of the Review Committee for the Applied Mathematics Division. Argonne: Argonne National Laboratory, 1979. folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.
[141] Ibid.

proofs by "reasoning." What "to reason" meant was hotly debated within the AI field as some researchers believed it should mimic the way humans solve problems while others sought methods that were distinctly different from that used by humans. Wos was a pioneer in automated reasoning and had been working on the "inhuman" approach to theorem proving since the early 1960s. He always had difficulty attracting support for his research, but in the mid 1970s he retooled his work as an experiment in software engineering. By this time, the "inhuman" approaches to theorem proving were out of favor in the AI community, but Wos and George Robinson, another Argonne mathematician, continued to work this angle. By the late 1970s, they had developed increasingly sophisticated "resolution-style" proof procedures called "demodulation" and "paramodulation" that helped theorem proving software fine-tune its reasoning.[142] Because their work was part of software engineering, Wos and Robinson produced a software program, AURA (AUtomated Reasoning Assistant), to test their theories. Despite the creation of a product, the Review Committee was critical of this work, stating "although Wos and his colleagues [had] produced a number of interesting computer proofs of a variety of theorems over the years, it is not completely clear where this research is leading, either in computer codes or in basic knowledge."[143] In light of this, the Review Committee was considering whether to recommend the cancellation of Wos' research project.

[142] For a discussion of automated reasoning and resolution, see Chapter 3, Donald MacKenzie, Mechanizing Proof: Computing, Risk, and Trust, ed. Weibe E. Bijker, W. Bernard Carlson, and Trevor Pinch, Inside Technology (Cambridge: The MIT Press, 2001). especially pp. 83-89
[143] Rosen. Report of the Review Committee for the Applied Mathematics Division.

Once again, the director of the Math and Computer Science section defended this experimental software engineering project (and thereby the theoretical research underlying it):

> "The objective of this work has been the development of basic knowledge about the kinds of inference techniques and strategies which are useful in automated proofs. The code has been used as a test-bed for various strategies, and the scientific results on the effects of using different procedures have been published in literature. In addition, the proofs of various theorems, including new ones, have also been published, both as matters of interest in their own right and as demonstrations that the procedures actually work. We are now concerned to see if the same approach will lead to progress in the automated proof of correctness of computer programs, and are also considering the desirability of rewriting the programs in portable form."[144]

The point here is that work considered theoretical and work considered experimental were held to different standards when it came to evaluations. That Kaper's work *might have* applications was enough; that Wos' work *was supposed to have applications* but hadn't demonstrated any made it suspect.

Ironically, by the early 1980s, Wos' and Robinson's work began to pay off as their software was applied to several highly specialized fields of mathematics to prove open conjectures that mathematicians had formed but were themselves unable to solve. In 1983, their work also won the first ever prize in Theorem Proving at the national meeting of the American Mathematical Society[145] and in 1996 it made the pages of the New York Times, when William McCune of Argonne used some of this software to solve a sixty year old open problem in Boolean algebra.[146]

By the end of the 1970s there is evidence that some researchers at Argonne were becoming interested in gaining access to supercomputers, particularly the aforementioned

---

[144] R.J. Royston. Comments on AMD Review Committee Report. Argonne: Argonne National Laboratory, 1979. folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.
[145] Gail W. Pieper. Major Accomplishments. Argonne: Argonne National Laboratory, 1985. Major Accomplishments, 1980-1994, box 1, MCS Archives, Argonne National Laboratory.
[146] MacKenzie, Mechanizing Proof: Computing, Risk, and Trust, pp. 88-9.

Cray 1.[147] The Cray-1, introduced in 1976, was a landmark supercomputer in part because it was the first to effectively exploit pipelined vector processors. In non-pipelined computers, the steps necessary to execute a single instruction in a program -- access and interpret the instruction, access the data, perform the operation and then return the answer to memory -- are performed sequentially. With pipelining, these operations are overlapped, so at the same time that the computer is performing an operation, it is also fetching the next instruction and so on. Vector processing is a hardware and software technique that allows one instruction to be performed on an entire ordered set of data simultaneously.[148] Both pipelining and vector processing were moves towards parallelism in computer design and thus were significant departures from the traditional von Neumann architecture, which performed operations sequentially. For some Argonne scientists, the vastly increased speed of the Cray 1 architecture meant that they could perform some experiments that were currently unfeasible.

In 1979, Argonne arranged for researchers to gain some access to the Cray 1 at the National Center for Atmospheric Research using the ARPANET. While this was considered highly desirable by the researchers, access to the Cray 1 presented its own problems. Foremost was that because of its parallel architecture, the algorithms and software packages that they used had been written for serial machines (von Neumann) and either would not work on the Cray or were woefully inefficient. Without new software, then, these researchers would not be able to take advantage of the Cray's power. Recognizing this, the Review Committee for the AMD recommended that the Mathematics and Computer Science section begin to investigate ways to write algorithms

[147] Rosen. Report of the Review Committee for the Applied Mathematics Division.
[148] MacKenzie, "Nuclear Weapons Laboratories and the Development of Supercomputing," p. 102-3.

for parallel machines and encouraged them to add this as a research component in their network activities.[149]

That same year, Paul Messina took over as Associate Director for Applied Mathematical Sciences, replacing Jim Pool, who then went to the DoE. A PhD in mathematics, Messina arrived at Argonne in 1973 and spent several years as the head of the User Services Group which dealt directly with other scientists at the Lab who had special computing needs.[150] As will be shown, this experience would prove important because it gave Messina a strong foundation in working with scientists in an interdisciplinary setting.

During his tenure, Pool had been instrumental in taking a group of researchers who were primarily analysis oriented and reshaping their work to include a strong software engineering component. In so doing, Pool enabled the work of the Math and Computer Science section to incorporate a strong experimental component which in turn kept the unit viable and relevant throughout the 1970s. Now, under Paul Messina's direction, the Mathematics and Computer Science section, which now numbered eighteen full-time researchers and four visiting scientists, was again reinvented as he began their transition into the world of parallel computing. As part of this transition, and also a reflection of his time as the head of User Services, Messina made a strong effort to reintroduce interdisciplinary collaboration in computing.

To give an idea of the rapidity with which supercomputing burst onto the scientific scene, despite the interest of some Argonne scientists in supercomputing and

---

[149] Rosen. Report of the Review Committee for the Applied Mathematics Division.
[150] Personal Interview with Paul Messina by author, May 13, 2002 at Argonne National Laboratory. Also see R.J. Royston. The Applied Mathematics Division in 1980. Argonne: Argonne National Laboratory, 1980. folder 4, box 154, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

the encouragement of the Review Committee to explore networks connections to them, in 1980 the AMD reported "our exploration of the use of Cray computers has shown that while such a facility would be valuable to certain scientific programs, there was certainly not a large enough demand to justify the serious consideration of procuring such a machine for the Laboratory."[151] Here was a National Laboratory doing work on the cutting edge of science, and it still found little justification to support supercomputing. This assessment further supports my contention that the emergence of computational science was not technologically determined, but instead, rested on events outside of the technical realm.

In 1980, as supercomputers began to incorporate more elements of parallel processing in their architectures, Paul Messina became convinced that the future of scientific computing lay in these machines. He was not the only one; it was widely believed that current supercomputers were approaching their physical limits in terms of speed.[152] For the most part, supercomputer manufacturers had been making incremental improvements in the von Neumann architecture which was already forty years old. Speed increases had been achieved by cramming more connections into chips and by shortening the interconnections between computer components. However, there were limits to both, and the fact that the Cray-1 had to be immersed in liquid Freon to keep it cool suggested that the laws of physics would impede future improvements.

---

[151] R.J. Royston. Response to the Report of the Aua Review Committee for the Applied Mathematics Division. Argonne: Argonne National Laboratory, 1980. folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.
[152] Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993, pp. 34-5.

That nature imposed limits on the development of computer technology is what historian Edward Constant would call a presumptive anomaly.[153] In scientific computing, the recognition of a presumptive anomaly in current designs allowed computer designers to begin exploring truly parallel computers. The idea behind this is simple. Why have the data pass through a single massive processor, why not break the problem down and have it computed on several (or many) processors simultaneously? Each processor could access the memory independently, pass messages to others when required, and then reassemble the answer at the end. The limits of these machines would thus be based on the number of processors and the creativity of the designers of the hardware and software rather than the physical distance between components.

Not only was such a machine feasible, but a parallel computer along these lines had already been built. With support from the Information Processing Techniques Office of ARPA, in 1972 the University of Illinois completed the design and construction of the ILLIAC IV. The computer, which at the time was the fastest in the world, incorporated 256 processors arranged in quadrants of 64 processors each. The machine was enormously expensive to build and notoriously difficult to program -- a characteristic innate in parallel machines. The ILLIAC was shut down in 1981 just as there was renewed interest in parallel computers. But this experimental project taught computer engineers and computer scientists lessons both economic and scientific. Thus, instead of

---

[153] Edward W. Constant II, The Origins of the Turbojet Revolution (Baltimore: Johns Hopkins University Press, 1980). Constant argued that a presumptive anomaly arises when an older technology still functions and may even provide room for some additional development, but it becomes increasingly apparent that future advances in that technology will have to be based on radically different foundations. For Constant, he examined the development of the turbojet for airplanes. Given the limits of streamlining aircraft, it became clear to some aerodynamicists that the propeller would quickly become the limiting factor to increasing speed. Recognition of this presumptive anomaly allowed a small number of scientists to begin looking at new technologies, such as the turbojet, while the existing technology (piston engines and propellers) were still working well. Eventually, the research led to a revolution in aircraft design as the turbojets became a technical and commercial success.

building a large parallel computer straight away, it was better to build prototypes of maybe 64 processors on which programming experiments could be run, and then scale them up to a massively parallel machine a thousand times that size.[154]

In an interview I had with Messina, he claimed that the true catalyst for his conviction that parallel computers were the future arose from a meeting he attended in early 1981 at the Department of Energy headquarters in Germantown, MD. The meeting was held so that all the directors of math and computer science programs within the DoE could share their work with each other. One of the presentations was by a group at Los Alamos who were building a parallel computer that linked together eight INTEL 8086 microprocessors. A second presentation was by Jack Schwartz of the Courant Institute, who was also trying something similar. Messina says that upon his return to Argonne, he immediately called a meeting of his researchers and told them that while their numerical software was currently used by scientists and engineers around the world, in order to remain relevant in a decade their software must run on parallel computers.[155] What this meant is that his researchers needed access to parallel machines in order to learn how to write programs for them. The question, then, was how to acquire parallel machines, especially considering that there were none yet available commercially.

It would be two years before Messina was able to secure a parallel machine, but before that he was able to achieve a tremendous victory of a different sort. Sometime in late 1981, he convinced the Laboratory Director that the Math and Computer Science section of the Applied Mathematics Division should be spun off as its own division.

---

[154] For a discussion of the ILLIAC IV project see Arthur L. Norberg and Judy E. O'Neill, Transforming Computer Technology: Information Processing for the Pentagon, 1962-1986, ed. Merritt Roe Smith, Johns Hopkins Studies in the History of Technology (Baltimore: The Johns Hopkins University Press, 1996), especially chapter 6.
[155] Personal Interview with Paul Messina by author, May 13, 2002 at Argonne National Laboratory

Messina argued that the services of the Central Computing Facility were so time consuming for the current director that the math and computer science section was not receiving proper attention. Jim Poole, who was now the Program Director, Applied Mathematical Science in the Department of Energy, strongly supported this effort and in mid-1982, the Math and Computer Science (MCS) Division was created with Paul Messina as its first director.[156]

Divisional status brought with it a new voice in the affairs of Argonne. As Messina mentioned in his interview, as a Division Director he now had a say in helping to determine Laboratory priorities. Mathematicians and computer scientists had not enjoyed influence in Laboratory affairs at this level since the resignation of Wallace Givens in 1970, and then Givens had been responsible for the computer service activities, too. As has been discussed, the socio-scientific arc of computer scientists and applied mathematicians had changed tremendously since the early 1970s. By 1981, on the cusp of the Japanese Fifth Generation announcement, computer scientists had for the most part established their credentials as legitimate scientists. To members of the former section their designation as a Division was recognition of this professional status. Messina further justified the move to ANL directors, saying that Divisional status raised morale and thereby contributed to greater productivity by his researchers.[157]

Securing Divisional status also allowed Messina broader latitude in shaping the overall mission of his researchers. Shortly after the Germantown meeting in 1981, Messina moved aggressively to channel his researchers' expertise towards parallel

---

[156] Personal Interview with Paul Messina by author, May 13, 2002 at Argonne National Laboratory; Interview with Jim Pool by Tom Haigh, CalTech, July 14, 2004.
[157] Paul C. Messina, to K.L. Kliewer. "MCS Division Accomplishments, July 1, 1982 - June 30, 1983," May 31, 1983. Major Accomplishments 1980-1994, box 1, MCS Archives, Argonne National Laboratory.

computers and supercomputers, but his efforts did not go uncontested. In 1982-1983, MCS had three basic core strengths – Applied Analysis, Computational Mathematics (which included the EISPACK project and optimization programs), and Software Engineering (which included efforts like TAMPR and automated reasoning). Of these, the group that would be affected most directly by a move into supercomputers was Computational Mathematics. Some of the researchers were very eager for this opportunity; Jack Dongarra, for instance, had been in contact with Cray and at their invitation had been running experiments on their new machine, the Cray XMP, for months. In addition, he had invited numerous speakers to MCS to discuss supercomputing initiatives and was helping Messina put together a proposal for the Division to acquire an advanced computer. On the other hand, the computational mathematicians working in the area of optimization were not sure that there was anything significant to be done in their field related to advanced architectures. They argued that in their field there were few efficient algorithms developed for sequential computers and that before they joined the "stampede" to study algorithms on parallel machines, they first needed to learn how to do them for standard architectures. While Messina felt that this was a "healthy discussion" he contended that these concerns would subside if only the computational mathematicians were given reasonable support and access to the advanced equipment. Another mathematician, Joe Cook, who had been with the Division since the early 1960s, was so disillusioned by the new initiative that he decided to leave entirely.

While there was some conflict within MCS over the new emphasis on supercomputers, in reports to his superiors, Messina increasingly presented the research of the division as either a direct contribution to high-performance computing or as applicable to it. By mid1983 Dongarra's algorithm experiments on the Cray XMP (introduced in 1982) led to the development of techniques by which so-called "supervector" speeds could be achieved with standard programs written in FORTRAN. Likewise, the automated reasoning program spearheaded by Larry Wos, which by this time was beginning to produce substantial results, became a natural candidate for experiments on parallel architectures because of the compute-intensive nature of the software. Finally, the need to translate existing programs so that they could be run on new architectures meant that the software engineering effort in automatic program transformation begun with TAMPR in the 1970s was ripe for exploitation. An outgrowth of this original project was TOOLPACK, a collection of software tools that aided in writing, modifying, and transforming computer programs so they could be run on different computers. Although TOOLPACK's main designers were from the MCS, it was a collaborative effort involving researchers at the Universities of Colorado and Arizona, Purdue, the Jet Propulsion Laboratory, Bell Communications Research, and the Numerical Algorithms Group, LTD (NAG), and thus had been justified as experimental again in part because it used the ARPANET extensively. Although the package would not be completed until 1985, Messina pointed out to his superiors that both the effort and underlying approach of the project had been publicly endorsed by Nobelist Kenneth Wilson as highly promising for use on advanced scientific computers.[158]

Messina went even further than simply recasting current MCS research as potentially useful to supercomputing. At some time in 1982, he learned that Los Alamos had acquired a DENELCOR Heterogeneous Element Processor (HEP). Designed by Burton Smith of MIT, this was the first semi-commercial parallel computer available.

[158] Ibid.

Almost immediately, Messina sent Rusty Lust, Ross Overbeck, and Jack Dongarra to New Mexico to work on the machine. Upon their return, Messina approached Argonne's Laboratory Director with a proposal to acquire their own HEP, based on the idea that it would allow MCS researchers to experiment with different ways to write software for parallel machines.

To support his case, in October 1983, he reported some preliminary results of research conducted by the MCS on these machines:

> "Argonne scientists have been experimenting with typical programs from widely used mathematical software libraries to find ways to exploit multiprocessing environments. They have found that algorithms can be expressed in terms of a few commonly-used high-level modules. The modules can then be programmed separately, verified at each step of the development process, and tailored for particular machines. Significantly, this approach preserves a high level of software portability. Early results with linear algebra programs have been promising. Minimal changes to the software were needed to produce near optimal performance on the Cray-1M, Cray X-MP, and Denelcor HEP, each of which supports a different concept of multiprocessing. Further work is under way to use this approach in other software areas and to develop new algorithms that take full advantage of anticipated future multiprocessing concepts."[159]

The 1983 MCS Review Committee applauded Messina's activities and recommended the Division "study the option available to it in the supercomputer area, given its current lack of expertise and equipment."[160] At the time, it was still unclear whether the DoE was going to finance the establishment of supercomputers but the Committee recommended that the Division try to acquire its own HEP upon which to experiment. If this could not be achieved, it was suggested that the MCS attempt to collaborate with another group

---

[159] Paul C. Messina, to K.L. Kliewer. "Programming in a Multiprocessing Environment," October 17, 1983. Major Accomplishments 1980-1994, box 1, MCS Archives, Argonne National Laboratory.
[160] Susan L Gerhart. Review Committee Report Mathematics and Computer Science Division. Argonne: Argonne National Laboratory, 1983. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.

such as the Ultracomputer project at the Courant Institute, the TRAC project at Texas, the Blue Chip project at Purdue, or the Finite Element Machine project at NASA-Langley.[161]

In his response to the review, Messina laid out his plans to move MCS into parallel computing. There were two main thrusts. The first, he said, was “a basic research program aimed at understanding interactions between architecture, software, and algorithms for advanced computers when applied to many of our areas of expertise: numerical methods in optimization, linear algebra, quadrature, and partial differential equations (PDE’s); automated reasoning and logic programming; and programming languages, techniques, and development aids.”[162] The second thrust was more ambitious: the establishment and operation of an Experimental Computing Facility (ECF). Messina’s vision was to install a succession of leading-edge computers with advanced architectures and then make them available for experimentation. At its heart, the facility would be a collaborative project:

> “The ECF will provide a superb research facility for the basic research components of our program. In addition, it will serve as a national user facility for studying how to use computers with advanced architectures and how to design better ones. The ECF will provide a natural focus for increased interactions between our research staff and other Laboratory staff, university faculty, and industrial researchers.”[163]

Messina was able to secure a Program Development Funds grant from Argonne to establish the ECF after which ANL entered into a joint research and development agreement with Denelcor Corporation that led to installation of an HEP in April 1984.

---

161 Ibid.
162 Paul C. Messina, to Hanna H. Gray. "Response to Mcs 1983 Review Committee Report," February 10, 1984. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.
163 Ibid. In total, only six HEP systems were delivered to customers during the years 1981-1985. In addition to Los Alamos and Argonne, the Ballistics Research Laboratory (BRL) received one, as did Messerschmidt in Germany. Of these, only Messerschmidt used it for production work while the labs used it extensively for research on parallel algorithms. See "Proceedings of the Workshop on Parallel Processing Using the Heterogeneous Element Processor," ed. S. Lakshmivarahan (Norman: The University of Oklahoma, 1985).

(See page 282 for a picture of the HEP) Messina was a good salesman; ANL Directors proposed that these activities be made a major Laboratory initiative and plans began to present their case at the federal level.

The effort to reorient MCS work toward supercomputing during the preceding two years was a tactical masterpiece. The Japanese Fifth Generation program had filtered into the American scientific, political, and public consciousness at about the same time that Messina secured the HEP. He had thus positioned the Division supremely well to take advantage of the forthcoming U.S. response. In addition, he had forged ties with computational scientists like Ken Wilson and had been able to expose some of his scientists to state-of-the-art supercomputers like the Cray XMP.

Although the ECF was ambitious, there was still quite a bit of uncertainty among subsequent Review Committees as to how it fit into the overall mission of ANL. The 1984 committee noted that while MCS had gotten "substantial mileage out of having made this move . . . it was not particularly good for satisfying the Argonne computing needs."[164] When it was pointed out that this was not the intent of the ECF, the committee then suggest that increased visibility was essential to its success and that the effort "would benefit by the addition of a superstar."[165] This suggestion generated considerable debate, but the more important point is that even in 1984, *after the Japanese announcement,* there was little consensus as to the relationship between computer science, computational science, supercomputing, and the standard production work to which computers had historically been applied.

[164] K.L. Kliewer, to P. Messina. "Comments on Exit Interview with Mcs Review Committee, Friday, 15 June 1984," June 20, 1984. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.
[165] Ibid.

At the same time though, Messina's vision for the future of the ECF was nothing less than prescient. He had begun with one machine, the HEP, in 1984. Now he was pushing to expand this into the Advanced Computer Research Facility (ACRF). In seeking to justify the ACRF, Messina played on the uncertainty that swirled around parallel computing technology, namely, *what* technology was going to be important. Again, he was not alone in this; DARPA's SCI program faced the same problem. The goal of SCI was to develop intelligent machines. Even at the beginning of the program, it was clear that achieving this goal would require computers faster than current supercomputers by several orders of magnitude. While this, in turn, strongly suggested that parallel machines were the future, there was no consensus as to which architecture was the most promising. The answer for SCI, not surprisingly, was to fund several parallel computer projects in the hopes that by pursuing different approaches to parallelism concurrently it could speed the process by which good ideas were weeded from bad ones.[166]

In a similar way, Messina argued that rather than wait for one parallel technology to rise to the top, it was prudent to acquire experimental machines of different architectures as they were developed, so that his researchers could gain experience developing software for them. Unlike the SC program, the ACRF was not interested in artificial intelligence. Instead, research would be geared towards ways to make parallel computers useable by computational scientists and for scientific computing in general.

[166] See Chapter 5 of Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993.

As proposed, the ACRF was established as a place "where scientists can experiment with innovative machines and develop programming tools for state-of-the-art computers."[167]

Messina recognized that new parallel computers were prohibitively expensive, so he proposed to work directly with the manufacturers when the machines were still in their early stages of development:

> "The key word here is *experiment*. We are willing to acquire and install in the ACRF machines in early stages of their development, when the software is sometimes minimal and the operating systems still may have 'bugs' in them. Our intent is to gain familiarity with various computer architectures at the forefront of technology – perhaps even to make suggestion for the next generation of computers."[168]

Leveraging the expertise within the MCS (which had for several years been recast for this very purpose), Messina argued that the main objective for his researchers was to determine ways to ensure high performance and portability of algorithms on advanced computers. "It is simply not practicable," he wrote, "to develop new methods, algorithms, and programs for each new computer design. At the same time, achieving portability at the expense of optimum performance is equally unacceptable and could seriously hamper the ability to utilize machines of the future effectively."

Beyond providing the experimental equipment long desired by computer scientists, the ACRF was purposely organized so as to allow other scientists to get their hands on these computers. Messina realized that the value of the enterprise hinged to a large extent on turning out real work, there was a strong effort to encourage scientists to bring their problems to the ACRF, where they could collaborate with members of the MCS on these machines. Casting the facility in this light also allowed him to head off

[167] Paul C. Messina, to Donald Austin. "Advanced Computing Research Seeks Answers to Questions Raised by Multiprocessing," May 1, 1986. Major Accomplishments 1980-1994, box 1, MCS Archives, Argonne National Laboratory.
[168] Ibid.

criticism that the facility did not contribute to the computing needs of Argonne scientists. As Messina wrote, the ACRF benefited Argonne's "production" computing environment in five ways: it assessed the suitability of different architectures; allowed researchers to learn how to use the machines before production work was put on them; allowed continual study of future systems; provided a head start in integrating these systems into Argonne's computing environment; and provided the opportunity to influence vendor's designs.[169] In addition, the MCS offered regular classes on parallel computing to scientists at Argonne, sponsored an extensive visitors program, ran symposia on parallel computing for industrial and university scientists in the Chicago area, developed a workshop on language issues for parallel computer manufacturers, and ran a week-long summer institute on parallel computing for university students.[170]

Although these educational programs were crucial to the success of the ACRF, possibly more important was that the experimental machines were accessible to outside researchers through national networks such as ARPANET, MILNET, and TYMNET. Network connections certainly helped the proposal fly in Washington. During his time as Director, Applied Mathematical Science at the DoE, Jim Pool put in place several requirements that had to be met for those seeking funds to purchase experimental computer equipment. Number one was that the computers had to be connected to the ARPANET; number four was that these machines could not be operated by the Computing Center, but had to be placed within a research group.[171] Even though Pool had moved on by 1984, these policies remained in place. But beyond satisfying funding

[169] Paul Messina. Advanced Computing Research Facility, Review Committee Meeting. Argonne, 1985. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.
[170] Messina, to "Advanced Computing Research Seeks Answers to Questions Raised by Multiprocessing,"
[171] Interview with Jim Pool by Tom Haigh, CalTech, July 14, 2004.

requirements, the ACRF was also designed to address what was now seen as a national concern: how to increase access for scientists to the kinds of architectures that would power supercomputers in the future.

In 1984, the ACRF proposal was imaginative, feasible, and above all, timely. The MCS Division was able to demonstrate that they had expertise in algorithm development and software design, experience with parallel machines (HEP and Crays), and a large scientific community that might be interested in high-performance computing. In 1985, Messina received approximately $2.5 million from Argonne and the DoE to officially establish the Advanced Computer Research Facility.[172]

Almost immediately, Messina began looking for ways to expand the program; he was hoping to build a case for the DoE to more than double its support -- to $8 million -- by 1988. He realized that this amount of funding would be difficult to justify if it was perceived to be used primarily for computer science research. Consequently, in 1985 he initiated efforts to establish a large-scale scientific computing group. This action is significant, for in looking at his proposal it is clear the extent to which the arguments of computational scientists were being adopted by the computer science community in order to support their own activities. In a letter to K.L. Kliewer, the Associate Laboratory Director for Physical Research at Argonne, Messina wrote:

> "Such a group will close a gap in our activities and will ensure that both our Applied Analysis and our Computational Mathematics research can be used for serious scientific simulation and modeling. The latter activity is emerging as the third scientific research method, alongside theory and experimentation. A strong MCS effort in this area should help all of Argonne in acquiring the capability to use computer simulation in research."

[172] Paul C. Messina, to K.L. Kliewer. ""New" Strategic Plan," November 26, 1985. Major Accomplishments 1980-1994, box 1, MCS Archives, Argonne National Laboratory.

The creation of a large-scale scientific computing group was nothing less than an effort to institutionalize the interdisciplinary collaboration between scientists, computer scientists, and mathematicians that William Miller had envisioned back in the early 1960s. The initial focus of the scientific computing activity was on the development and analysis of methods for the numerical solution of problems involving reaction flows and especially problems involving combustion. This effort would be strongly computationally oriented, rely heavily on supercomputers, and would link together several computational mathematics projects with the work of scientists in other divisions.

The idea of establishing a scientific computing group was commendable; actually doing it proved to be much more difficult in 1985 and this says much about the nascent state of computational science. Messina had hoped that the computational scientist Alvin Bayliss would join the faculty at Northwestern University and then accept a joint appointment in the MCS Division. However, he chose not to come to Chicago, so Messina approached two other researchers, only one of whom was in the area, regarding the possibility of a joint appointment at Argonne and another university. The difficulty in attracting computational scientists to work on these machines is indicative of both their scarcity and their demand at the time. Messina feared that hiring only one person for this type of work would be ineffective, especially since that person would need support people for the programming effort involved in large-scale scientific computing. Better would be two or three researchers, but that, in turn, required additional money from the DoE.[173]

[173] Paul Messina. Response to MCS Review Committee Report. Argonne: Argonne National Laboratory, 1985. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.

Despite the early troubles in forming the group, the facility itself continued to expand. By 1986, the ACRF had acquired five more commercial experimental machines: an Encore Multimax with 20 processors sharing 20 megabytes of memory; a Sequent Balance 8000 with 12 processors sharing 16 megabytes of memory; an Alliant FX/8 with 8 vector processors sharing 32 megabytes of memory; an Intel iPSC hybercube system with 32 nodes, each having half a megabyte of memory; and another machine designed by Ray Hagstrom from Argonne's High Energy Physics Division. This quickly grew to seven by 1988, as the MCS added a 16-processor Intel iPSC-VX hypercube, a 1024-processor Active Memory Technology DAP, and a 16384-processor Thinking Machines CM-2.[174]

On the cusp of the formulation the High-Performance Computing Initiative, Messina's brainchild had established itself as a leading research facility for interdisciplinary teams of computer scientists, mathematicians, and computational scientists to solve problems on novel computer architectures. Moreover, the ARCF had forged ties to both industry and academia. In 1990, there were fifteen industrial affiliates and twenty academic affiliates, including universities in Australia and Newfoundland. The facility boasted 1100 users with about 315 active each month; 60% of whom came from universities, 12% from industrial labs, and 28% from government labs. Although the majority of the work done in the facility (65.3%) was for math and computer science

[174] William J. Cody, to E. Vanberkum. "Major Accomplishments for MCS 1983-88," June 22, 1988. Major Accomplishments 1980-1994, box 1, MCS Archives, Argonne National Laboratory, Messina, to "Advanced Computing Research Seeks Answers to Questions Raised by Multiprocessing,"

research, 17.1% of it was in the physical sciences, 9.2% in programming languages and tools, 6.3% in scientific visualization, and 2.1% in biology.[175]

That the preponderance of research in the ACRF was related to math and computer science research was deliberate. While Messina had created the large scale scientific computing group as a venue for computational scientists to gain access to advanced architectures, he actively discouraged the use of these machines for production purposes. In this, the ACRF stayed true to the long-held argument that mathematicians and computer scientists had to be free from service duties. What I want to stress however, is that extent to which the ACRF also reflected a particular conception of computer science *as a science*.

Even in 1987, the issue as to whether computer science was a science continued. On the fortieth anniversary of the Association of Computing Machinery, Peter Denning addressed this issue yet again. In particular, the debate now hinged on whether computer science was really an engineering discipline. For almost two decades (going back to his 1972 article) Denning had been trying to show that computer science had a strong experimental tradition, but his attempts had fallen short, primarily because they could not explain the presence of subdisciplines such as computer system design, which was part of computer science, but did not fit the classical scientific tradition. Yet science and engineering were inseparable in CS because of an overriding interest in efficiency as both computer designers and mathematicians sought the most efficient way to achieve a

[175] ACRF Industrial and Academic Affiliates. Argonne: Argonne National Laboratory, 1990. Meetings with Don Austin, box 2, MCS Archives, Argonne National Laboratory.

particular goal. In an article, “Paradigms Crossed”, Denning sought some way to reconcile these two.[176]

Denning believed that the solution lay in the recognizing the interaction of the central traditions of science and engineering. In experimentation, he argued that researchers follow four steps, which could all be iterative, in the investigation of a hypothesis: “design the experiment, collect the data, analyze the results, and share the findings.” The central process for engineering was design, which also consisted of four iterative steps followed in the construction of a device: “state requirements and specifications, implement the device, test the device, and share the findings.”[177] Denning contends that many of the debates about the relative value of science or engineering are based on the assumption that one of the two main processes -- experiment or design -- is more fundamental. What needed to be recognized is that these processes are hopelessly intertwined:

> “At every stage of the experimentation process, you can observe instances of the design process – for example, in the design of the experimental apparatus, in the design of the data collection procedures, in the design of the analytic methods, and in the design of the mechanisms for disseminating information. At every stage of the design process, you can observe instances of the experimental process – for example, in verifying hypotheses about the requirements, in discovering whether a particular specification language is effective, in evaluating alternative design decisions, in assessing the accuracy of a simulator versus its speed, and in testing whether a device meets its specifications. Thus there is much truth in the statement that experimental physicists are good electrical engineers, and also in the statement that electrical engineers are good experimental physicists.”[178]

The same holds true for computer scientists, argued Denning. Depending upon which subfield of CS one looked at, either the experiment or design paradigm was dominant. In scientific computing, performance analysis of systems, or the testing of algorithms, the

---

[176] Peter Denning, "Paradigms Crossed," Communications of the ACM 30, no. 10 (1987).
[177] Ibid.: p. 808.
[178] Ibid.: p. 809.

scientific experiment process is dominant. In subfields like computer architecture and operating systems, the design process dominates. Uniting and underlying both paradigms is, of course, mathematics. Moreover, he said these two processes are inseparable. Experimentalists use supercomputers to explore mathematical models and to run simulations, and use networks to disseminate their findings from these scientific experiments. Without these experiments, computer designers could not do their work, for activities such as the design of new computer chips would be impossible without logic simulation. At the same time, experimentalists would be unable to do their work without supercomputers or networks which were designed by computer engineers. Computer science was a unique blend of these two central processes of experimentation and design, and thus stood at the crossroads between science and engineering.

Denning's project was more than a rhetorical exercise; there were practical reasons related to issues such as funding and institutional status for reconciling the scientific and engineering aspects of the discipline. While new collaborative initiatives like Paul Messina's ACRF were creating an institutional setting in which the different elements of computer science -- experiment and design -- could flourish, at the same time the *perceived value* of these facilities might well hinge on whether the work is considered primarily science or engineering. As I will discuss briefly in the concluding chapter, Denning's concern may be well justified.

The Federal High-Performance Computing Initiative

The FCCSET panel's 1983 report suggested that federal agencies set up their own supercomputer centers, and not surprisingly, many of the agencies did just that. The

DoD, NSF, DoE, NASA, Department of Commerce (DoC), the National Bureau of Standards (NBS), National Security Agency (NSA) and the CIA all provided support for basic research in supercomputing. Also not surprising was that the lack of cooperation between these different agencies meant that some areas of research were neglected entirely, while there was duplication of efforts in other areas. Although the NSF was slated to establish five academic supercomputer centers in 1985 -- at San Diego, Pittsburgh, Illinois, Princeton, and Cornell (Ken Wilson's institutional home) -- in general, institutions were reinventing the wheel. According to one estimate, in 1985 there were thirty-eight separate parallel architecture projects supported by seven different organizations.[179]

Recognition that the lack of federal coordination was leading to duplicative efforts spurred FCCSET in 1983 to begin laying plans for what would become the High Performance Computing Initiative. FCCSET was part of the Executive Branch and had been established in 1975 by Congress as an interagency committee operating under the Office of Science and Technology Policy (OSTP). Its mission was to serve as an advisor on technical matters to the Executive branch and to look for ways to coordinate policies regarding these issues. In spring 1983, FCCSET created three panels. Two of these addressed questions concerning the acquisition of and access to supercomputers. The third panel focused on issues concerning symbolic processing and artificial intelligence.[180]

[179] Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993, p. 289. As my intent is not to offer a new analysis of the political process that lead to the creation of the HPC program, the following discussion draws heavily on this text, especially chapter 9.
[180] Ibid., p. 290.

Despite the urgency to coordinate high-performance computing initiatives, the FCCSET panels moved with the speed of bureaucracy; after almost three years of meetings they had yet to produce a comprehensive plan. One group that did make a real contribution was the networks working group which had been established as a fourth panel in 1985. Networking, which had been one of computer science's most successful experimental projects, was beginning to garner attention at the highest levels of policy-making. After studying industrial R&D and commercial developments in the U.S. and abroad, in February 1986 the panel recommended that all existing federal telecommunications networks should be interconnected.[181]

Networking also began to attract Congressional attention. Specifically, Senator Albert Gore, Jr., of Tennessee had become enamored of the possibility of creating an "information superhighway" based on computer networks. Under his leadership, in 1986 Congress directed the OSTP to examine the state of computer networks in the United States. In early 1987, President Ronald Reagan's science advisor, William Graham, also became interested in networks. However, he asked for the OSTP investigation to not only provide an inventory of network activity, but also to investigate the context in which they were being used. After numerous workshops involving hundreds of researchers in academia, industry, and government, as well as soliciting reports from the NSF, in mid 1987 the OSTP released their recommendations. One report proposed a "national computing initiative" that would focus on creating supercomputers using scalable parallel processing; a second report called for a "national research network" to develop high-

[181] Ibid., p. 293.

speed networks. These two proposals became the basis for the High-Performance Computing and Communications program.[182]

In November 1987, the OSTP released its report A Research and Development Strategy for High Performance Computing which William Graham forwarded to Congress for debate. This document laid out a five-year strategy for federally supported research and development on high performance computing, including hardware for supercomputers, software, computer networks, and supporting infrastructure (education). The report called for a coordinated federal effort to advance these areas that would unite government, industry, and academic researchers in closely knit collaborative projects. Federal money would be used to offset the risk of developing these new technologies; then the results of this research would be transferred to the private sector for commercialization.[183]

The strategy was just that -- a strategy. It did not say how the goals of the program would be accomplished, but it did provide a preliminary budget estimate in the neighborhood of $1.75 billion with about half of that money directed into basic research. This total was in addition to the $2.5 billion that was already earmarked for high-performance computing in the various federal agencies over the same period of time.[184]

For the computer science community, the FCCSET report was widely heralded as the dawning of a new day for the discipline. In the wake of the report, several federal agencies began to consider research programs that responded to its main points. As one senior policy analyst at the OSTP remarked, "It's [the report] getting tremendous play, and it's not even controversial. As a matter of fact, one of the criticisms we've heard is

[182] Ibid., pp. 294-95.
[183] Ibid., p. 295.
[184] Ibid., pp. 295-96.

that it does not go far enough in telling us what to do."[185] Congress was especially interested; in August 1988 Senator Gore held a series of hearings on the FCCSET report after which he asked the OSTP to formulate a plan by which the recommendations might be implemented. However, what was considered precedent setting by computer scientists was not the report itself, but divergent groups that came together to create it:

> "It is the first time in the history of the field that senior people from all the major federal agencies having an interest in computing, and the senior people as the OSTP, got together and produced an executive summary on what the problem is and what can be done. That's never happened before."[186]

The FCCSET report was seen by many as the beginning of a revolution for computer science as parallel computing would eventually move out of the laboratories and into industry, and as network technology proliferated throughout science and engineering. In the words of Stephen L Squires, a former director of the SCI program for DARPA and a chief architect of the FCCSET report, this revolution included a general reassessment of the role that computer science played in the advancement of science:

> "The first part of the revolution is that science and engineers are, for the first time, taking computing seriously. The second part is that the transition from conventional computing to scalable parallel computing means that the entire field of computer science must reinvent itself. It will require revolutionizing all aspects of computer science by building upon our past success."[187]

Despite the widespread enthusiasm for the report, it wasn't until September 8, 1989, almost two years after their first report, that the OSTP issued The Federal High Performance Computing Program (HPC).[188]

---

[185] Diane Crawford, "U.S. Computing Strategy One Year Later," Communications of the ACM 31, no. 10 (1988): p. 1172.
[186] Ibid.
[187] Ibid.: p. 1174.
[188] Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993, pp. 296, 301.

The HPC plan was forwarded to Congress by D. Allan Bromley, who had succeeded William Graham as Director of the OSTP. His cover letter to the transmittal incorporates the arguments that had been made about supercomputing by computer and computational scientists since 1983:

> "High performance computing is a vital and strategic technology, exerting strong leverage on the rest of the computer industry and other cutting-edge areas. However, U.S. leadership and diversity in the supercomputer industry itself has declined dramatically; and history shows that a scant 15 years separates the first appearance of a top-of-the-line supercomputer from the appearance of that same computing power in the higher end of the personal computer market. A future national high speed computer network could have the kind of catalytic effect on our society, industries, and universities that the telephone system has had during the twentieth century. We cannot afford to cede our historical leadership in high performance computing and in its applications."[189]

As laid out in this new report, the HPC program sought to maintain and extend U.S. leadership in this critical technology, encourage innovation by increasing the diffusion and assimilation of these tools into science and engineering communities, and support U.S. competitiveness by transferring the developed technologies to industry. That computational science had arrived is clear in the Executive Summary of the programs description:

> "High performance computing offers scientists and engineers the opportunity to simulate conditions that are difficult or impossible to create and measure. This new paradigm of computational science and engineering offers an important complement to traditional theoretical and experimental approaches."[190]

The goals of the HPC would be achieved through coordinated, concurrent efforts in four areas. In High Performance Computing Systems, the report called on the federal government to purchase experimental computer systems, especially of parallel design, and then make them available to the research community. A second area of

[189] D. Allan Bromley. The Federal High Performance Computing Program. Washington, D.C.: Office of Science and Technology Policy Executive Office of the President, 1987. HPCC Program Correspondence Files 1990-1991, box 5, MCS Archives, Argonne National Laboratory.
[190] Ibid.

concentration was the creation of a National Research and Education Network (NREN) which would develop and deploy a gigabit (billion bits per second) network that would allow industry, academia, and government to share computer resources. The third component of HPC was vastly to increase funding for training workers in new computer technologies.[191]

Finally, the centerpiece of the program was research in Advanced Software Technology and Algorithms. This initiative was intended to develop the tools necessary to apply these computers effectively. Because improvements in software were responsible for most of the improvements in computational performance, this component sought new ways to improve software for the parallel computers developed in the computing systems program. Significantly, the manner in which this research effort is structured is a direct reflection of the new cultural and scientific power wielded by computational science, and stimulating computational science research was the primary goal of this program. Taking a page from SCI, again it would be applications that drove the technology. The architects of SCI firmly believed that if the end goal could be defined in enough detail, it would guide the development of the technologies and techniques needed to attain that goal. But, in the case of the HPC program, and the Software Technology and Algorithms component in particular, the end goal was much more open-ended. Rather than a quest for machine intelligence, this component was framed in the context of solving "Grand Challenges." Coined by Ken Wilson in 1987, and placed at the core of the HPC, Grand Challenges were "fundamental problems in science and engineering, with broad economic and scientific impact that could be

[191] Ibid.

advanced by applying high performance computing resources."[192] If SCI might be characterized as a well-defined problem in search of a solution, I contend that the HPC represents a solution in search of an ill-defined problem.

Moreover, the Software Technology and Algorithms component laid out not only the kinds of problems it would address (vague Grand Challenges), but it also provided the organizational framework in which these problems would be tackled. In the HPC Program, Grand Challenge problems were to be tackled *collaboratively*:

> "Collaborative groups will include scientists and engineers concerned with Grand Challenge areas, software and systems engineers, and algorithm designers. These groups will be supported by shared computational and experimental facilities, including professional software engineering support teams, linked together by the National Research and Education Network. Groups may also create a central administrative base, which can be located anywhere on the network."[193]

To foster computational scientists, the HPC program supported the creation of high performance computing research centers which would perform experiments on prototype computers having novel architectures. The specific inclusion of experimental computers is also significant because it validates the two-decade old effort of computer scientists and mathematicians to cast themselves as experimentalists in need of the latest equipment. As the report made clear:

> "Researchers in areas such as algorithms, software environments, and operating systems require experimental access to new generation hardware. For example, there are a number of theoretical models for parallel computation in general use among algorithm designers, but only through empirical work can these models be adjusted to reflect more faithfully the models embodied in the parallel systems. Crucial systems parameters, for example, the relation of processing time to communications time and memory speed, interact with algorithm design parameters in ways that can best be explored empirically."

[192] Wilson originally coined this term in 1987 in a document he drafted while at Cornell. This article was later published here: Kenneth G. Wilson, "Grand Challenges to Computational Science," Future Generation Computer Systems 5, no. 171 (1989).
[193] Bromley. The Federal High Performance Computing Program.

Although the overall structure of the HPC owes much to DARPA's Strategic Computing Initiative -- and in fact many of the same people were involved in both programs -- the interdisciplinary collaboration that is the centerpiece of the HPC also reflects the technological requirements of supercomputing. (**see diagram pg. 284 for organizational pyramid**) As one 1990 report to the DoE noted, the massively parallel computing systems that provided the computing speeds necessary for grand challenge applications, "have made it impossible to uncouple science and engineering from computing. The successful partitioning of problems among nodes requires intimate knowledge of the science in the application. Conversely, an understanding of new computer architectures provokes the solution of problems otherwise considered intractable. This interaction is the essence of computational science."[194]

Where the HPC departed from the goals of SCI, was in its abandonment of artificial intelligence applications as the goal of the program. SCI program managers had projected that to achieve machine intelligence would require the development of supercomputers, most likely of parallel design, that were capable of teraflops performance. Thus the Strategic Computing Architectures Program had supported the development of many of the experimental machines that found their way into the Advanced Research Computing Facility at Argonne.[195] While teraflops were seen as one component to the overall goals of SCI, the technology found a much broader community of users outside the program. Indeed, as this chapter has argued, faster machines were also the holy grail of computer scientists and computational scientists.

---

[194] Grand Challenges, High Performance Computing, and Computational Science.

[195] One of the most successful of these was Thinking Machines' Connection Machine (CM). See chapter 5 Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993.

In the face of the Japanese challenge, the United States embarked on two responses, one specific and one more general. DARPA's Strategic Computing program was a direct assault on the technological challenges impeding artificial intelligence. The broader initiative, and the one I see as more significant, was the effort to increase researcher access to supercomputers. It was this initially uncoordinated program, carried out by a variety of federal agencies including the DoE, NSF, NASA, and DoD, that provided the economic, political, and intellectual context in which computational science could emerge. When the High-Performance Computing program was drafted, it jettisoned the artificial intelligence component of Strategic Computing, but revitalized the architecture program to include networks. The technological core of computational science would now be ever-faster computers connected by high-speed data networks capable of supporting large-scale collaborative research on Grand Challenge class problems.

Despite the strong support of the OSTP, federal agencies and Congress, HPC had almost no foothold at the White House. It seems that part of the reluctance on the part of the President was that HPC appeared to set industrial policy. Even with the combined efforts by the OSTP and the other the federal agencies to repackage HPC as a program to develop the infrastructure for networks and high-performance computing rather than an effort to improve the nation's economic competitiveness, the program found little purchase throughout 1990.[196]

---

[196] Ibid., p. 315.

In 1991, Senator Gore introduced another bill in support of HPC, but it failed to get out of committee.[197] Finally, due to the diligence of Alan Bromley, the President's Science Advisory, the White House agreed to include the high-performance computing initiative in its budget for fy 1992.[198] By this time, too, networks had gained enough attention that there was some pressure from the OSTP to include "Communications" in the HPC initiative, but the feeling among the various agencies was that this was better handled as a separate program. In order to avoid the appearance of setting industrial policy, HPC was labeled a Category B item (new projects) and was presented as a "generic enabling technology at a precompetitive stage."[199] That behind the scenes some confusion still remained as to the nature of the program is seen in the way that collaborative proposals were evaluated: industry first, universities second and other federal agencies third.[200] Although HPC still had to be squeezed through different Congressional committees, and the full five-year plan for the program was not endorsed, the OSTP plan was eventually adopted and on December 9, 1991 the High Performance Computing Act of 1991 (known as the "Gore Act") was signed into law.[201] What it lacked, however, was funding.

---

[197] AMS Program Managers' Meeting. Washington, D.C.: Argonne National Laboratory, 1990. AMS PI Meeting, box 2, MCS Archives, Argonne National Laboratory.
[198] Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993, p. 315.
[199] AMS Program Managers' Meeting.
[200] Ibid.
[201] Roland and Shiman, Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993, p. 316.

Conclusion

The following year, supporters of HPC believed they had much to celebrate; the chief Congressional sponsor of HPC, Al Gore, was now the Vice-President of the United States. Surely in this changed environment HPC could expect to see some money flow its way. In this, they were not mistaken. Under the leadership of Clinton and Gore, HPC was expanded to include Communications (networking) as a central component of the program. Reintroduced as an integrated program that would help create the Internet and put a computer in every classroom, the High Performance Computing and Communication (HPCC) Initiative was passed in 1993.

Funded at almost $5 billion over five-years, it should not be surprising with a program this large that it meant different things to different people. HPCC supported particular lines of technological development (parallel computers and high-speed networks) and endorsed a particular organization of labor to solve Grand Challenge problems. Collaboration was mandatory; researchers in academia and government laboratories were expected to work closely with their counterparts in industry to explore new computer technologies and then apply these to problems deemed important by the state. By supporting such work, the federal government absorbed much of the risks involved in developing new computer architectures while also speeding up the process by which experimental machines became available to scientists, engineers, and industrial users.

Beyond the technology, HPCC also made a significant statement about the professional identity of computational scientists and computer scientists. As to the former, the program officially recognized computational science as a distinct third methodology for doing science, and that it was an inherently collaborative and interdisciplinary practice. That computational science was crucial to issues like national security, economic competitiveness, and the U.S. leadership position in science and technology was now firmly established. Also no longer in doubt was the symbiotic relationship with computational science and supercomputing technology. Although supercomputers still made up only a small percentage of the overall computer market (sales and usage) it was clear that money needed to be pumped into this cutting-edge technology if advances in science and technology were to continue.

For computer scientists, the HPCC can be seen as a mixed blessing. On the one hand, the initiative did provide tremendous financial support for the discipline of computer science in terms of training students and in providing experimental equipment on which computer science research depended. In this, HPCC had absorbed the arguments made by computer scientists over the preceding two decades that theirs was an experimental science, like physics, and thus they needed experimental computers. Both supercomputers and networks seemed to be just what the doctor ordered; they provided the experimental environments in which computer science research could flourish and served as the technological base on which collaborative projects could be built. In addition, the potential contribution of computer scientists to scientific research was no longer questioned. They were now seen as playing a direct role in the development and progress of computational science and especially in the application of parallel supercomputers to Grand Challenge class problems.

In many ways, then, HPCC seemed the realization of the “hybrid zone” which William Miller, the second director of Argonne’s Applied Mathematics Division, had proposed in 1961. Although he had imagined that applied mathematicians would be the central figure in collaborative projects within the hybrid zone, his myopia can be seen as a product of the times -- in 1961 nobody knew exactly what kinds of expertise would be important in computing. As it turned out, applied mathematics became but one subfield of the broader discipline of computer science (and applied mathematicians will claim that they are *not* computer scientists). Although computer scientists and applied mathematicians succeeded during the 1960s in carving out a disciplinary space in which they could pursue their own research, I suggest that this came at a cost. By separating themselves, for the most part, from working on the scientific problems of others, it was difficult for the people that mattered (i.e. had money) to connect theoretical computer science research to concrete methods for getting work done on computers. As a result, the discipline languished in a period of funding irrelevancy.

In response to this, the computer science profession launched a concerted effort during the 1970s and early 1980s to change the way the discipline was perceived by other scientists and especially by federal funding agencies. The renewed emphasis on the experimental component of computer science succeeded in connecting their work to the bottom line: fund computer science experiments and real computational tools get developed. By the time of the Japanese announcement of the Fifth Generation Project in late 1981, computer scientists were poised to take advantage of the ensuing national response. The focus on supercomputers, and then on the emergence of computational science as a third methodology for doing science, produced an environment ripe for collaborative, interdisciplinary projects. High-performance computing and computational science are both, by necessity, social activities. In this, the technology and the science it enabled reflected their Big Science heritage. And it is this context which led to a reappraisal of the contributions that computer scientists could make to the development of the technology and by extension, to computational science itself. HPCC institutionalized the hybrid zone at the core of its activities. Solving Grand Challenge problems became the goal for computational and computer science, and very quickly, its benchmark of success. What, exactly, constituted a Grand Challenge class problem was open to interpretation; how it would be solved was not. Computer scientists and mathematicians found their hybrid area at the cutting edge of computing. The question I will address briefly in the conclusion, is at what cost?

11. The DENELCOR Heterogeneous Element Processor. This was the first computer acquired by the Experimental Computing Facility, Math and Computer Science Division, Argonne National Laboratory.

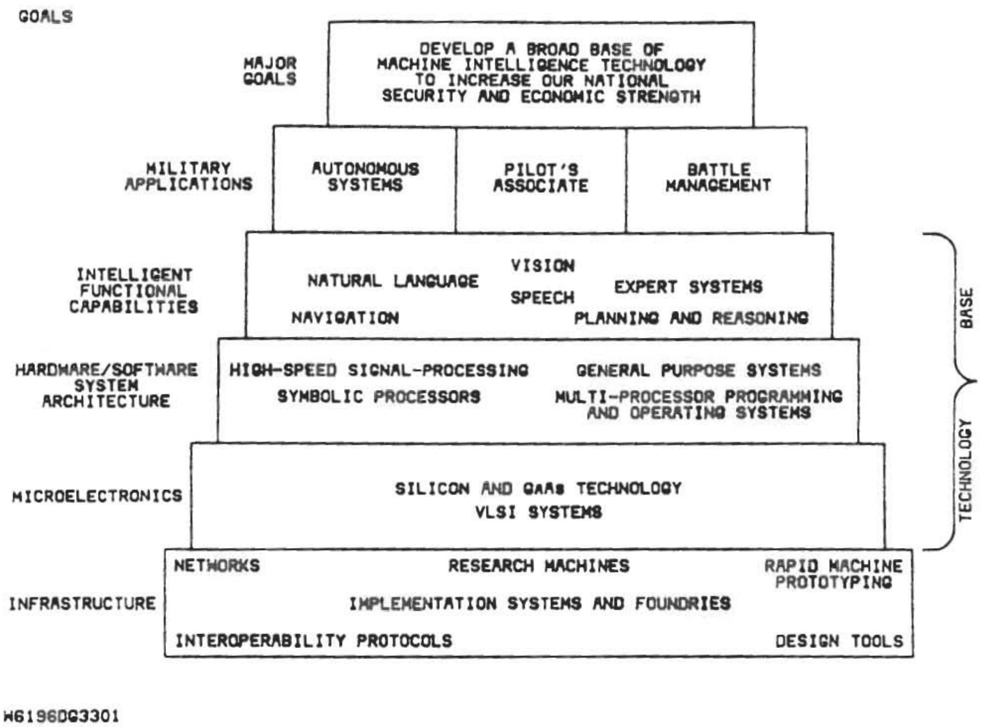


FIGURE 1. PROGRAM STRUCTURE AND GOALS

12. Strategic Computing Initiative Program Structure and Goals.

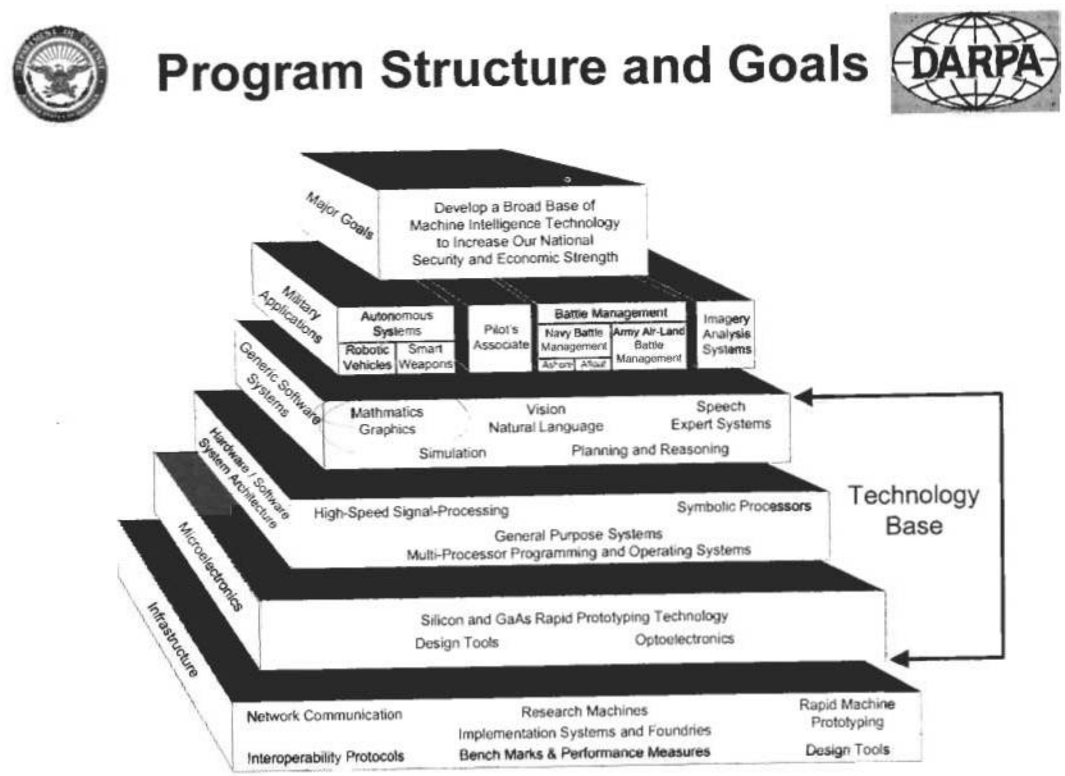


Figure 3.3
The fully articulated SC pyramid. Adapted from a viewgraph in Robert Kiggans's files.

13. The full articulated SC Pyramid. Reproduced from Roland, Strategic Computing.

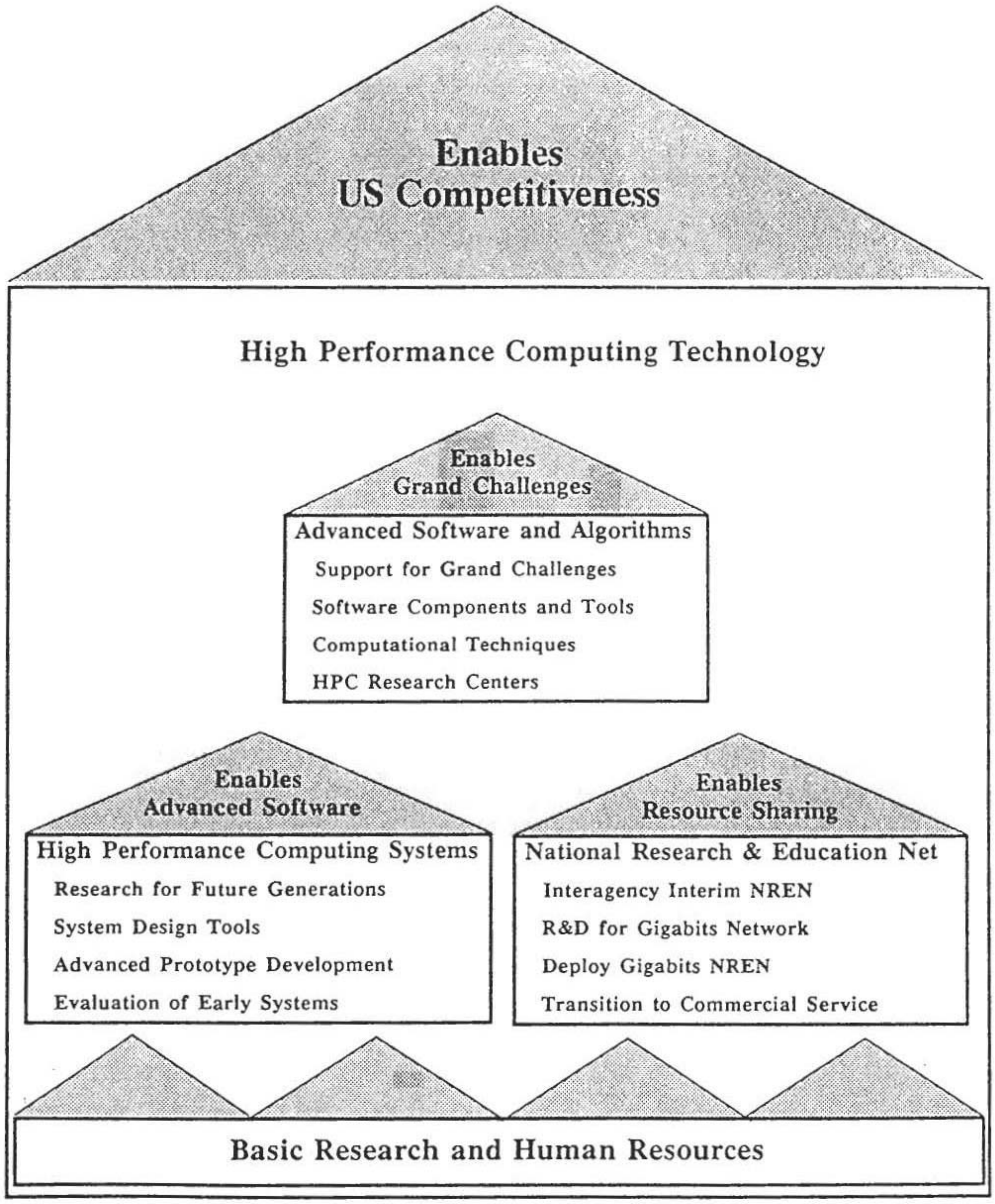


Fig. 1 – Relationship of HPC Program Components

14. The High-Performance Computing Program Components. Note that the HPC borrowed the pyramid structure of SCI, but the goal – Grand Challenges – were much less concrete that the goals of SCI.

Chapter 4

The Computational Science Machine

> "Reality check: We need to make ourselves aware of the new reality. Computational science we should not ignore. It is dependent on Numerical methods, new algorithms, and new architectures. For those that can package their work for consumption by this ever increasing area will win. Those that try to ignore it will lose."
> -- Hans Kaper, Director MCS Division, Argonne National Laboratory to MCS research staff, April 16, 1990.[1]

The passage of the High Performance Computing and Communications (HPCC) Act in 1993 was in many ways the culmination of the efforts by computer scientists to achieve disciplinary independence. In terms of funding, it represented by far the most amount of money ever given in support of computer science research. In addition, the language of the Act implied interdisciplinary collaboration among equals. Physicists, chemists, biologists, meteorologists, computational scientists and computer scientists would combine their respective expertise to solve fundamental problems in engineering and science. The goals, potential reward (actual and perceived), and rhetoric might

[1] Hans G. Kaper. Notes from Don Austin Visit. Argonne: Argonne National Laboratory, 1990. Federal Grand Challenges, box 5, MCS HPCC, MCS Archives, Argonne National Laboratory.

suggest something akin to the computational science equivalent of the moonshot. HPCC promised many things to many different groups. Yet clearly, the two groups who had the most to gain from the program, both professionally and monetarily, were computational scientists and computer scientists. After all, they had been the organizing force pushing for the program. In the mid-1980s, they had found common cause in their desire for access to high performance computers with advanced architectures. Together they made a convincing case that their research was critical to the security and economy of the United States, as well as to its scientific and technological future.

This last point was well understood by both computer and computational scientists. In late 1991, the ACM's Washington correspondent mused in her monthly column that "competition, a concept nary whispered in pure science circles, might just have been the catalyst that finally drove the High Performance Computing and Communications policy over the Hill."[2] She noted that years of government studies had failed to help legislators appreciate the potential of high-performance computing, "but toss in the competitive angle, and the story needs little translation." A study by the Gartner Group, a company that analyzes high-tech industries, highlights the importance that economic competition played in gaining support for HPCC. After surveying the goals of the program, the Gartner Group study indicated that if Congress passed the bill, it would likely create a return of nearly 140 times the original investment.[3]

If their political fortunes of computer and computational scientists rose due to the Japanese challenge of the early 1980s, so too, did their newfound disciplinary status. But while computer scientists achieved disciplinary independence, their computational

---

[2] Diane Crawford, "Supercomputing: From Here to Economy," Communications of the ACM 32, no. 9 (1989): p. 1.
[3] Ibid.: p. 26.

science counterparts far outstripped them in terms of prestige and power. In fact, by the beginning of the 1990s, computational science, both as a discipline and as a methodological mode of inquiry, had succeeded in shaping national priorities and science policy.[4]

HPCC was intended to jumpstart computational science and results from the program were not long in coming. Within a year of its passage, representatives from the High-Performance Computing and Communications Initiative reported on the program's accomplishments to the Science Subcommittee of the House Science, Space, and Technology Committee. HPCC had created over a dozen high performance computing research centers nationwide and extended the Internet to nearly three million computers world wide. In addition, program representatives boasted that "teams of researchers are using scalable systems to discover new knowledge and demonstrate new capabilities that were not possible with earlier technologies."[5]

Teamwork sells in America and the notion that teams of computer and computational scientists were "discovering new knowledge" sold well on Capital Hill. The rhetoric of teamwork is more than superficial. HPCC was crafted to produce teams of researchers who could tackle specific problems. At the same time, there is a dual nature to HPCC that is only partially captured by the team metaphor. "Team" applies most directly to the hardware, software, and people-ware that support the computational science *methodology.* But I also propose that computational science produced its own

---

[4] For the sake of clarity, because the 1993 High Performance Computing and Communications Act absorbed the entire program of the 1991 High Performance Computing Act with the addition of "communications" I will refer to the two Acts together as HPCC. My discussion applies equally to both programs.

[5] "High Performance Computing and Communications: Technology for the National Information Infrastructure: Supplement to the President's Fiscal Year 1995 Budget," (National Coordination Office, 1994). Available on-line at: http://www.hpcc.gov/congressional/testimony/supplement-10May94.html

ideology, and that it was this ideology that gained purchase in the halls of Congress and within federal funding agencies.

The Oxford English Dictionary defines "ideology" as "a set of beliefs characteristic of a social group or individual." Computational scientists successfully sold their methodology (beliefs) to sponsors as inherently interdisciplinary, collaborative, high-tech, and applied. They deployed the rhetoric of teamwork to suggest that all these elements would work together to achieve a desired goal. But computational science is also amenable to another metaphor: that of the "machine." Machines have multiple parts, but each one has a specific purpose and together they perform a particular task. Computational science and its attendant ideology appealed to politicians, science advisory boards, and program managers at federal funding agencies because it implied that technologies and people could be fine-tuned like a machine and directed towards solving problems that were important to the state. The computational science machine was oblivious to details like disciplinary boundaries and research agendas; what mattered was making sure the different parts worked together correctly in order to achieve the desired results. In this light, HPCC can be seen as creating a socio-technical machine tuned to the needs of the state. Together, the metaphors of "team" and "machine" are useful for understanding the social, political, and scientific appeal of computational science and, hence, why it achieved prominence so quickly.

At the same time, the metaphor of "team" has an entailment that "machine" does not, namely hierarchy. Team and "teamwork" suggest a particular organizational structure whereby individuals subsume their personal goals and instead play a specific role that will help the group achieve its goals. There is a reason why NFL quarterbacks are paid more than almost every other player on the team. Quarterbacks are the brains of the team and they touch the ball on every play in which they are on the field. Decisions made by quarterbacks in the heat of the moment can dramatically alter the fortunes of the entire team over an entire season. In most cases, quarterbacks are the *indispensable* member of the team and are rewarded for this by having more authority in team matters both on and off the field.

Despite the rhetoric of equality that infused HPCC, it was clear that some teammates were more equal than others. This point did not escape Rick Stevens, the Director of Argonne's MCS Division and the person responsible for implementing HPCC programs at the lab. "The fundamental challenge in the 'Grand Challenge'," he mused, "is their interdisciplinary nature."[6] Stevens' observation suggests that if we look behind the smooth face of collaboration that proponents of HPCC presented, we can see something more about the social organization of computational science. In particular, I argue that computational science, as methodology, ideology, and discipline significantly altered the directions of computer science research. My assertions are preliminary and are based on evidence from the Math and Computer Science Division at Argonne, which is a different environment from that of academic computer scientists. Nonetheless, I believe my conclusions are applicable to academia as well.

In general, after 1990, it became much more difficult for members of MCS to pursue computer science research that was inspired by issues within the discipline. Instead, the structure and goals of HPCC pushed them to reorient their research towards solving specific problems that originated outside of discipline. Specifically, computer

[6] Rick L. Stevens. MCS and the Washington Plan. Argonne: Argonne National Laboratory, 1990. MCS Files -- General, box 1, MCS Archives, Argonne National Laboratory.

scientists were pressured to assume duties they thought they had shed long ago, namely doing service work for other scientists. Despite the efforts by the profession to achieve disciplinary independence, and with it a concomitant ability to set the parameters of their research, under HPCC their position within interdisciplinary groups was the same in 1991 as it was in 1961. In contrast, computational scientists promised to solve important problems for their sponsors, and the close alignment of their agenda with national needs translated into power within collaborative projects. Computational scientists would be the quarterbacks directing research on Grand Challenge problems.

The Computational Science Hierarchy

While computer scientists were cast into a service role again, at least they possessed a valuable expertise when it came to making computers useful. This expertise, they discovered, could be mobilized in the crafting of proposals. By leveraging their knowledge, computer scientists were able to carve out a niche in which they could, within limits, pursue their own research.

In a sense, the entailment of hierarchy associated with the team concept was always visible to those who paid attention to the rhetoric of computational scientists. In the previous chapter I examined some of the characteristics of computational science as espoused by Kenneth Wilson. As HPCC began to coalesce after 1989, Wilson's ideas, and especially that of the Grand Challenge, became guiding principles for the entire program. The application-oriented nature of these problems should have been an indication to computer scientists that they would not call the shots. But hierarchy was more than a subtext in computational science; it was also made explicit by computational scientists as they strove to articulate the characteristics of their discipline to sponsors. In December, 1990 while the original HPC program was held up by the White House, the Scientific Computing Staff (SCS) of the DoE received a White Paper entitled "Grand Challenges, High Performance Computing, and Computational Science." The authors sought to establish reasons for why computational science should be considered an independent discipline. A large component of this project in self-fashioning was distinguishing computational science from computer science in terms of its goals, intellectual heritage, and methods of research.

Although both disciplines use supercomputers in their research, the White Paper asserts that the goals of the two intellectual activities are very different. "Computational science is applications driven: it deals with the intelligent use of computers to solve problems. In computer science, computers are the end. In computational science, computers are one of the tools: a means to the end."[7] By foregrounding the two related ideas that computational science was an applied discipline, and that the computer was its tool, the authors firmly situated the new discipline in opposition to computer science which had long struggled to demonstrate concrete applications. This position also subtly played on the stereotype that computer scientists were internally focused on machines and had little connection to the outside world.[8]

[7] Grand Challenges, High Performance Computing, and Computational Science. Washington: Scientific Computing Staff Office of Energy Research USDOE, 1990. box 5, MCS Archives.

[8] Computer science and computer scientists have long struggled with problems of image. To a large extent, the discipline has never been able to escape the stereotypical image of the hacker – a person more adept at dealing with machines than people, a-social with poor hygiene, and fond of techno-speak which no one but another computer scientist could understand. This image is well-established in popular culture, too, as evidenced by movies such as Desk Set (1957), War Games (1983), and Hackers (1995). Sociological studies of hackers have also reinforced this stereotype. See for example chapter 6, Sherry Turkle, The Second Self: Computers and the Human Spirit (New York: Simon & Schuster, Inc., 1984). and Gerald M. Weinberg, The Psychology of Computer Programming (New York: Von Nostrand Reinhold, 1971).

However, the disciplinary independence of computational science was based on more than its goals and chosen instrument. Ironically, computational scientist claimed to represent the new kind of "hybrid" researcher that William Miller had proposed in 1961. Back then, Miller believed that the most likely candidates to assume the hybrid mantle were the applied mathematicians. With one foot in applied mathematics, another in a scientific discipline, and a deep understanding of computers, he felt that applied mathematicians would bridge the gap between applications and computers. As I argued in Chapter 2, however, that applied mathematicians failed to achieve this position was due to external and internal tensions, most notably those arising from the emergence of computer science as a distinct discipline in the late 1960s and a corresponding disinclination to pursue collaborative projects. Now it was computational scientists who seized upon the idea of a hybrid researcher. "Its practitioners," the authors of the White Paper noted, "draw upon skills from scientific and engineering applications disciplines, computational mathematics, and computer science."[9] Moreover, the composite nature of the computational approach served not only as a bridge between applications and computers, but also between disciplines. Groups of computational scientists, they argued, drawn from different applications disciplines "will probably have as least as much in common with each other as any of the individuals do with their experimentalist and theoretician colleagues in the same applications discipline."[10]

One question the White Paper addressed directly was whether the use of supercomputers and simulations necessarily merited the creation of a new discipline. Certainly, the authors argued:

[9] Grand Challenges, High Performance Computing, and Computational Science.
[10] Ibid.

> “Computational science is necessary because, although it incorporates elements from applications disciplines, applied mathematics, and computer science, it does so in a way which results in a unique approach to science and engineering. Furthermore, as previously observed, computational science is not computer science. In computer science, computers are the end objective. In computational science, computers are a means to accomplish other scientific and engineering objectives. Computational science is also not simply applied mathematics. While drawing heavily on applied mathematics for methods and algorithms, computational science adds a strong linkage to observation and an intimate understanding of applications disciplines which is not found in applied mathematics.”[11]

Although the authors acknowledged that some mathematicians and a few computer scientists had made the transition to computational science “the requirement for intimate knowledge of an applications discipline constitutes a formidable barrier to entry by these groups. [Thus], their principal mode of contribution has been interdisciplinary teaming with applications experts.”

A second justification they offered for the creation of a distinct discipline stemmed from the reluctance of traditional applications disciplines to embrace the methodology of simulations. Because computational science cut across established scientific disciplines and also held the potential to create new application disciplines in the future, the authors argued that disciplinary independence was crucial while computational science was in its infancy because otherwise its practitioners would be marginalized within their traditional disciplines.

The White Paper suggests, too, that computational scientists had the right kind of temperament necessary to lead interdisciplinary projects, especially in those that included industrial partners. That computer scientists were not up to this challenge was emphasized in the report, and here, the similarities to the statements made about the mathematicians on Warren Weaver’s Applied Mathematics Panel during World War II are stunning:

[11] Ibid.

> "Because of the absence of programs to educate computational scientists, computer scientists had been hired by industry in their stead. These computer scientists have been largely ill-equipped for the engineering and scientific workplace (for which they were never trained in the first place) and have performed poorly enough that there are signs of a backlash against their further wholesale hiring."

If computer scientists and applied mathematicians might lack the requisite skills (and by extension, the personality) to be the lead investigators in Grand Challenge class projects, they were nonetheless significant to enabling computational science. The application of parallel computers to previously intractable problems required teams of applications experts, computational mathematicians, and computer scientists who could build the necessary tools. But as the authors made clear, "their work will be applications driven."

More subtle, but also more telling, was the assertion by the authors that computational science "should be developed from the scientist's perspective" and "computing environments should be built outwards from the scientist's desk." The underlying message is that the needs of computational scientists would dictate the kinds of projects that computer scientists would pursue. The computational science machine, while preaching collaboration, was without question hierarchical.

The DoE White Paper, "Grand Challenges, High Performance Computing, and Computational Science" is, I believe, a significant document. It goes beyond the simple refinement of arguments used to justify computational science; it also implies that the research activities of computer scientists and applied mathematicians needed to be redirected in order to suit the needs of computational scientists. Thus, it should not be surprising that the incorporation of this computational science ideology at the core of HPCC and within the directorates of federal funding agencies sparked fears among computer scientists that the power to set their own research agenda was in jeopardy. Indeed, trying to reconcile their desire for research autonomy with the juggernaut that was computational science became a key issue for members of the Math and Computer Science section at Argonne.

Computational Science and New Research Priorities

Signs that computational science promised to alter the activities of the MCS, and especially its Advanced Computing Research Facility, became apparent even in the earliest planning stages of HPCC. In late 1987, the Office of Science and Technology Policy (OSTP) released its report "A Research and Development Strategy for High Performance Computing" which formed the basis for their High Performance Computing program two years later. As has been noted, a major part of this strategy was based on the position of Nobel Laureate Kenneth Wilson that computational science offered a solution to certain "Grand Challenge" type problems. In direct response to this report, a few months later the Office of Energy Research of the Department of Energy initiated its own "grand challenge" (lower case "g" and "c") program and its Scientific Computing Staff (SCS) set aside a third of its annual budget beginning in late 1988 for these projects. This smaller grand challenge program was less ambitious and supported efforts to solve computational problems related to the DoE's Energy Research division, and financial support was based on three criteria. The first criterion ensured that the projects fit within the responsibilities of the DoE, while the last two emphasized immediate opportunities rather than the long-range projects that Wilson ascribed to the term Grand Challenge.[12] Although collaboration was not mandatory in the DoE grand challenges, in general the projects were conducted by teams of scientists, programmers, and computer scientists using supercomputers of parallel design.

That the SCS shifted one third of its budget to computational science projects as early as 1988 suggested to members of the MCS at Argonne that a broader reorientation of computer science funding was imminent. The SCS oversaw the DoE's Applied Mathematical Sciences (AMS) section which, in turn, was the primary supporter of all the activities of the MCS, including the operation of its Advance Computing Research Facility. Thus, changes in SCS priorities were considered a bellwether for what was to come.

When researchers' access to supercomputing cycles became easier in 1990, the SCS abandoned the small-scale grand challenges and adopted, *in toto*, the grandiose definition of "Grand Challenge" contained in HPC. Over the next two years, as HPC evolved into HPCC, the SCS also assumed responsibilities for directing and providing support for the Department of Energy's activities related to this program. HPCC aimed high and SCS directors wanted to ensure that any Grand Challenge projects it supported would be equally ambitious. As one SCS manager stated, "The DOE GCs [Grand Challenges] will be quite visible to both the Congress and to the Office of Science and Technology Policy and probably to the public as well: the DOE GCs must be able to succeed well in this bright light."[13]

---

[12] Tom Kitchens, to Hans G. Kaper. "Draft on Doe Grand Challenges for Discussion Nov. 18, 1992," November 14, 1991. AMS PI Meetings, box 2, MCS Archives, Argonne National Laboratory.
[13] Ibid.

The large sums of money involved in HPCC placed intense pressure on its managers to produce results. But the very nature of the problems to be addressed by the program -- those that were “previously intractable” -- meant that “results” might be very difficult to attain. For the SCS managers, it seemed their best chance for success lay in reorienting the entire computer science and mathematics research program to the goals of HPCC. That computer science would be subservient to computational science imperatives can be seen in the very criteria by which the new Grand Challenge proposals were evaluated. In priority order, they were: a project’s fundamental significance in terms of economic, social, and/or scientific impact; its contribution to international competitiveness; its applicability to the DoE mission; its ability to generate needed technologies; and finally, its interdisciplinary approach.

Unlike the first grand challenge program, where a proposal’s contribution to the mission of the DoE was the most important criteria, under HPCC, this had fallen to third, behind issues of fundamental significance and international competitiveness. In addition, under the former program, computer science and mathematics research activities were affected very little. These new guidelines, however, mandated that computer science research had to be application-oriented. The guidelines also institutionalized the hierarchy of disciplines that was implicit in the computational science methodology. For example, enabling technologies and interdisciplinary collaboration, the two areas in which computer scientists and applied mathematicians were supposed to make their biggest contributions, had fallen to fourth and fifth place, respectively.

HPCC and the MCS Advanced Research Computing Facility

The computational science methodology promised to link and coordinate disparate technologies, people, and disciplines into a scientific problem-solving machine. HPCC was intended to supply the framework in which this could happen. However, when the ACRF was established by Paul Messina in 1986, it was intended to do something similar, albeit on a much smaller scale. As I showed in the preceding chapter, the majority of computing cycles went to mathematicians and computer scientists who were testing out the experimental architectures, not to computational scientists who were tackling large-scale problems. By adhering to this approach, the ACRF had established itself as a leading center in the development of programming tools for parallel computers and as a training facility for researchers. However, when the computational science ideology began to gain a foothold at the DoE, the ACRF was pressured to use their parallel computers to do more "production" work at the expense of doing research on novel architectures. One suggestion for how to accomplish this was for the ACRF to acquire a very large supercomputer in tandem with Argonne's Computing and Telecommunications Division (CTD). The CTD had been created in 1983 when Paul Messina succeeded in separating his math and computer science research section from the computing service activities. Now there was pressure for the MCS to provide its expertise in parallel computers so that the CTD could run an Advance Computing Facility (ACF) for production work. For many MCS researchers, this was perceived as an assault on their hard won autonomy.

Facing certain changes in the operation, and possibly the mission of MCS, its directors sought a solution that would limit their impact on the Division's culture and research program. In all matters pertaining to high performance computing, Rick Stevens was the voice of MCS. He had joined the Division in 1985 and from 1987 to 1991 managed the Advanced Computing Research Facility. With degrees in physics, applied mathematics, and computer science, Stevens had distinguished himself as one of the world's foremost experts in large-scale scientific computing. In 1991, he served as associate director of MCS and then the following year became the Division's director.

In early 1990, with HPCC looming on the horizon and pressure mounting for the ACRF to transition to a production facility, Stevens explored several options that he hoped would secure the autonomy of the mathematicians and computer scientists. In particular, he wanted to preserve the research environment that had made the ACRF unique; this meant not working with CTD. In an internal note discussing potential strategies, Stevens wrote:

> "The ACF is a good solution but not a solution to "our" research needs. CTD ultimately will move into parallel computing when the time is right for the staff to take it on and I don't think we can nor should try to change that. We should be worrying about our own research programs and let them go forward at their own pace. It has profited us many times in the past to minimize our ties to production computing and I think this is another one of those times."[14]

Rather than join with CTD, Stevens wanted the ACRF to continue as a research facility where computer scientists could experiment with new hardware, develop software tools for that new hardware and then make them available to other researchers through high-speed networks. However, in order to remain viable in this new funding environment, Stevens realized that the ACRF would also have to produce technology demonstration projects of some kind in order to justify the money spent on the facility.

---

[14] Rick L. Stevens. ACRF Projects and Ideas. Argonne: Argonne National Laboratory, 1990. Federal Grand Challenges, box 5: MCS HPCC, MCS Archives, Argonne National Laboratory.

As the DoE began to embrace HPCC, the ACRF came under increased scrutiny from the Scientific Computing Staff, in part because of its high profile and past accomplishments. In 1990, funding for the ACRF constituted more than 20% of the entire budget of the DoE's Applied Mathematics Sciences section, and managers of that program made it clear that they were "unhappy with the ratio of the profile to results of the ACRF."[15] The DoE had pumped a lot of money into computer science and this in turn put pressure on managers to show results. It was the SCS's opinion that the ACRF was not working on the right kinds of problems -- "crack in the airplane versus noble prize" -- and the former made for weak PR.

Hans Kaper, who was MCS director in 1990, bore the brunt of these criticisms. After a meeting at DoE headquarters in April of that year, Kaper reported, "The Sandia effect has happened." Managers at the DoE saw themselves as "spending a whole lot of money at ANL for parallel computing and [they] are not winning accolades directly. True we may be teaching lots of people, but [they are] not getting the feeling that the work could not have been done someplace else. To be certain [they are] not advocating removing computing and replacing it with more mathematics or something less visible. [The] complaint is that the ACRF needs to be more visible (with results) and that one way to do that is by playing the biggest hardware game or the strangest hardware game."

The above passage suggests several ways in which the computational science ideology was beginning to reorient computer science research at Argonne. As the ideas of HPCC permeated senior DoE managers, they became fixated on supercomputers. As Kaper notes:

---

[15] Kaper. Notes from Don Austin Visit.

> "The psychology of the hardware game is really important. We have not been playing it for all it is worth. The simple fact is that people and resources are attracted to where the hardware is. Since it is relatively easy to buy the hardware compared to building up a team of people, most people underestimate the importance of the perception of hardware. The ACRF must make some significant move in the next six months or face a steady decline."

Solving problems on expensive computing equipment won accolades for program managers. Thus, the SCS looked to bring all its resources to bear on Grand Challenge class problems, and the ACRF was one of its most visible and important resources. As the needs of computational science began to drive computer science funding, programs like the ACRF had to adjust their mission to suit this new environment. "Reality check," Kaper wrote. "We need to make ourselves aware of the new reality. Computational science we should not ignore. It is dependent on Numerical methods, new algorithms, and new architectures. For those that can package their work for consumption by this ever increasing area will win. Those that try to ignore it will lose."

To a large extent, not ignoring computational science meant playing the hardware game by either acquiring or gaining access to the most powerful computers. Yet for a facility like the ACRF, which did not do "production" computing, building a case to acquire a state-of-the-art system would have a profound impact on the Division's research culture. As Stevens stated in a 1990 internal report:

> "The movement toward large configurations and the necessary compromise in the use of machines for 'production' will also involve closer working relationships with applications groups and the science programs will consume the vast majority of the cycles. The choice of pursuing a $20 million computer will alter the research programs in other ways, most likely projects will need to become aligned to this new resource whether it is appropriate or not simply to maintain credibility and demonstrate need."[16]

[16] ACRF Junta. Future ACRF Machine Considerations. Argonne: Argonne National Laboratory, 1990. Program Presentations to Cavallini and Johnson, Washington 11/7/90, box 2, MCS Archives, Argonne National Laboratory.

As a preliminary step in this direction, though, Kaper prodded members of the Division "to improve connections to applications groups" at Argonne with the goal of creating "four applications areas, all of the scope of the climate modeling project [in order to] provide adequate demand for such a facility."[17] Collaboration was a hallmark of HPCC; implementing it at the ACRF became vital to its existence. It was a difficult position to be in, no doubt. While connections to applications groups were necessary to justify the acquisition of a supercomputer, access to supercomputers were necessary to establish the connections:

> "The failure to accumulate the computing resources will reduce the quality of the visitors, students and the ability to attract and keep funding. To have a strong program we need to have large and powerful machines and a modern computing environment. We need to have top people too, but it is much easier to attract people to resources than resources to people."

If these connections could not be established, Kaper warned his staff that the MCS "could become like 'Brookhaven' a backwater of computing and mathematics as support will decline over the next five years."[18]

Stevens, especially, realized that the ACRF had to make its own moves towards supporting production type computational science as early as possible or face the prospect of having it imposed from above. Reluctant to give up research time on the computers already in the facility, Stevens instead put together the Concurrent SuperComputing Consortium (CSCC) in 1990. Consisting of twelve institutions, including CalTech, DARPA, NSF, NASA, and Purdue University, the CSCC installed the Intel Touchstone/Delta at CalTech, which was the world's fastest parallel supercomputer at the time. The creation of CSCC and the installation of the Intel machine was a coup.

---

[17] Kaper. Notes from Don Austin Visit.
[18] Ibid.

Rather than turn the ACRF itself into a production facility, it instead became the gateway through which scientists could access the supercomputer at CalTech. Stevens had been able to provide supercomputing cycles for production work to computational scientists at the lab without having to provide the support staff necessary to run such a computer.[19]

While the consortium was a huge success, it was still only a stop gap measure. The MCS still had to look for ways to align its research program with the goals of HPCC if it wanted to remain at the forefront of computer research. As Hans Kaper announced to the MCS staff after a DoE meeting in November, 1990:

> "The direction of the AMS (Applied Mathematics section) has changed. It is being driven by the HPC Program. Basic research in mathematics and computer science will be supported only if it contributes to the objectives of the high-performance computing initiative. The HPC initiative will not lead to an expansion of the core research program. It has its own short-term objective. If the MCS chooses not to align itself with the new direction, its budget will certainly decrease."[20]

It is ironic that the computer science community had for decades been pushing funding agencies to provide them with access to state-of-the-art computers. Yet when they finally got their wish it came at the cost to their independence. HPCC made high-performance computers available to computational and computer scientists, but it also imposed certain preconditions concerning what kind of work each would do. The difficulty faced by the computer scientists and mathematicians of the MCS was reconciling their desires for scientific autonomy with the realities of HPCC.

[19] Hans G. Kaper, to Joseph G. Asbury. "Issues for Discussion by the Mcs Division Review Committee," April 18, 1991. HPC White Paper, box 1, MCS Collection, Argonne National Laboratory. Stevens success in putting together the CSCC was no doubt helped by the fact that Paul Messina, after leaving the MCS, became the Director of the Supercomputing program at CalTech.
[20] Hans G. Kaper. Meeting with Scientific Computing Staff, Doe Headquarters, Germantown; Nov 7, 1990. Argonne: Argonne National Laboratory, 1990. Program Presentations to Cavallini and Johnson, Washington 11/7/90, box 2, MCS Archives, Argonne National Laboratory.

By 1991, the computational science ideology had profoundly reshaped the DoE's support of computer science research. This, in turn, generated discontent within the MCS Division. As Kaper reported to the MCS Review Committee:

> "Traditionally, the MCS Division's research program has been discipline driven. We concentrate on the development of methods, algorithms, and tools and operate principally within the applied mathematics and computer science community. New program managers in DoE's Applied Mathematical Sciences program, which funds most of our activities, have indicated that our program needs to become more applications oriented, maybe even applications driven. This change is causing strains in our relationship with DoE, which the committee may wish to analyze."[21]

But change was inevitable. As HPCC initiative moved closer to passage, John Cavallini, the director of the DoE's Scientific Computing Staff called a meeting in Washington for managers of advanced computer labs that received AMS money. The purpose of the meeting was to discuss the DoE's implementation and goals of HPCC. In contrast to the grandiose language of Grand Challenges, the long-term goals of the program seem somewhat prosaic. "The initiative will be applications oriented," Cavallini told his audience bluntly:

> "and will emphasize economic competitiveness and industrial partnership…. Its main objective is to enable transfer of advanced computing technology to U.S. industry. If, at the end of the 5-year period, we can demonstrate that the U.S. waste management industry has made the transition from PCs to massively parallel supercomputers, we have made our case."[22]

Although HPCC effectively pushed computer scientists back into doing service work for other disciplines, their expertise with computers did allow them to exert influence in the framing of Grand Challenge problems. If computational scientists held the most power and were the most visible members of the "team," at the same time

[21] Kaper, to "Issues for Discussion by the MCS Division Review Committee,"
[22] Hans G. Kaper, to MCS Staff. "Meeting at Doe Headquarters, Germantown. Thursday, January 17, 1991," January 17, 1991. HPCC Program Correspondence File 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

computational science was such a new discipline that few of its practitioners had the intimate knowledge of computers needed to develop quality proposals.

Stevens was highly attuned to the fact that the expertise wielded by computer scientists provided them some leverage. For example, when scientists from the Chemistry Division at Argonne wanted to develop a Grand Challenge proposal, they sought the advice of the MCS Division. Stevens, now Division director, was able to steer the proposal in such a way that it preserved the research programs of the Division as much as possible. Therefore, he counseled his colleagues in the Chemistry Division to make specific Grand Challenge proposals that would act as a proof of concept, were ambitious enough to "knock the socks of Cavallini," and would be endorsed by industry as an important problem. More importantly, Stevens suggested that the proposal should incorporate prototype software currently under development by MCS researchers, and that requests should be made both for new equipment for the MCS and for scads of supercomputer hours.[23]

Stevens' recommendations effectively appeal to the computational science ideology in terms of scale, ambition, and the technologies used to solve problems. At the same time, when crafted according to these guidelines, the Grand Challenge programs at Argonne allowed members of the MCS Division to continue some of their own research although it was modified to fit the needs of the lead discipline. There is strong evidence that scientists in other divisions seeking Grand Challenge funding routinely adopted MCS recommendations.[24]

---

[23] Al Wagner, to Rick L. Stevens. "Chemistry HPCC Component," June 3, 1991. HPCC Program Correspondence 1990-1991, box 5, MCS Archives, Argonne National Laboratory.
[24] Computational Materials Science. Argonne: Argonne National Laboratory, 1991. HPCC Program Correspondence File 1990-1991, box 5, MCS Archives, Argonne National Laboratory; Computational

Although the preceding analysis is somewhat preliminary and relies on sources drawn solely from Argonne, given the scale of HPCC is seems reasonable that computer science programs across the country were affected similarly. Scholars have demonstrated that it is important to pay attention to how issues such as funding profoundly influence the direction of scientific inquiry.[25] HPCC provides a similar opportunity. With this program, the computational scientist was still the poster-child for the Grand Challenge program, and building the tools to enable their work was the highest priority. That the computational science ideology permeated the highest levels of various funding agencies ensured that mathematicians and computer scientists would have to reconcile themselves to this new arrangement. At the same time, while HPCC placed strong pressure on computer scientists to reorient their work to the needs of the computational science team, as can be seen from Stevens' suggestions to the Chemistry Division, the expertise of computer scientists could be mobilized to carve out a middle ground between their service requirements and their own research interests. However, as HPCC became a full-blown initiative and industrial partners began to play a more prominent role in shaping research goals, it may be that even this autonomy became threatened.

Molecular Science. Argonne: Argonne National Laboratory, 1991. HPCC Program Correspondence File 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

[25] Robert Proctor, Cancer Wars: How Politics Shapes What We Know and Don't Know About Cancer (New York: Basic Books, 1995).

Summary and Conclusion

Computational Science in the History of Science

I have argued that computers are a unique scientific technology and that they have spurred the creation of entirely new scientific disciplines and new methodologies for scientific investigation. In exploring this question, I have attempted to situate the origins of computer and computational science concretely within human practices and to pay attention to the specific technological and social roots of these disciplines, as well as the characteristics of the computational science methodology. It is my contention that the emergence of computational science in the mid-1980s needs to be understood in conjunction with the efforts of computer scientists to establish their discipline *as a science*. In both computational and computer sciences, computers were the material basis on which practitioners from each group sought to build professional and scientific identities. To a certain extent, my work addresses a question asked by historian Theodore Porter: "How does one achieve intellectual authority in a society of strangers?"[1] Computer scientists struggled with this issue for over thirty years; computational

[1] Theodore M. Porter, Trust in Numbers: The Pursuit of Objectivity in Science and Public Life (Princeton: Princeton University Press, 1995), p. 221.

scientists for less than five. The stark difference in these two time spans is reflected in the amount of space devoted to each in this dissertation. Because computer science coalesced almost two decades before its computational counterpart, my story focuses heavily on the early effort of computer scientists to establish an independent disciplinary identity. By way of conclusion, I suggest that the radically different experiences of computer and computational scientists in establishing their disciplines were rooted in their ability (or lack thereof) to establish a research agenda. Whereas computer scientists still struggle to articulate their agenda, computational scientists aligned themselves closely with the needs of the state and built an agenda focused on solving problems of national import.

Computer science emerged in the early 1960s as a response to technical and mathematical difficulties inherent first in designing and then applying electronic digital computers to scientific problems. Thus, my story begins with the hardware and a discussion of the computer engineering efforts at Argonne that produced, among other experimental computers, the AVIDAC and ORACLE. Argonne’s computer engineers were some of the best, and when ORACLE was completed in 1953 it ranked as the fastest computer in the world. Scientists at Argonne were eager users of the machine, and the AVIDAC was quickly integrated into the scientific activities at the Laboratory.

While the AVIDAC lead to the centralization of scientific calculations at Argonne, it did not automatically lead to the consolidation of the people responsible for its programming and maintenance, as evidenced in mathematician Donald Flanders’ attempts to create a separate Division for the computing services. Flanders appealed to Cold War sentiments and especially to the widely held belief that the advancement of science was crucial to defeating Communism. Thus, the creation of the Applied Mathematics Division was presented as a way to improve scientists' efficiency.

At the same time, uniting scientists and applied mathematicians interested in computing within one division was a key step towards the creation of a shared disciplinary identity among practitioners. What form this discipline would take was uncertain, as were questions concerning how to integrate computer experts into interdisciplinary, collaborative research projects. William Miller, the second director of the AMD, offered a coherent vision for how to accomplish this and identified a "hybrid zone" that existed between scientific problems in the natural sciences and the calculating power of the computer. Within Miller's hybrid zone, applied mathematicians would become valued collaborators because of their expertise in crafting mathematical models of physical phenomena that were suitable for machine computation. I coined the term "translational junction" as a way to analyze Miller's idea because I think it captures more clearly the kind of interaction he envisioned for the hybrid zone. Quite directly, the scientific power and authority of applied mathematicians are based on their ability to translate physical or biological problems into a mathematical form that the computer could understand. Such work required intimate knowledge of both the application area and of computers, and thus Miller's hybrid zone applied mathematicians would themselves be hybrids. It was Miller's hope that interactions within the hybrid zone would provide guidance for the kinds of problems that needed to be solved by computer specialists and thereby lay the foundation for the formation of a distinct discipline.

Miller's use of the term "hybrid" reflected a general uncertainty within the computing community about what kinds of knowledge and activities were appropriate for a discipline focused on computers. Intense interest among practitioners during the 1960s culminated in the release of Curriculum ’68 by the Association of Computing Machinery (ACM). The ACM was the most important professional organization for computer scientists, and Curriculum ’68 was an attempt to standardize undergraduate training in computer science. Significantly, the report also sought to articulate those activities which did not belong within the discipline. Changes in the organizational structure of the AMD at Argonne reflected these comprehensive efforts to define computer science as a distinct discipline.

In both the effort to stake out the translational junction for computer experts and the eventual reorganization of the AMD which separated the consulting and research work from the blue-collar service activities, I suggest that computer scientists sought to establish their independence by controlling the social relations of work. That computer scientists and applied mathematicians possessed esoteric knowledge of computers was thought to provide them with the leverage needed to become critical participants in interdisciplinary research projects. However, their efforts were hindered on both socio-scientific and technological fronts. The social impediments can be further subdivided into issues external and internal to computer science. Externally, few scientists and engineers believed that computer science *was* a science; rather they saw it as a technical activity and thus were inclined to view its practitioners as performing a service instead of making an intellectual contribution to scientific research. Calls for disciplinary recognition by computer scientists were accompanied by a desire to shed these service duties, which, in turn, hardened the resolve of computer users not to financially support any activity that might hinder their own use of computers.

Internally, the discipline of computer science faced an identity crisis that continued for decades. As noted above, there was little consensus among computing practitioners about what activities and intellectual inquiries constituted their discipline. Historian Michael Mahoney interprets this debate in terms of the ability of computer scientists to set their own agenda.[2] The ACM's Curriculum 68' failed to settle this internal debate, but more significantly, it had little or no effect on how other scientific disciplines viewed computer science. Unable to reach a consensus on exactly what computer science was, its practitioners instead looked for ways to at least establish their field *as a science*. I argue that efforts during the 1970s to locate and emphasize an experimental tradition within computer science was a response to this uncertainty as well as a recognition that computer science would have to be more "applied" if it hoped to receive more funding.

The root of the issue, I proposed, was that producers and users of computational tools ascribed very different meanings to the computer. For computer scientists, they were the basis for a unique disciplinary identity; for those outside the field, computers were scientific instruments. Over time, the external and internal socio-scientific dynamics reinforced one another and effectively limited the ability of computer scientists to establish their intellectual authority among other scientists. While their expertise with computers was recognized, this did not lead to an elevation of the computer scientists in relation to other scientific disciplines. At the same time, the very technology upon which computer scientists tried to establish their discipline was also working against them. In the early days of electronic computing, mathematicians and nascent computer scientists

[2] Michael S. Mahoney, "Computer Science: The Search for Mathematical Theory," in Science in the Twentieth Century, ed. John Krige and Dominique Pestre (Amsterdam B.V.: Harwood Academic Publishers, 1997).

were akin to a priesthood because they understood these new engines of calculation. However, much of their research involved developing new tools and methods whereby computers became easier for non-specialists to use. In a sense, computer scientists were instantiating in software and hardware the very expertise they hoped would provide leverage *vís a vís* other scientific disciplines. As computer-using scientists and engineers learned how to do their own programming, they were able to satisfy their own computing needs. Overall, this produced two interrelated effects. First, it reduced a scientist's reliance on the computer priesthood to get work done. New programming languages, distributed computer terminals, and hundreds of already made software programs left scientists disinclined to pursue collaborative projects with computer scientists. Second, and more importantly, it undermined the ability of computer scientists to attract additional funding for their research.

This latter point needs some additional explanation. Computer science, by its very nature, is not interested in what *is* being done on computers. Instead, it is interested in what *can* be done on computers. Thus, its practitioners are always pushing the forefront of their science. This dynamic of computer science is reflected in the usage statistics of members of the AMD back in the 1960s. Whenever the AMD installed a new computer system, applied mathematicians and computer scientists were heavy users of the equipment as they sought to develop new tools and techniques to make the computer useful or more efficient. Over time, though, as software was developed for the computer and it transitioned into being primarily a production machine -- in other words, when the computer and software are transformed into what Latour calls a "black box" --

computer scientists and mathematicians had less to do with it.[3] The vitality of computer science research thus depended upon either getting a newer machine or pursuing more speculative research projects on the current machine. Both paths required money. Since new computer systems were few and far between, more often computer scientists pursued the latter path and pushed into research areas that had a much longer payoff (if any). In my discussion of the AMD, I pointed out that until the mid-1960s, the research activities of computer scientists and mathematicians at Argonne were paid for out of overhead charges to other scientific divisions at the Laboratory that used computers. For scientists who desired time on the computer, the overhead charge was perceived as subsidizing research that was not directly applicable to their own needs. Computer scientists and mathematicians were to a large extent powerless to affect this situation where their customers were generally satisfied with the computing tools they already had and were thus unwilling to support theoretical computer science research with little foreseeable payoff. It was in this way, I contend, that computing technology worked against the coalescence of a distinct disciplinary identity for computer scientists.

It would be interesting to investigate the latter point for what it might tell us about the willingness of the federal government to assume the risks entailed in creating new scientific disciplines. As discussed in Chapter 1, it is well known that the federal government was responsible for funding the development and creation of computers.[4] Initially, the government assumed this role because there were no commercial producers

---

[3] The black box was adopted by cyberneticists to denote a piece of complex machinery whose input and output to the machine are known. It is called a black box because the actual transformations that occur within the machinery, as well as the controversies that occurred in its design, are unknown and opaque. For the user of the machine, it is enough that a given input will produce a consistent and predictable output. See the Introduction, Bruno Latour, Science in Action (Cambridge: Harvard University Press, 1987).

[4] Kenneth Flamm, Targeting the Computer: Government Support and International Competition (Washington, D.C.: The Brookings Institution, 1987), Kenneth Flamm, Creating the Computer: Government, Industry, and High Technology (Washington, D.C.: The Brookings Institution, 1988).

of computers, and, given their extremely high cost and experimental nature, not likely to be any until the technology was proven. Even after a commercial computer industry emerged, the federal government supported the development of cutting-edge computers both by funding their construction and later by purchasing the equipment. In essence, the federal government assumed the risks entailed in bringing a new technology to market.

In contrast, it seems that the federal government was much less willing to support the computer science research necessary to make effective use of computer hardware. After all, it was only after John von Neumann's insistent prodding that the Atomic Energy Commission set aside any money at all for applied mathematical research. For the most part, it was assumed that the mathematical research that needed to be done would be supported, as noted above, by the overhead charges to computer users. My research makes clear, however, that scientists were resistant to subsidizing research that was not directly to their benefit, even if it might lead to the emergence of a new scientific discipline. Computer science, especially in the 1960s and 1970s, was perceived as an unwise investment. Theoretical research, such as Larry Wos' automated reasoning program at Argonne, for example, was considered risky it because it was unable to demonstrate immediate applications. Routine slashing of computer science research budgets by federal funding agencies from the mid-1960s through the mid-1970s supports my contention that at least in the early years, the government was much more willing to assume the financial risks involved in developing a new technology than in assuming the risks necessary to develop a new scientific discipline.

Together, the internal disputes among computer scientists, the external ambivalence (and hostility) of other scientists, rapid technological innovation, and a general unwillingness of funding agencies to support long-term theoretical research mitigated against computer science gaining recognition as an independent "scientific" discipline. It is quite telling that as late as 1979, the National Science Foundation (easily the most sympathetic federal funding source for computer science research) still did not categorize computer science as a distinct discipline. Although their professional and scientific aspirations had stalled, the products of computer science and applied mathematics research continued to enhance and extend the work of other scientists. In doing so, computer scientists laid the groundwork for the emergence of computational science in the 1980s.

At this point a second group that sought to construct a distinct scientific identity based on the technology of computing comes into focus. According to the rhetoric of computational scientists, their sudden emergence on the scientific scene was a by-product of vastly more powerful computer technologies which could support sophisticated simulations. More importantly, the use of computer simulations constituted a third methodology, alongside theory and experiment, for doing science. To accept this first claim at face value, however, is to accept a technologically deterministic account of computational science that ignores factors external to both technology and science. Thus, I stress the contingent nature of the emergence of computational science. As Paul Edwards has written, if the federal government had not supported the ENIAC project in World War II, it might have been decades before a commercial industry to produce digital electronic computers would have emerged.[5] Likewise, I contend that without the Japanese Fifth Generation project in the early 1980s, computational science would not

[5] See Chapter 2, Paul N. Edwards, The Closed World: Computers and the Politics of Discourse in Cold War America (Cambridge: The MIT Press, 1996).

have coalesced so rapidly nor would the federal High Performance Computing and Communications (HPCC) initiative have been funded at almost $5 billion within a decade.

But at the same time, I think that the rhetoric of computational science is important in understanding its social and cultural power. In this respect, computational scientists were little different than their predecessors in the Scientific Revolution. Steven Shapin has noted that during the sixteenth and seventeenth centuries, "many key figures…expressed *their* view that they were proposing some very new and very important changes in knowledge of natural reality and in the practices by which legitimate knowledge was to be secured, assessed, and communicated." Moreover, the claim by computational scientists that their methodology constituted something new can be compared to Shapin's revolutionaries who "identified *themselves* as 'moderns' set against 'ancient' modes of thought and practice."[6] But the truth is that computational science *was not new*. Numerical simulations, the basis for the computational methodology, had been used by scientists and engineers since the ENIAC first crunched numbers for the "Super" in 1946. Yet the ensuing decades of numerical simulations and technological advances did not spontaneously create independent, full-blown computational scientists.

Instead (as I argued in Chapters 3 and 4), computational scientists were given life by Fifth Generation, and then they actively engaged in a process of self-fashioning. Computational science was enabled by computer science, but computational scientists had grander ambitions. Thus, a strong component of their self-fashioning was to distance their discipline both from its computer science heritage and from connections with

[6] Steven Shapin, The Scientific Revolution (Chicago: University of Chicago Press, 1996), p. 5.

established disciplines like physics that might marginalize computational scientists. Their project was extremely successful; within a few years the belief that computational science was a new methodological approach and that computational scientists had unique needs and concerns formed the basis of the HPCC. Chapter 4 explores some of the implications that computational science had for computer science. In particular, I argue that as the "computational science ideology" found purchase in federal funding agencies, it reoriented computer science research towards its own ends. In a sense, the progeny became the master.

One of my goals in this project has been to explore the process by which disciplinary identities are established in science, and in this I think that the disparity in the fortunes of computer and computational scientists can shed light on this issue. The machine that is computational science came to dominate computer science, and computational scientists assumed the leadership positions within collaborative, interdisciplinary research projects. In both cases practitioners based their claims to disciplinary independence on the technology of computing. What I would like to do now is speculate a bit on why computational scientists have succeeded where computer scientist have failed.

I suggest that the inability of computer scientists to articulate a coherent agenda has been the main impediment to achieving disciplinary recognition. In part, this failure arises from the unique nature of computer science as a practice. Computers and programs are inherently mathematical; thus it might be easy to say it is a mathematical discipline. However, computer science is also interested in finding the most effective and efficient procedures for transforming information because quite literally, time on a computer is money. Finally, despite efforts to devise the best methodology for programming, to a large extent computer science is still a craft practice. Because it blurs the distinctions between science, engineering and craft practice, computer scientists have had trouble agreeing on what ought to be done, its order of importance, how it should be solved, or even what constitutes a solution.[7]

Seen in this light, computer science might constitute what Theodore Porter refers to as a "weak" scientific community because it lacks "widely shared assumptions and meanings."[8] In the sciences, weak communities struggle until their practitioners are able to establish within the discipline those elements that are recognized externally as scientific. For computer science, its mathematical basis seemed to offer an objective basis for the construction of new knowledge, but the strong presence of both craft and engineering practices militated against its acceptance as a science. This weakness, I argue, persisted until computer science was given a boost by the Fifth Generation. At this point, the expertise of computer scientists was seen as crucial to fulfilling the larger goals of the state. National needs produced increased funding for computer science research and training and this, in turn, amounted to a state-sanctioned endorsement of computer science as an independent discipline.

In contrast, computational scientists came from established disciplines like chemistry, physics, and meteorology. Consequently, they did not have to face issues of scientific credibility because their authority was already established. Ken Wilson brought to the table more than a Nobel Prize; he was also a physicist and thus possessed a cultural and scientific authority which eluded computer scientists. Thus, the arguments of

---

[7] Mahoney, "Computer Science: The Search for Mathematical Theory."
[8] Porter, <u>Trust in Numbers: The Pursuit of Objectivity in Science and Public Life</u>, p. 228.

computational scientists found a receptive audience within funding agencies in part because these agencies had long histories of dealing with the natural sciences.

While I believe that the above assertions shed some light on the vastly different experiences of computer and computational scientists, I want to speculate a bit more in an effort to link the establishment of these two disciplinary identities more firmly in computer technology. As I discussed in Chapter 3, computer and computational scientists found common cause in the 1980s in their desire for supercomputers. However, when that equipment was provided -- either through initiatives within individual government agencies or coordinated federal programs -- it empowered computational science much more than computer science. I suggest that the primary reason for this is that the two groups had widely different ideas for how to use the *experimental space* created by computers.

Computers, I propose, constitute an experimental space -- they manipulate numbers and symbols, transform information, and create numerical simulations. Steven Shapin and Simon Schaffer drew attention to the importance of experimental spaces in creating new knowledge in their book Leviathan and the Air-Pump. Whether it was the evacuated chamber in Robert Boyle's air-pump or the nascent laboratory of the Royal Society, the experimental space was seen as a place where new knowledge was created.[9] That computers constituted an experimental space was not lost on computer scientists. Remember, in 1967 Allen Newell, Alan Perlis, and Herbert A. Simon held forth in the pages of Science that the computer was not just an instrument but a phenomenon as well and thus required description and explanation. At its core, then, computer science was

[9] See Chapter 8, Steven Shapin and Simon Schaffer, Leviathan and the Air-Pump: Hobbes, Boyle, and the Experimental Life (Princeton: Princeton University Press, 1985).

interested in computers as an *object* of study. Algorithm development, automated reasoning, program transformations -- these were all research activities that took place within the experimental space. However, despite the best efforts to emphasize the value of their experiments within this space, the discipline did a poor job of explaining how these experiments met the needs of their sponsors.

Computational science, again, had none of these problems; the experimental space provided by supercomputers would be used to solve Grand Challenges -- those fundamental problems in science and engineering whose solution would have broad economic and scientific impact. Where computer science was inward looking, computational science was applied to solving problems of national importance. If computer science could only produce a fuzzy agenda, computational science aligned its agenda with the needs of its sponsors. Give some wood to computer scientists and they will test its strength, check it for knots, and measure its length. Do the same with computational scientists and they will build you a cabinet. As Shapin and Schaffer note, "scientific activity, the scientist's role, and the scientific community have always been dependent; they exist, are valued, and supported insofar as the state or its various agencies see point in them."[10] I suggest that the rapidity with which computational science became established as a discipline had much to do with its clear focus on applications that the state valued.

As a final thought for why computational science and its practitioners achieved disciplinary independence so readily, I want to refocus on the methodology they claimed made them distinct. In Chapter 3, I asserted that computational science was the methodological extension of big science: it was high-tech, collaborative, interdisciplinary

[10] Ibid., p. 339.

and very, very expensive. Although computer science shared these characteristics (although it was less interdisciplinary as it matured), it was never entirely clear to federal sponsors exactly what they were going to gain by supporting it. Thus, for decades the discipline limped along, saw its funding reduced at times, and was constantly forced to justify its existence.

Big science had helped to create ever more advanced computer technologies which had then been used to improve the nation's security, economic competitiveness, and its leadership in science and technology. However, much of the computerization of society had occurred in fits and spurts, and to a large extent the development of the technology was disconnected from its eventual applications. I think that some of the blame (perhaps not consciously) was attributed to computer scientists and their focus on the machine-as-object-of-study. In contrast, to computational scientists the computer was the *tool* and they put the potential contributions to the state of using this tool front and center. Computational science, however, was not just about applying computer technology -- it was a *methodology* unto itself. Moreover, it was a methodology that was enabled by the kinds of large-scale, interdisciplinary collaboration which characterized big science. As a methodology, it was appealing to state sponsors because it promised to link and coordinate disparate elements -- people, technologies, disciplines -- into a scientific problem-solving machine. The structure of HPCC was a conscious attempt to connect technological development to applications and federally-supported research to industrial needs. That computational science implied a hierarchy in the social organization of scientists was perceived in terms of improving the efficiency of the technological system.[11] Computer science promised to improve the efficiency of computers; computational science and its methods promised to improve the efficiency and productivity of *science*.

I think that each of these preceding points is worthy of further investigation. It would be especially interesting to trace the computational science Diaspora as it has spread rapidly throughout every scientific discipline. A Google search for "computational science" produces over 750,000 hits and a quick survey of the listings reveal that computational science has its own fellowships, institutes, conferences, training requirements, and history. I have tried to lay out the characteristics of the discipline and methodology, but what are needed now are case studies for how it has changed the internal dynamics of other scientific disciplines.

Equally important, I think it is necessary to explore how computational science has affected scientific priorities in a broader sense. The sheer scale and expense of doing cutting-edge research in computational science has profoundly restructured the relationships between federal research programs and private industry. It is not coincidence that Argonne National Laboratory dates it industrial affiliate program to 1990; this was the same year that the Lab's Advanced Computing Research Facility entered into partnerships with several large companies (most notably Intel) to support their computational science research. The HPCC Grand Challenge program made industrial participation mandatory and Argonne readily complied. The 1992 report from Argonne's High-Performance Computing Research Center lists twenty-one industrial partners, including Boeing, Bristol-Meyers, DuPont, Exxon, Ford, General Motors,

[11] Here I use Thomas Hughes' definition of technological system whereby these systems include not just the technology but also people and institutions. See Thomas P. Hughes, Networks of Power: Electrification in Western Society, 1880-1930 (Baltimore: Johns Hopkins University Press, 1983).

Merck & Company, Phillips Petroleum, and Searle.[12] On the one hand, the mix of companies -- from petrochemical to automotive to pharmaceutical -- reflects the diversity of scientific disciplines engaged in computational science. On the other hand, it raises questions about the power of industry to shape the course of scientific inquiry at its most fundamental level. In defense of HPCC, one consultant to several government agencies argued that the structure of the program did not "sacrifice science" but rather "[added] the competitive element." Introducing competition in science, he proposed "is one of the most positive things to come out of the government in this area."[13] A study that asked penetrating questions about the role of industry and competition in science would be a substantial contribution to the growing literature on agnatology, which is interested in the cultural production of ignorance. Increasingly, what we don't know in science and why may have much to do with the core agenda and methodological imperatives of computational science.

In less than six decades, computers have profoundly changed almost every facet of modern life. They have changed the way people interact, shop, conduct business, fight wars, and commit crimes. Computers have also profoundly changed science, its culture, its methodologies, and perhaps its goals. It is a technology that is constantly evolving and as a consequence computers have resisted technological "closure" and instead exist in a perpetual state of interpretive flexibility. Computers are weapons, musical instruments, photo shops, entertainment centers, and games. The Holy Grail of supercomputing in 1990 -- to achieve teraflop performance (a trillion floating point

---

[12] Argonne High-Performance Computing Research Center. Argonne National Laboratory, 1992. MCS Archives, Argonne National Laboratory.
[13] Diane Crawford, "Supercomputing: From Here to Economy," Communications of the ACM 32, no. 9 (1989): p. 26.

operations per second) –--is now standard in the X-Box 360, Microsoft's home computer gaming console which is to be released December, 2005. But as this dissertation has argued, computers are also the basis on which professional identities in science have been created. It is critical that scholars turn their attention to the implications of this technology.

Selected Bibliography

Government Reports

AEC 553/8 Establishment of a Computer Council to Advise the Division of Research. Washington, D.C.: Atomic Energy Commission, 1959, AEC 553/8. 12, 1334, 20, NARA.

AEC Computer Advisory Group. Five-Year Forecast of the Computer Research Program. Washington, D.C.: Atomic Energy Commission, 1961, AEC 553/14. 12, 326, 1334, NARA II.

ANL -5031 "Physics Progress Report," Physics Division Quarterly Report. 1953. Folder 2: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5080, "Physics Progress Report," Physics Division Quarterly Report. May 1953. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5140, "Physics Progress Report," Physics Division Quarterly Report. 1953. Folder 2: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5174 "Physics Progress Report, Feb. 1954," Physics Division Quarterly Report. Dec. 1952-Feb. 1953. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5237, "Physics Progress Report, Sept. 1954," Physics Division Quarterly Report. Sept. 1954. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5412 "Physics Progress Report, Sept. 1955," Physics Division Quarterly Report. 1955. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5420, "Physics Progress Report, May 1956," Physics Division Quarterly Report. May 1956. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5497, "Physics Progress Report, May 1956," Physics Division Quarterly Report. May 1956. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

ANL-5498, "Physics Progress Report, June 1956," Physics Division Quarterly Report. June 1956. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

Applied Mathematics Division. Argonne National Laboratory, 1964. folder 4, Box 2, COHC #196, Series 2: Supplemental Docs, Subseries C: Henry Tropp Files, Museum of American History Archives, Smithsonian Institute.

Applied Mathematics Division. Argonne National Laboratory, 1964. Folder 4: Press Releases, Box 2, Computer Oral History Collection #196, Series 2: Supplemental Docs, subseries C: Henry Tropp Files, American History Museum, Smithsonian Institute.

The Applied Mathematics Division in 1980. 1980. AMD File #4, 154, Records of the AUA, University of Illinois, Urbana-Champaign Special Collections.

Applied Mathematics Division Summary Report, Nov., 1956- June, 1958. AEC Research and Development Report, 1958, ANL-5954, Mathematics and Computers (TID-4500) 14th ed. MCS Files, ANL.

Applied Mathematics Division Summary Report, July 1, 1958 through June 30, 1959. ANL, 1959. AEC Research and Development Report, ANL-6090 Mathematics and Computers, (TID-4500), 15th Ed. MCS archives.

Applied Mathematics Division Summary Report. Argonne: Argonne National Laboratory, 1960, ANL-6195. MCS archives.

Applied Mathematics Division Summary Report, July 1, 1960 through June 30, 1961. AEC Research and Development Report, 1961, ANL-6453 Mathematics and Computers, (TID-4500), 16th Ed. MCS archives.

Applied Mathematics Division Summary Report. Argonne: Argonne National Laboratory, 1962, ANL-6641. MCS archives.

Applied Mathematics Division Summary Report. Argonne: Argonne National Laboratory, 1964, ANL-6952. MCS archives.

Applied Mathematics Division Summary Report, July 1, 1965 through June 30, 1966. AEC Research and Development Report, 1966, ANL-7280. MCS Archives.

Applied Mathematics Division Summary Report, July 1, 1966 through June 30, 1967. AEC Research and Development Report, 1967, ANL-7418 Mathematics and Computers (TID-4500). MCS Archives.

Applied Mathematics Division Summary Report, July 1, 1967 - June 30, 1968. Argonne: Argonne National Laboratory, 1968, ANL-7521 Mathematics and Computers. MCS Archives.

Argonne National Laboratory Applied Mathematics Division: A Synopsis of Long Range Plans. Argonne National Laboratory, 1975. Folder 2, Box 53, AMD Reports 1965-1975, Records of the Argonne University Association, University of Illinois, Urbana-Champaign Special Collections.

Argonne National Laboratory Preliminary Program Budget FY 1963 through FY 1967. Argonne National Laboratory, 1962. Director's Subject Files, B93-00125, Argonne National Laboratory Archives.

Argonne High-Performance Computing Research Center. Argonne National Laboratory, 1992. MCS Archives, Argonne National Laboratory.

Committee on ANL Computer Needs to Laboratory Director's Office. Recommendations for Future Computer Needs at ANL. Feb. 20, 1969. Computer Needs Committee Jan. 1969-Dec. 1969. Box 7: Computer Information, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Computational Materials Science. Argonne: Argonne National Laboratory, 1991. HPCC Program Correspondence File 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

Computational Molecular Science. Argonne: Argonne National Laboratory, 1991. HPCC Program Correspondence File 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

Grand Challenges, High Performance Computing, and Computational Science. Washington: Scientific Computing Staff Office of Energy Research USDOE, 1990. box 5, MCS Archives.

High Performance Computing and Communications: Technology for the National Information Infrastructure: Supplement to the President's Fiscal Year 1995 Budget. National Coordination Office, 1994.

Meeting of the Mathematics and Computer Sciences Research Advisory Committee with the GAC. Washington, D.C.: Atomic Energy Commission, 1964, AEC 553/24. 14, 1334, 20, NARA.

Newsletter Concerning Recent Developments in Various Digital Computer Projects. Office of Naval Research, Physical Sciences Division, 1956. Folder 21, Box 1, RG 196: COHC #196, Series 2: Supplemental Docs., Subseries C: Henry Tropp files, American History Museum, Smithsonian Institution Archive.

Office of Energy Research Scientific Computing Staff. Summaries of the FY 1989 Applied Mathematical Sciences Research Program. Washington, D.C.: U.S. Department of Energy, 1989. MCS-HPC "Federal HPC Program", Box 5, Records of the MCS, Argonne National Laboratory.

Office of Science and Technology Policy. "The Federal High Performance Computing Program." Washington, D.C.: Executive Office of the President, 1989.

Physics Division Quarterly Report. Sept. 1955. Folder: AVIDAC, Box 2: "Argonne Early Machines, Physics Division Reports", COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, American History Museum Archives, Smithsonian Institute.

Report of the Review Committee Applied Mathematics Division. ANL, 1962. ANL Review Committee for AMD, 1958-1980, B93-00147, MCS Archives.

Report to the Federal Coordinating Council for Science, Engineering, and Technology Supercomputer Panel on Recommended Government Actions to Retain U.S. Leadership in Supercomputers. Washington: U.S. Department of Energy Report, 1983.

Semiannual Report of Accomplishments in the Use and Management of ADP. Argonne National Laboratory, 1968. Computer Needs Committee, Jan 1970, Box 17: Computer Information, RG 326: Lab Director's Project Files, 1955-1970, GLFRCNARA.

Summaries of the FY 1989 Applied Mathematical Sciences Research Program. Washington: Department of Energy, 1989, DOE/ER-0422 UC-32. Federal HPC Program, box 5, MCS Archives.

Primary and Secondary Sources

Abbate, Janet. Inventing the Internet. Cambridge: the MIT Press, 1999.

Abstract of Portion of Minutes of AUA Executive Committee Meeting on March 30, 1970, Argonne University Association Archives, 1945-82, University of Illinois Special Collections.

ACRF Industrial and Academic Affiliates. Argonne: Argonne National Laboratory, 1990. Meetings with Don Austin, box 2, MCS Archives, Argonne National Laboratory.

ACRF Junta. Future ACRF Machine Considerations. Argonne: Argonne National Laboratory, 1990. Program Presentations to Cavallini and Johnson, Washington 11/7/90, box 2, MCS Archives, Argonne National Laboratory.

Ad Hoc Computer Study Group. Draft Description of a National Institute for Computer Science. AUA/Argonne National Laboratory, 1968. Personal Papers of Dr. Wayne Cowell.

Akera, Atsushi. Calculating a Natural World: Scientists, Engineers, and Computers in the United States, 1937-1968. Dissertation, University of Pennsylvania, 1998.

________. "Voluntarism and the Fruits of Collaboration: The IBM User Group, Share." Technology and Culture vol. 42, no. 4 (2001): 710-36.

Allardice, Corbin, and Edward R. Trapnell. The First Reactor. Washington, D.C.: United State Department of Energy, 1982.

AMS Program Managers' Meeting. Washington, D.C.: Argonne National Laboratory, 1990. AMS PI Meeting, box 2, MCS Archives, Argonne National Laboratory.

Anderson, Herbert L. "Metropolis, Monte Carlo, and the Maniac." Los Alamos Science no. 14, Fall (1986): 96-106.

ANL Public Relations. Press Release: Objectives of Argonne's Applied Mathematics Division, 1963. Argonne National Laboratory, RG 128 Records of the JCAE, 1946-77: General Correspondence, vol. 1-4, 1956-69., NARA.

"Scientists Complete and Operate World's Fastest Electronic Brain." The Argonne News, Sept. 2 1953, 1-2.

"Figures Are Their Forte." The Argonne News, Feb. 1958, 7-9.

"Introducting Gus -- a Multi-Computer System." The Argonne News, Feb. 1964, 6-9.

"Exit George: Old Computers Never Die. They Just Fade into History." The Argonne News, May 1966, 6-8.

"ANL's Computing Capacity Doubled." The Argonne News, Sept. 1967, 14-15.

Aspray, William. From Mathematical Constructivity to Computer Science: Alan Turing, John Von Neumann, and the Origins of Computer Science in Mathematical Logic. Ph.D Diss, University of Wisconsin, 1980.

________. "The Emergence of Princeton as a World Center for Mathematical Research, 1896-1939." In History and Philosophy of Modern Mathematics, ed. William Aspray and Philip Kitcher, XI, 346-66. Minneapolis: University of Minnesota Press, 1988.

________. "The Transformation of Numerical Analysis by the Computer: An Example from the Work of John Von Neumann." In The History of Modern Mathematics: Institutions and Applications, ed. David E. Rowe and John McCleary, 2, 307-25. Boston: Academic Press, 1989.

________. John Von Neumann and the Origins of Modern Computing History of Computing, ed. I. Bernard Cohen and William Aspray. Cambridge: The MIT Press, 1990.

________. "Arming American Scientists: Nsf and the Provision of Scientific Computing Facilities for Universities, 1950-1973." IEEE Annals of the History of Computing vol. 16, no. 4 (1994): 60-74.

Backus, John. "The History of Fortran I, II, and III." IEEE Annals of the History of Computing 20, no. 4 (1998): 68-78.

Baily, James. After Thought: The Computer Challenge to Human Intelligence. New York: Basic Books, 1996.

Bardini, Thierry. Bootstrapping: Douglas Engelbart, Coevolution, and the Origins of Personal Computing. Stanford: Stanford University Press, 2000.

Bardon, M, and K Curtis. A National Computing Environment for Academic Research. Washington: NSF Working Group on Computers for Research, 1983.

Barschall, H. H. to R. W. Harrison. Review Committee for Physics and Applied Mathematics Divisions. Nov. 10, 1960. B93-00147, ANL Review Committee for AMD, Argonne National Laboratory.

Bashe, Charles J., Lyle R. Johnson, John H. Palmer, and Emerson W. Pugh. Ibm's Early Computers. Cambridge: The MIT Press, 1986.

Baum, Claude. The System Builders: The Story of SDC. Santa Monica: System Development Corporation, 1981.

Bemer, Robert W. "Computing Prior to Fortran." IEEE Annals of the History of Computing 6, no. 1, January (1984): 16-18.

Bethe, Hans A. "Introduction." In Computers and Their Role in the Physical Sciences, ed. S. Fernbach and A. Taub, 1-9. New York: Gordon and Breach, Science Publishers, Inc, 1970.

Biagioli, Mario. Galileo, Courtier: The Practice of Science in the Culture of Absolutism. Chicago: University of Chicago Press, 1993.

Bijker, Wiebe E., Thomas P. Hughes, and Trevor F. Pinch, eds. The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology. Cambridge: The MIT Press, 1987.

Bingaman, Jeff. "Supercomputer: A Congressional Perspective." In Frontiers of Supercomputing, ed. Nicolas Metropolis, D.H. Sharp, W.J. Worlton and K.R. Ames, 1-3. Berkeley: University of California Press, 1986.

Bobbitt, J. T. to W. H. Zinn. Summary of Significant Factors with Respect to the Electronic Computer Arrangements with ORNL. Oct. 4, 1950. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Bolter, Jay David. Turing's Man: Western Culture in the Computer Age. Chapel Hill, NC: University of North Carolina Press, 1984.

Bolton Jr., William W. to Kenneth A. Dunbar. January 14, 1971. Personal Papers of Dr. Wayne Cowell.

Brandin, David H. "The Challenge of the Fifth Generation." Communications of the ACM 25, no. 8 (1982): 509-10.

Brill, Thomas. to P.A. Dana. April 22, 1949. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Bromley, D. Allan. The Federal High Performance Computing Program. Washington, D.C.: Office of Science and Technology Policy Executive Office of the President, 1987. HPCC Program Correspondence Files 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

Brooks, Frederick P. The Mythical Man-Month: Essays on Software Engineering. New York: Addison-Wesley Publishing Company, 1995.

Browder, Felix E. "Mathematics and the Sciences." In History and Philosophy of Modern Mathematics, ed. William Aspray and Philip Kitcher, XI, 278-92. Minneapolis: University of Minnesota Press, 1988.

Burdette, Earl W., and Rudolph J. Klein. "A Decade of Oracle Experience." In National Computer Conference, Pioneer Day, ed. Margaret Butler, 3-14. Chicago, IL: American Federation of Information Processing Societies, July 17, 1985.

Burks, Arthur. "Introduction." In Papers of John Von Neumann on Computing and Computer Theory, ed. William Aspray and Arthur Burks, 12, 3-16. Cambridge: The MIT Press, 1987.

Butler, Margaret. "Argonne National Laboratory Computing." pp. 3-4. Personal Papers of Margaret Butler.

________. to J. R. Huffman. Oct. 18, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to Robert Johnson. UNIVAC Time Records. July 6, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to B. Spinrad. Reactor Programs for the IBM 704. January 24, 1957. Laboratory Director's Reading File, ANL, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Campbell-Kelly, Martin. "Punched-Card Machinery." In Computing before Computers, ed. William Aspray, 122-155. Ames: Iowa State University Press, 1990.

Campbell-Kelly, Martin, and William Aspray. Computer: A History of the Information Machine The Sloan Technology Series. New York: Basic Books, 1996.

Cannon, William B. to Robert Duffield. Report of Review Committee Applied Mathematics Division. Jan. 14, 1972. ANL Review Committee for AMD, 1958-1980. B93-00147, MCS Archives.

Cavallini, John. to Hans G. Kaper. Notes from June 12th Meeting for Your Review/Comment. June 21, 1990. AMS PI Meetings. box 2, MCS Archives, Argonne National Laboratory.

Ceruzzi, Paul E. A History of Modern Computing. Cambridge: The MIT Press, 1999.

Chposky, James, and Ted Leonsis. Blue Magic: The People, Power, and Politics Behind the Ibm Personal Computer. New York: Facts on File, 1988.

Chu, J. C. to E. J. Doolittle. Oct. 26, 1951. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to James Mikulski. Jan. 4, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. "Computer Development at Argonne National Laboratory." In A History of Computing in the Twentieth Century, ed. Nicolas Metropolis, J. Howlett and Rota Gian-Carlo, 345-6. New York: Academic Press, Inc, 1980.

________, ed. Vol. ANL-5181, A Symposium on Large Scale Digital Computing Machines. Museum of Science and Industry, Chicago: Argonne National Laboratory, 1953.

Clippinger, R.F., B. Dimsdale, and J.H. Levin. "Automatic Digital Computers in Industrial Research." Society for Industrial and Applied Mathematics 1, no. 1 (1953): 1-15.

________. "Automatic Digital Computers in Industrial Research III." Society for Industrial and Applied Mathematics 2, no. 1 (1954): 36-56.

Cochrane, Rexmond C. Measures for Progress: A History of the National Bureau of Standards. Washington: U.S. Department of Commerce, 1966.

Cody, William J. "Observations on the Mathematical Software Effort." In Sources and Development of Mathematical Software, ed. Wayne R Cowell, 1-19. Englewood Cliffs: Prentice-Hall, Inc., 1984.

________. to E. Vanberkum. Major Accomplishments for MCS 1983-88. June 22, 1988. Major Accomplishments 1980-1994. Box 1, MCS Archives, Argonne National Laboratory.

________. Arithmetic Standards: The Long Road. Argonne National Laboratory, 1992. Personal Papers of Dr. Wayne Cowell.

Cohen, I. Bernard. Howard Aiken: Portrait of a Computer Pioneer. Cambridge, MA: MIT Press, 1999.

"Computer Science and Engineering Board Established at Academy of Sciences; Oettinger Named Chair." Communications of the ACM 11, no. 7 (1968): 530.

Computer Science and Telecommunications Board, National Research Council. "Computing the Future: A Broader Agenda for Computer Science and Engineering." National Academy Press, 1992.

Constant II, Edward W. The Origins of the Turbojet Revolution. Baltimore: Johns Hopkins University Press, 1980.

________. "The Social Locus of Technological Practice: Community, System, or Organization?'" In The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology, ed. Wiebe E. Bijker, Thomas P. Hughes and Trevor F. Pinch, 223-242. Cambridge: The MIT Press, 1987.

Cortada, James W. "Commercial Applications of the Digital Computer in American Corporations, 1945-1995." IEEE Annals of the History of Computing 18, no. 2 (1996): 18-29.

Courant, Richard. "Methods of Applied Mathematics." In Recent Advances in Science: Physics and Applied Mathematics, ed. Morris H. Shamos and George M. Murphy. New York: New York University Press, 1956.

Cowan, Ruth Schwarz. "The Consumption Junction: A Proposal for Research Strategies in the Sociology of Technology." In The Social Construction of Technological Systems: New Directions in the Sociology and History of Technology, ed. Wiebe E. Bijker, Thomas P. Hughes and Trevor F. Pinch, 261-80. Cambridge: The MIT Press, 1987.

Cowell, Wayne R. Minutes, Fifth Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal papers of Dr. Wayne Cowell.

________. Minutes, First Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal papers of Dr. Wayne Cowell.

________. Minutes, Second Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal papers of Dr. Wayne Cowell.

________. Minutes, Third Meeting Ad Hoc Computer Study Group. Argonne National Laboratory, 1968. Personal papers of Dr. Wayne Cowell.

________. to Glenn R. Ingram. August 8, 1969. Personal Papers of Dr. Wayne Cowell.

________. Draft Budget Summary, Institute for Computer Science. 1969. Personal Papers of Dr. Wayne Cowell.

________. Minutes First Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.

________. Minutes, Eighth Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.

________. Minutes, Eleventh Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.

________. Minutes, Fourteenth Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.

________. Minutes, Ninth Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.

________. Minutes, Sixth Meeting Technical Steering Committee of the Institute for Computer Science. Argonne National Laboratory, 1969. Personal Papers of Dr. Wayne Cowell.

________. A Proposal to the National Science Foundation for the Development of a Certified Subroutine Library. Argonne: Argonne National Laboratory, 1970. Personal Papers of Dr. Wayne Cowell.

________. to R.J. Royston. Information for Review Committee. October 17, 1977. folder 2. box 53, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

________. "Preface." In Sources and Development of Mathematical Software, ed. Wayne R Cowell, xi-xii. Englewood Cliffs: Prentice-Hall, Inc., 1984.

Crane, H.R. to L.R. Kimpton. Review Committee Report. June 1, 1958. Box 11, RG 326: Records of the Policy Advisory Board, 1957-1967, GLFRCNARA.

Crawford, Diane. "U.S. Computing Strategy One Year Later." Communications of the ACM 31, no. 10 (1988): 1171-73.

________. "Supercomputing: From Here to Economy." Communications of the ACM 32, no. 9 (1989): 1048-50.

Creview, Daniel. AI: The Tumultuous History of the Search for Artificial Intelligence. New York: Basic Books, 1993.

Crewe, Dr. A. V. to George W. Beadle. Jan. 15, 1963. Box 11, RG 326: Policy Advisory Board 1957-67, GLFRCNARA.

________. to George W. Beadle. Report on Review Committee for Applied Mathematics. April 16, 1964. Box 11: Review Committee for Applied Mathematics, RG 326: Policy Advisory Board, 1957-67, GLFRCNARA.

________. to Argonne Division Directors. Staff Announcement. Dec. 2, 1964. Vol. 3, 1964-65. General Correspondence, RG 128: Records of the Joint Committee on Atomic Energy, NARA II.

Curtiss, J. H. The Institute for Numerical Analysis of the National Bureau of Standards. Washington, D.C.: Office of Naval Research, Department of the Navy, 1951. box 3, COHC #196, Series 2, Subseries C, American History Museum, Smithsonian Institute.

________. "The New Significance of Computation in Higher Education." In The Computing Laboratory in the University, ed. Preston C. Hammer, 11-14. Madison: University of Wisconsin Press, 1957.

________. "The National Applied Mathematics Laboratories of the National Bureau of Standards: A Progress Report Covering the First Five Years of Its Existence." IEEE Annals of the History of Computing 11, no. 2 (1989): 69-98.

Dallaire, Gene. "American Universities Need Greater Access to Supercomputers." Communications of the ACM 27, no. 4 (1984): 292-98.

Dalmedico, Amy Dahan. "Mathematics in the Twentieth Century." In Science in the Twentieth Century, ed. John Krige and Dominique Pestre, 651-67. Amsterdam B.V.: Harwood Academic Publishers, 1997.

Daston, Lorraine J. "Fitting Numbers to the World: The Case of Probability Theory." In History and Philosophy of Modern Mathematics, ed. William Aspray and Philip Kitcher. Minneapolis: University of Minnesota Press, 1988.

Davis, D. A. to J. R. Foote.   Dec. 18, 1956.   Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Denning, Peter. "Paradigms Crossed." Communications of the ACM 30, no. 10 (1987): 808-09.

Denning, Peter, Douglas E. Comer, David Gries, Michael C. Mulder, Allen Tucker, A. Joe Turner, and Paul R. Young. "Computing as a Discipline." Communications of the ACM 32, no. 1 (1989): 9-23.

Denning, Peter J. "What Is Experimental Computer Science?" Communications of the ACM 23, no. 10 (1980): 543-44.

Denning, Peter J., Edward Feigenbaum, Paul Gilmore, Anthony Hearn, Robert Ritchie, and Joseph Traub. "The Snowbird Report: A Discipline in Crisis." Communications of the ACM 24, no. 6 (1981): 370-74.

Dietrich, J. R. to E. Bromberg.   April 6, 1956.   Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Duffield, Robert B. to Division Directors and Department Heads.   May 14, 1970.   Laboratory Director's Reading File, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

DuPree, A. Hunter. "The *Great Instauration* of 1940: The Organization of Scientific Research for War." In The Twentieth-Century Sciences, ed. Gerald Holton, 443-67. New York: W.W. Norton & Company Inc., 1972.

Eberlein, P.J. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1975. folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign Special Collections.

Eckert, J. P. Jr. "A Survey of Digital Computer Memory Systems." IEEE Annals of the History of Computing vol. 20, no. 4 (1998): 15-28.

Edwards, Paul N. The Closed World: Computers and the Politics of Discourse in Cold War America. Cambridge: The MIT Press, 1996.

Ensmenger, Nathan L. From "Black Art" to Industrial Discipline: The Software Crisis and the Management of Programmers. Ph.D. diss., University of Pennsylvania, 2001.

Estrin, G. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1976. folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign Special Collections.

Feigenbaum, Edward, and Pamela McCorduck. The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World. New York: Addison-Wesley Publishing, Inc., 1983.

Feldman, Jerome A., and William R. Sutherland. "Rejuvenating Experimental Computer Science: A Report to the National Science Foundation and Others." Communications of the ACM 22, no. 9 (1979): 497-502.

Fernbach, S., and A. Taub, eds. Computers and Their Role in the Physical Sciences. New York: Gordon and Breach, Science Publishers, Inc., 1970.

Fields, T.H. Computer Performance Evaluation. 1968. Laboratory Director's Reading File, RG 326 Argonne National Laboratory, Records of the Atomic Energy Commission, GLFRCNARA.

Flamm, Kenneth. Targeting the Computer: Government Support and International Competition. Washington, D.C.: The Brookings Institution, 1987.

________. Creating the Computer: Government, Industry, and High Technology. Washington, D.C.: The Brookings Institution, 1988.

Flanders, Donald A. to George Shortley. Nov. 10, 1950. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to Cuthbert C. Hurd. July 6, 1951. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to J.W. Givens. Nov. 29, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to Harvey Cohn. Feb. 10, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to T. Brill. Fast Output for George. Feb. 24, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to J. R. Foote. Procurement of a Digital Computer. Nov. 7, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to R. L. Chapman. Psychological Tests. Feb. 22, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to James Alexander. Report on Logical Structure of AVIDAC. Feb. 22, 1956. Laboratory Director's Reading File, 1949-1957, RG: 326 Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to N. Hilberry. Visits of Dr. Reimar H. Luest. June 13, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to J. R. Foote. Jan. 28, 1957. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Forsythe, George E. "The Educational Program in Numerical Analysis of the Department of Mathematics, U.C.L.A." In The Computing Laboratory in the University, ed. Preston C. Hammer, 145-52. Madison: University of Wisconsin Press, 1957.

Foster, Ian. MCS and the High Performance Computing Initiative. Argonne: Argonne National Laboratory, 1990. MCS Proposals Late Period, box 2, MCS Archives, Argonne National Laboratory.

Furney, L.C. "Lab Announces Completion of Electronic Brain." The Argonne News, Feb. 4 1953, 1-2, 10.

Gabriel, J.R. to R.L. Logan. The Maclaren Paper on Arc and Its Implementation Relating to Computer Acquisition Economics. Dec. 23, 1968. Computer Needs Committee Aug. 1968-Dec. 1968. Box 17: Computer Information, RG 326: Laboratory Director's Project Files 1955-1970, GLFRCNARA.

Galison, Peter. "Artificial Clouds, Real Particles." In The Uses of Experiment: Studies in the Natural Sciences, ed. David Gooding, Trevor Pinch and Simon Schaffer, pp. 225-76. Cambridge: Cambridge University Press, 1989.

________. Image and Logic: A Material Culture of Microphysics. Chicago: University of Chicago Press, 1997.

Galison, Peter, and Bruce Hevly, eds. Big Science: The Growth of Large-Scale Research. Stanford: Stanford University Press, 1992.

Galler, Bernard A. "Distinction of Computer Science." Communications of the ACM 17, no. 6 (1974): 300.

General Information Manual: Machine Information, 704's Installed. 1956. Applied Science, IBM Corporate Archives.

Gerhart, Susan L. Review Committee Report Mathematics and Computer Science Division. Argonne: Argonne National Laboratory, 1983. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.

Gilbert, T.L. to O.C. Simpson. Memo of 3/18/69 of R.M. Adams to Distribution from Computer Needs Committee to Laboratory Director on Recommendations Concerning Applied Programming Services. March 19, 1969. Computer Needs Committee Jan. 1969-Dec. 1969. Box 7: Computer Information, RG: 326 Records of the Atomic Energy Commission, Argonne National Laboratory, GLFRCNARA.

Givens, J.W. to J. O. Hirschfelder. Design of AMD Building. July 24, 1964. Records of the Policy Advisory Board, General. RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to W. M. Manning. Comments on the 1965 Report of the Review Committee of AMD. Sept. 21, 1965. Box 11: Review Committee AMD, RG 326: Records of the Policy Advisory Board, 1957-67, GLFRCNARA.

________. Argonne National Laboratory, 1968. box 2, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to Winston Manning. Comments on the 1967 Report of the Review Committee of the Applied Mathematics Division. March 13, 1968. folder 1. box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

________. to R.M Adams, R. Avery and T.H. Fields. Computer Use. Oct. 18, 1968. Computer Needs Committee Aug.- Dec. 1968. Box 17: Computer Information, RG 326: Laboratory Director's Project Files 1955-1970, GLFRCNARA.

________. to Richard Adams. Four Research Areas Proposed for Phase I of the Proposed Institute for Computer Science. Oct. 10, 1968. Computer Needs Committee Aug.-Dec.

1968. Box 17: Computer Information, RG: 326 Lab Director's Project Files 1955-1970, GLFRCNARA.

________. to W. M. Manning. Response to the Report of the Anl Review Committee for the Applied Mathematics Division. Feb. 13, 1969. Folder 1. Box 153: AMD Reports, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Glass, Robert L. Software Creativity. Englewood, NJ: Prentice Hall, Inc., 1994.

Goldstine, Herman H. The Computer from Pascal to Von Neumann. Princeton: Princeton University Press, 1972.

Gooding, David, Trevor Pinch, and Simon Schaffer, eds. The Uses of Experiment: Studies in the Natural Sciences. Cambridge: Cambridge University Press, 1989.

Gosling, F. G. The Manhattan Project: Making the Atomic Bomb. Oak Ridge, TN: The United States Department of Energy, 1999.

Grattan-Guinness, I. "Modes and Manners of Applied Mathematics: The Case of Mechanics." In The History of Modern Mathematics: Institutions and Applications, ed. David E. Rowe and John McCleary, 2, 109-26. Boston: Academic Press, 1989.

Greenbaum, Leonard. A Special Interest: The Atomic Energy Commission, Argonne National Laboratory, and the Midwestern Universities. Ann Arbor: The University of Michigan Press, 1971.

Gries, David, Terry Walker, and Paul Young. "The 1988 Snowbird Report: A Discipline Matures." Communications of the ACM 32, no. 3 (1989): 294-297.

Grosch, H.R. J. "The Computer Laboratory in Industry." In The Computing Laboratory in the University, ed. Preston C. Hammer, 87-90. Madison: University of Wisconsin Press, 1957.

________. "Changes in Washington." Communications of the ACM 19, no. 12 (1976): 653.

Haigh, Thomas. "The Chromium-Plated Tabulator: Institutionalizing an Electronic Revolution, 1954-1958." IEEE Annals of the History of Computing 42, no. 4 (2001): 75-104.

________. Technology, Information and Power: Managerial Technicians in Corporate America, 1917-2000. Ph.D. thesis, University of Pennsylvania, 2001.

Hall, Jean F. to L. R. Wallis. AVIDAC and Programming Time Report for April 26 through May 25, 1956. May 29, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to L. R. Wallis. AVIDAC and Programming Time Report for July 26 through August 23, 1956. August 24, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to L. R. Wallis. AVIDAC and Programming Time Report for November 26 through December 25, 1955. January 4, 1956. Laboratory Director's Reading File, ANL 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to L. R. Wallis. AVIDAC and Programming Time Report for the Period December 26, 1956, through January 25, 1957. January 29, 1957. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Harrell, W. B. to Nicolas Metropolis. Proposal for Review Committee. April 21, 1960. Policy Advisory Board, 1957-1967. 11, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to Kenneth A. Dunbar. New Building for AMD. May 1, 1962. ANL Construction Projects, 1961-64. Box B93-00150: ANL Administration/Appointments, MCS Archives, Argonne National Laboratory.

Haugeland, John. Artificial Intelligence: The Very Idea. Cambridge, MA: MIT Press, 1985.

Heilbron, J.L., and Robert W. Seidel. Lawrence and His Laboratory: A History of the Lawrence Berkeley Laboratory. Vol. I. Berkeley: University of California Press, 1989.

Hewlett, Richard G., and Jr. Oscar Anderson. The New World Order, 1939-1946, Volume I of a History of the United States Atomic Energy Commission. University Park: Pennsylvania State University Press, 1962.

Hilberry, N. to George W. Beadle. Report on Review Committee for the Applied Mathematics Division. July 17, 1961. Box 11, RG 326: Records of the Atomic Energy Commission, Policy Advisory Board 1957-1967, GLFRCNARA.

Hiltzik, Michael. Dealers of Lightning: Xerox PARC and the Dawn of the Computer Age. New York: HarperCollins, Inc., 1999.

Holl, Jack M. Argonne National Laboratory, 1946-96. Chicago: University of Illinois Press, 1997.

Hollingsworth, R.E. to Clinton P. Anderson. Jan. 21, 1960. 12. 1334, RG: 326 Records of the Atomic Energy Commission, # 20, NARA II.

Hounshell, David A. From the American System to Mass Production, 1800-1932: The Development of Manufacturing Technology in the United States Studies in Industry and Society; 4. Baltimore: Johns Hopkins University Press, 1984.

Hoyt, Frank C. to David M. Rubel. Oct. 18, 1949. Laboratory Director's Reading File ANL, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. to W.T. Martin. Feb. 24, 1950. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Hughes, Edwin L. "Williams Tubes: A Remembrance." In National Computer Conference, Pioneer Day, ed. Margaret Butler, 15-19. Chicago, IL: American Federation of Information Processing Societies, July 17, 1985.

Hughes, Thomas P. Networks of Power: Electrification in Western Society, 1880-1930. Baltimore: Johns Hopkins University Press, 1983.

Hughes, Thomas P., and Agatha C. Hughes, eds. Systems, Experts, and Computers: The Systems Approach in Management and Engineering, World War Ii and After, Dibner Institute Studies in the History of Science and Technology. Cambridge: The MIT Press, 2000.

Illingworth, Valerie, ed. Dictionary of Computing. New York: Oxford University Press, 1986.

Inman, B.R. "Supercomputer Leadership: A U.S. Priority." In Frontiers of Supercomputing, ed. Nicolas Metropolis, D.H. Sharp, W.J. Worlton and K.R. Ames, xvii-xxiv. Berkeley: University of California Press, 1986.

Interview with Wallace Givens. 1966. COHC #196, Series 2: Supplemental Docs; Subseries C: Henry Tropp Files, National Archives and Records Administration.

Johnson, Leland, and Daniel Schaffer. Oak Ridge National Laboratory: The First Fifty Years. Knoxville: The University of Tennessee Press, 1994.

Johnson, T. R. to Edward Teller. Dissolution of the AEC Computer Council. June 15, 1955. 11: Computer Council. 2229, von Neuman, J. AEC, DOE Archives, Germantown, MD.

Kaper, Hans G. Meeting with Scientific Computing Staff, Doe Headquarters, Germantown; Nov 7, 1990. Argonne: Argonne National Laboratory, 1990. Program Presentations to Cavallini and Johnson, Washington 11/7/90, box 2, MCS Archives, Argonne National Laboratory.

________. Notes from Don Austin Visit. Argonne: Argonne National Laboratory, 1990. Federal Grand Challenges, box 5 MCS HPCC, MCS Archives, Argonne National Laboratory.

________. to Joseph G. Asbury. Issues for Discussion by the MCS Division Review Committee. April 18, 1991. HPC White Paper. Box 1, MCS Collection, Argonne National Laboratory.

________. to MCS Staff. Meeting at Doe Headquarters, Germantown. Thursday, January 17, 1991. January 17, 1991. HPCC Program Correspondence File 1990-1991. box 5, MCS Archives, Argonne National Laboratory.

________. to J.G. Ashbury. Strategic Plans for High-Performance Computing. June 18, 1991. HPCC Program Correspondence File 1990-1991. Box 5, MCS Archives, Argonne National Laboratory.

Kaper, Hans G., and Rick L. Stevens. Argonne National Laboratory High-Performance Computing and Communications Program. Argonne: Argonne National Laboratory, 1991. HPCC White Paper, box 1, MCS Archives, Argonne National Laboratory.

Kassel, Loretta. George Programming Manual. Argonne National Laboratory, 1959. Argonne National Laboratory.

Keller, Herbert B. to Edward H. Levi. Review Committee Report, Applied Mathematics Division, Argonne National Laboratory. July 8, 1970. ANL Review Committee for the AMD. B93-00147, AMD Archives, Argonne National Laboratory.

Kevles, Daniel J. The Physicists: The History of a Scientific Community in Modern America. New York: Alfred A. Knopf, 1978.

Kidder, Tracy. Soul of a New Machine. New York: Little, Brown, and Co., 1981.

Kitchens, Tom. to Hans G. Kaper. Draft on Doe Grand Challenges for Discussion Nov. 18, 1992. November 14, 1991. AMS PI Meetings. box 2, MCS Archives, Argonne National Laboratory.

Kleinman, Daniel Lee. Politics on the Endless Frontier: Postwar Research Policy in the United States. Durham and London: Duke University Press, 1995.

Kliewer, K.L. to P. Messina. Comments on Exit Interview with Mcs Review Committee, Friday, 15 June 1984. June 20, 1984. University of Chicago Review of MCS. box 11, MCS Archives, Argonne National Laboratory.

Knuth, Donald E. "George Forsythe and the Development of Computer Science." Communications of the ACM 15, no. 8 (1972): 721-26.

________. "Von Neumann's First Computer Program." In Papers of John Von Neumann on Computing and Computer Theory, ed. William Aspray and Arthur Burks, 12, 83-96. Cambridge: The MIT Press, 1987.

Kramer, R. F. to H.C. Nickel. Development and Construction Costs, Oak Ridge Computer. Oct. 10, 1950. Laboratory Director's Reading File, ANL 1949-1957, RG326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Lakshmivarahan, S., ed. "Proceedings of the Workshop on Parallel Processing Using the Heterogeneous Element Processor." The University of Oklahoma, 1985.

Landes, David S. The Unbound Prometheus: Technological Change and Industrial Development in Western Europe from 1750 to the Present. Cambridge: The University Press, 1972.

Laney, R.V. to Edward H. Levi. Response to the Report of the Applied Mathematics Division Review Committee. March 30, 1973. Folder 2. Box 53, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Latour, Bruno. Science in Action. Cambridge: Harvard University Press, 1987.

Lax, Peter D. Report of the Panel on Large Scale Computing in Science and Engineering. Washington: National Science Foundation NSF 82-13, 1982.

Leslie, Stuart W. The Cold War and American Science: The Military-Industrial-Academic Complex at MIT and Stanford. New York: Columbia University Press, 1993.

Levin, Eugene. "Grand Challenges to Computational Science." Communications of the ACM 32, no. 12 (1989): 1456-57.

Light, Jennifer. "When Computers Were Women." Technology and Culture 40, no. 3 (1999): 455-83.

Lloyd, G.E.R. Early Greek Science: Thales to Aristotle. New York: W.W. Norton & Co., 1970.

Long Range Central Scientific Computer Facility Needs and Plans of the Argonne National Laboratory. 1968. Computer Needs Committee Aug. 1968-Dec. 1968, Box 17: Computer Information, RG 326: Lab Director's Project Files 1955-1970, GLFRCNARA.

Lundstrom, David E. A Few Good Men from UNIVAC, History of Computing, ed. I. Bernard Cohen and William Aspray. Cambridge: The MIT Press, 1987.

Lusk, Rusty. HPCCI Plans Division Level, Comments from Rusty Lusk. 1991. HPCC Program Correspondence 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

MacKenzie, Donald. "Nuclear Weapons Laboratories and the Development of Supercomputing." In Knowing Machines: Essays on Technical Change, ed. Weibe E. Bijker, W. Bernard Carlson and Trevor Pinch, 99-129. Cambridge: The MIT Press, 1998.

________. Mechanizing Proof: Computing, Risk, and Trust Inside Technology, ed. Weibe E. Bijker, W. Bernard Carlson and Trevor Pinch. Cambridge: The MIT Press, 2001.

Mahoney, Michael S. "Computer Science: The Search for Mathematical Theory." In Science in the Twentieth Century, ed. John Krige and Dominique Pestre, 617-634. Amsterdam B.V.: Harwood Academic Publishers, 1997.

McCorduck, Pamela. Machines Who Think. New York: W.H. Freeman, 1979.

________. "Introduction to the Fifth Generation." Communications of the ACM 26, no. 9 (1983): 629-30.

McCracken, Daniel D., Peter J. Denning, and David H. Brandin. "An Acm Executive Committee Position on the Crisis in Experimental Computer Science." Communications of the ACM 22, no. 9 (1979): 503-04.

McDaniel, Paul W. to A. V. Crewe. Nov. 9, 1961. Users Advisory Committee - Acclerator Users Group - K.R. Symon, Chmn. 1960-62, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Meints, H.R. Design Coordination -- Physics Building Occupancy Meeting, Jan 19, 1950. 1950. RG 326: Laboratory Director's Project File, GLFRCNARA.

Messina, P. Advanced Computing Research Facility, Review Committee Meeting. Argonne, 1985. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.

________. Response to MCS Review Committee Report. Argonne: Argonne National Laboratory, 1985. University of Chicago Review of MCS, box 11, MCS Archives, Argonne National Laboratory.

Messina, Paul C. to K.L. Kliewer. MCS Division Accomplishments, July 1, 1982 - June 30, 1983. May 31, 1983. Major Accomplishments 1980-1994. box 1, MCS Archives, Argonne National Laboratory.

________. to K.L. Kliewer. Programming in a Multiprocessing Environment. October 17, 1983. Major Accomplishments 1980-1994. Box 1, MCS Archives, Argonne National Laboratory.

________. to Hanna H. Gray. Response to Mcs 1983 Review Committee Report. February 10, 1984. University of Chicago Review of MCS. Box 11, MCS Archives, Argonne National Laboratory.

________. to Alan Schriesheim. Suggestion for Item to Be Included in State of the Laboratory Address -- October 12. October 4, 1984. Major Accomplishments 1980-1994. Box 1, MCS Archives, Argonne National Laboratory.

________. to K.L. Kliewer. "New" Strategic Plan. November 26, 1985. Major Accomplishments 1980-1994. Box 1, MCS Archives, Argonne National Laboratory.

_________. to Donald Austin. Advanced Computing Research Seeks Answers to Questions Raised by Multiprocessing. May 1, 1986. Major Accomplishments 1980-1994. Box 1, MCS Archives, Argonne National Laboratory.

Metropolis, Nicolas, J. Howlett, and Rota Gian-Carlo, eds. A History of Computing in the Twentieth Century: A Collection of Essays. New York: Academic Press, Inc., 1980.

Metropolis, Nicolas, D.H. Sharp, W.J. Worlton, and K.R. Ames, eds. Frontiers of Supercomputing. Berkeley: University of California Press, 1986.

Meyerott, R.E. to Louis A. Turner. Estimate of Computing Required by Opacity Group. Feb. 26, 1951. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Miller, W.F. to G.A. Erskine. December 7, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

_________. Mathematics and Computer Research at the Argonne National Laboratory. 1961. Box 11, Laboratory Director's Reading File, ANL, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

_________. Applied Mathematics Division Long Range Plan. Argonne, 1962. AMD Budget Report, 1962, B93-11025 Director's Subject File, MCS Archives, Argonne National Laboratory.

_________. to J. R. Gilbreath. Remarks on the Report of the Review Committee for the Applied Mathematics Division, Argonne National Laboratory. Jan. 10, 1963. Policy Advisory Board, 1957-1967. 11, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Morse, Jr., E.H. to Glenn T. Seaborg. Multiple Procurement of Computers on Deferred Payment Plans. February 9, 1971. Folder 1. 7847, RG 326 Records of the Atomic Energy Commission, collection #9, NARA II.

Murray, Charles J. The Supermen: The Story of Seymour Cray and the Technical Wizards Behind the Supercomputer. New York: John Wiley & Sons, Inc., 1997.

Nash, Stephen G., ed. A History of Scientific Computing, Acm Pres History Series. New York: ACM Press, 1990.

NATS Test Sites as of April 8, 1971. Argonne: Argonne National Laboratory, 1971. Personal Papers of Dr. Wayne Cowell.

Naur, Peter, and Brian Randell, eds. Software Engineering: Report on a Conference Sponsored by the Nato Science Committee. Garmish, Germany 7th to 11th October, 1968, 1969.

Newell, Allen, Alan J. Perlis, and Herbert A. Simon. "Computer Science." Science 157, no. 3795 (1967): 1373-74.

Norberg, Arthur L. "High Technology Calculation in the Early Twentieth Century: Punched Card Machinery in Business and Government." Technology and Culture 31 (1990): 753-779.

Norberg, Arthur L., and Judy E. O'Neill. Transforming Computer Technology: Information Processing for the Pentagon, 1962-1986 Johns Hopkins Studies in the History of Technology, ed. Merritt Roe Smith. Baltimore: The Johns Hopkins University Press, 1996.

November, Joe. "Dendral: Automating Hypothesis Formation." Unpublished conference paper, Princeton University, 2004.

Okrent, D. to Donald A. Flanders. Choice of Ibm 704 over Other Computing Machines. Nov. 5, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Oudshoorn, Nelly, and Trevor Pinch, eds. How Users Matter: The Co-Construction of Users and Technology. Cambridge: The MIT Press, 2003.

Owens, Larry. "Mathematicians at War: Warren Weaver and the Applied Mathematics Panel, 1942-1945." In The History of Modern Mathematics: Institutions and Applications, ed. David E. Rowe and John McCleary, 2, 207-305. Boston: Academic Press, 1989.

Pargal, Sheoli S. Strategic Incompatibility in Networks: Essays on the Standardization in Three Industries. Dissertation, Northwestern University, 1992.

Pasta, John R. Minutes of the September 28, 1961 Meeting of the Computer Advisory Group of the Atomic Energy Commission. Washington, D.C.: Atomic Energy Commission, 1961, AEC 553/17. 12, 1334, RG 326: Records of the Atomic Energy Commission, collection #20, NARA.

Pestre, Dominique. "Science, Political Power, and the State." In Companion to Science in the Twentieth Century, ed. John Krige and Dominique Pestre, 61-75. London: Routledge, 1997.

Pieper, Gail W. Major Accomplishments. Argonne: Argonne National Laboratory, 1985. Major Accomplishments, 1980-1994, box 1, MCS Archives, Argonne National Laboratory.

Pinkston, J.T. "Supercomputer Trends and Needs." In Frontiers of Supercomputing, ed. Nicolas Metropolis, D.H. Sharp, W.J. Worlton and K.R. Ames, 124-40. Berkeley: University of California Press, 1986.

Pinson, E.N. Report of the Review Committee for the Applied Mathematics Division. Argonne: Argonne National Laboratory, 1977. Folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Pollack, Seymour V. "The Development of Computer Science." In Studies in Computer Science, ed. Seymour V. Pollack, v. 22, 1-51: The Mathematical Association of America, 1982.

Pool, Robert. "The Third Branch of Science Debuts." Science 256, no. 3 (1992): 44-62.

Porter, Theodore M. Trust in Numbers: The Pursuit of Objectivity in Science and Public Life. Princeton: Princeton University Press, 1995.

Powers, Philip N. to J.W. Givens. Charge to Technical Steering Committee, Institute for Computer Science. December 24, 1968. Personal Papers of Dr. Wayne Cowell.

________. to Milton Rose. February 21, 1969. Personal Papers of Dr. Wayne Cowell.

Price, Derek J. de Solla. Little Science, Big Science. New York: Columbia University Press, 1965.

Proctor, Robert. Cancer Wars: How Politics Shapes What We Know and Don't Know About Cancer. New York: Basic Books, 1995.

Pugh, Emerson W. Building Ibm: Shaping and Industry and Its Technology History of Computing, ed. I. Bernard Cohen and William Aspray. Cambridge: The MIT Press, 1995.

Pugh, Emerson W., and William Aspray. "Creating the Computer Industry." IEEE Annals of the History of Computing 18, no. 2 (1996): 7-17.

Ralston, Anthony. "Computer Science Research -- Storm Clouds in Washington." Communications of the ACM 16, no. 12 (1973): 725-26.

Ralston, Anthony, and Mary Shaw. "Curriculum '78 -- Is Computer Science Really That Unmathematical?" Communications of the ACM 23, no. 2 (1980): 67-70.

Redmond, Kent C., and Thomas M. Smith. Project Whirlwind: The History of a Pioneer Computer. Bedford: Digital Equipment Corporation, 1980.

Rees, Mina. "The Federal Computing Machine Program." Science 112, no. Dec. (1950).

Reid, Constance. Hilbert-Courant. New York: Springer-Verlag, 1986.

Rheingold, Howard. Tools for Thought: The People and Ideas Behind the Next Computer Revolution. New York: Simon & Schuster, 1985.

Rhodes, Richard. The Making of the Atomic Bomb, 1986.

Rice, John R. "On the Present and Future of Scientific Computation." Communications of the ACM 15, no. 7 (1972): 63739.

Richardson, R.G.D. "Applied Mathematics and the Present Crisis." American Mathematical Monthly 50 (1943): 415-423.

Rojas, Raul, and Ulf Hashagen, eds. The First Computers-- History and Architectures. Edited by I. Bernard Cohen and William Aspray, History of Computing. Cambridge, MA: MIT Press, 2000.

Roland, Alex, and Philip Shiman. Strategic Computing: Darpa and the Quest for Machine Intelligence, 1983-1993. Cambridge: The MIT Press, 2002.

Rose, Frank. Into the Heart of the Mind. New York: Harper & Row, 1984.

Rose, J.E. to J. J. Gallagher. Use of IBM Machines at ANL. Jan. 27, 1950. RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, Great Lakes Regional Federal Records Center, NARA, Laboratory Director's Reading File, 1949-1957.

Rosen, Saul. Report of the Review Committee for the Applied Mathematics Division. Argonne: Argonne National Laboratory, 1979. Folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Rowe, David E. "Klein, Hilbert, and the Göttingen Mathematical Tradition." Osiris 2nd series, no. 5 (1989): 186-213.

Royston, R.J. A Synopsis of Long Range Plans. Argonne: Argonne National Laboratory, 1975. Folder 2, box 53, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

________. to Members of the Applied Mathematics Division Review Committee. The Role of the Applied Mathematics Division at Argonne National Laboratory. October 19, 1977. Folder 2, box 53, Records of the AUA, University of Illinois Urbana-Champaign Archives.

________. Comments on AMD Review Committee Report. Argonne: Argonne National Laboratory, 1979. Folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

________. The Applied Mathematics Division in 1980. Argonne: Argonne National Laboratory, 1980. Folder 4, box 154, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

________. Response to the Report of the AUA Review Committee for the Applied Mathematics Division. Argonne: Argonne National Laboratory, 1980. Folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Sachs, Robert G. to Paul W. McDaniel. Response to the Review Committee Report for the Applied Mathematics Division. February 20, 1974. Folder 2, box 53, Records of the AUA, University of Illinois Urbana-Champaign.

________. to John T. Wilson. Comments on the Report of the Review Committee for the Applied Mathematics Division. April 4, 1978. Folder 3, box 153, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Schriesheim, Alan. to James F. Decker. A Parallel Supercomputing Facility for Computational Science. March 22, 1990. Meetings with Don Austin, Friday April 13, 1990. Box 2, MCS Archives, Argonne National Laboratory.

Schwartz, J.T. An Organizational Proposal for the Improvement of U.S. Computer Science. Courant Institute of Mathematical Science, New York University, 1967. Personal papers of Dr. Wayne Cowell.

Seidel, Robert W. "The Cold War and National Laboratories of the Atomic Energy Commission." In Second Conference on Laboratory History, ed. Catherine Westfall, 9-36. Jefferson Laboratory, Newport News, Virginia: U.S. Department of Energy, Thomas Jefferson National Accelerator Facility, 2001.

Shapin, Steven. The Scientific Revolution. Chicago: University of Chicago Press, 1996.

Shapin, Steven, and Simon Schaffer. Leviathan and the Air-Pump: Hobbes, Boyle, and the Experimental Life. Princeton: Princeton University Press, 1985.

Shonka, F.R. to N. Hilberry. Objectives of Instrument Research and Development Division. Feb. 15, 1949. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Spinrad, B.I. to Peter Egelstaff. Aug. 23, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Steier, Rosalie. "NSF Takes the Initiative." Communications of the ACM 27, no. 6 (1984): 528-29.

________. "Panel Cites Need for Common Focus in Computational Mathematics." Communications of the ACM 28, no. 9 (1985): 891.

Stern, Nancy. From ENIAC to UNIVAC: An Appraisal of the Eckert-Mauchly Computers. Bedford: Digital Equipment Corporation, 1981.

Stevens, Rick L. ACRF Projects and Ideas. Argonne: Argonne National Laboratory, 1990. Federal Grand Challenges, box 5: MCS HPCC, MCS Archives, Argonne National Laboratory.

________. MCS and the Washington Plan. Argonne: Argonne National Laboratory, 1990. MCS Files -- General, box 1, MCS Archives, Argonne National Laboratory.

________. to MCS Staff. Criteria for HPCC Project Continuation. May 14, 1991. HPCC Program Correspondence File 1990-1991. Box 5, MCS Archives, Argonne National Laboratory.

Stevens, Rick L., and Rusty Lusk. HPCCI Plans: ACRF. Argonne: Argonne National Laboratory, 1991. HPCC Program Correspondence 1990-1991, box 5, MCS Archives, Argonne National Laboratory.

Stubbins, Warren F. to Philip N. Powers. Aug. 9, 1970. Folder 1. Box 153, Records of the AUA, Argonne Review Committee, University of Illinois Urbana-Champaign, Special Collections.

Sutherland, W. R. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1973. Folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign Special Collections.

Symon, Keith R. to Roger H. Hildebrand. Nov. 17, 1961. Users Advisory Committee - Acclerator Users Group - K.R. Symon, Chmn. 1960-62. Argonne Universities Association Archives, 1945-82, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Tartar, John, Bruce Arden, Taylor Booth, Peter Denning, Ray Miller, and Andries Van Dam. "The 1984 Snowbird Report: Future Issues in Computer Science." Communications of the ACM 28, no. 5 (1985): 490-93.

Titus, James P. "The New NAS Board as a Government Advisor." Communications of the ACM 11, no. 8 (1968): 580.

Todd, John. to Philip N. Powers. Report of the Review Committee for the Applied Mathematics Division. June 27, 1969. Folder 1. AMD Reports, Records of the AUA, University of Illinois Urbana-Champaign Special Collections.

Traub, J. F. "Signum Panel on the Future of Numerical Mathematics." In International Federation of Information Processing Congress 71. Ljubljana, Yugoslavia, 1971.

________. Numerical Mathematics and Computer Science. Pittsburgh: Carnegie-Mellon University, 1972. 5, 14, Computer Oral History Collection, series 2, subseries C, Museum of American History, Smithsonian Institute.

________. "*Quo Vadimus*: Computer Science in a Decade." Communications of the ACM 24, no. 6 (1981): 351-69.

Tropp, Henry. Interview with Margaret Butler, Jim Butler, Dave Jacobsohn, Charles Harrison, Claire Kilty, Burt Garbow, Stan Zawadzki, Bob Kroupa, Franz Morehouse, and Wallace Givens. 1972. box 1, COHC #196, Series 1: Transcripts, subseries B: Research Transcripts, American History Museum, Smithsonian Institute.

Turkle, Sherry. The Second Self: Computers and the Human Spirit. New York: Simon & Schuster, Inc., 1984.

Turner, Louis A. to E. T. Henson. Oct. 20, 1950. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

________. The Role of the National Laboratories During the Next Decade. 1959. Policy Advisory Board Meeting, Oct. 21, 1959, Box 2, RG 326: Policy Advisory Board Meetings, General July 1957-Aug. 1960, GLFRCNARA.

United States Atomic Energy Commission. Atomic Energy and the Life Sciences, Sixth Semi-Annual Report. Washington D.C, Government Printing Office, July, 1949.

Usselman, Steven W. "Fostering a Capacity for Compromise: Business, Government, and the Stages of Innovation in American Computing." IEEE Annals of the History of Computing 18, no. 2 (1996): 30-9.

Varga, R.S. Report of the Review Committee for Applied Mathematics Division. Argonne: Argonne National Laboratory, 1974. folder 2, Box 53, Records of the AUA, AMD Reports 1965-1973, University of Illinois Urbana-Champaign Special Collections.

von Neumann, John. Memorandum. 1945. John von Neumann papers, Library of Congress.

________. to Admiral Bowen. Jan. 23, 1946. John von Neumann Papers, Library of Congress,

________. "First Draft of a Report on the EDVAC." In Papers of John Von Neumann on Computing and Computer Theory, ed. William Aspray and Arthur Burks, 12, 17-82. Cambridge: The MIT Press, 1987.

von Neumann, John, and Herman H. Goldstine. "On the Principles of Large Scale Computing Machines." In Papers of John Von Neumann on Computing and Computer Theory, ed. William Aspray and Arthur Burks, 12, 317-48. Cambridge: The MIT Press, 1987.

Wagner, Al. to Rick L. Stevens. Chemistry HPCC Component. June 3, 1991. HPCC Program Correspondence 1990-1991. Box 5, MCS Archives, Argonne National Laboratory.

Wahl, A.C. to R. M. Adams. Oct. 8, 1968. Computer Needs Committee, August 1968-Dec. 1968. Box 17, RG 326: Lab Director's Project Files, 1955-1970, GLFRCNARA.

________. to Committee on ANL Computer Needs. Summary of Oct. 9, 1968 Meeting of ANL Computer Needs Committee. Oct. 10, 1968. Computer Needs Committee, August 1968-Dec. 1968. Box 17: Computer Information, RG 326: Lab Director's Project Files 1955-1970, GLFRCNARA.

Warwick, Andrew. "The Laboratory of Theory or What's Exact About the Exact Sciences." In The Values of Precision, ed. M. Norton Wise, 311-51. Princeton: Princeton University Press, 1995.

Weber, E. C. to R. M. Adams. Comments Regarding Draft -- Computer Needs Committee Recommendations Concerning Applied Programming Services. March 28, 1969. Computer Needs Committee Jan. 1969-Dec. 1969. Box 7: Computer Information, RG 326: Records of the Atomic Energy Commission, Argonne National Laboratory, GLFRCNARA.

Weinberg, Alvin M. Reflections on Big Science. Cambridge: MIT Press, 1967.

Weinberg, Gerald M. The Psychology of Computer Programming. New York: Von Nostrand Reinhold, 1971.

Westwick, Peter J. The National Labs: Science in an American System, 1947-1974. Cambridge: Harvard University Press, 2003.

Weyl, F.J. "Summary of Conference Discussions and Proposals: Panel Discussion; Opening Remarks." In Proceedings First Conference Training Personnel for the Computing Machine Field: Detroit: Wayne University Press, 1955.

Wheeler, H. J. to L.D.P. King. Abstract for Paper to Be Presented at the American Nuclear Society. Oct. 30, 1956. Laboratory Director's Reading File, 1949-1957, RG 326: Records of the Atomic Energy Commission, Records of Argonne National Laboratory, GLFRCNARA.

Wilkinson, J. H. Rounding Errors in Algebraic Processes. Englewood Cliffs, NJ: Prentice-Hall, 1963.

Wilson, Kenneth G. "Planning for the Future of U.S. Scientific and Engineering Computing." Communications of the ACM 27, no. 4 (1984): 279-80.

________. Grand Challenges to Computational Science, 1987. Applied Mathematics Division Records, Argonne National Laboratory,

________. "Grand Challenges to Computational Science." Future Generation Computer Systems 5, no. 171 (1989).

Wolanski, H.S. "Applications of Computing in the Aircraft Industry." In The Computing Laboratory in the University, ed. Preston C. Hammer, 91-102. Madison: University of Wisconsin Press, 1957.

Worthy, James C. William C. Norris: Portrait of a Maverick. Cambridge: Ballinger Publishing Company, 1987.

Yates, JoAnne. "Co-Evolution of Information-Processing Technology and Use: Interaction between the Life Insurance and Tabulating Industries." Business History Review 68, no. 1 (1993): 1-51.

Oral Interviews

Interview with Margaret Butler, 11-27-2001; 5-15-2002; 9-10-2002, Argonne National Laboratory.

Interview with Joe Cook, 9-10-2002; 10-27-2003, Chicago, IL.

Interview with Wayne Cowell, 11-30-2001; 5-15-2002, Argonne National Laboratory.

Interview with Paul Messina, 5-13-2002, Argonne National Laboratory.

Interview with William Miller, 11-11-2003, via phone.

Special thanks to Thomas Haigh for his interview with James Pool, CalTech, Pasadena, CA July 14, 2002 (unpublished at the time).

Charles N. Yood
Curriculum Vitae
yood@msu.edu Tel: 517-353-9528

Currently: Visiting Assistant Professor, History, Philosophy, and Sociology of Science, the Lyman Briggs School of Science, Michigan State University.

Education:

Ph.D. Pennsylvania State University: History of Technology, History of Science, Environmental History, 2005
M.A. Florida State University 1995: American History, Environmental History, Modern Germany
B.A. Florida State University 1991, History; Philosophy
---- Mercer University 1987-89, Biomedical Engineering

Dissertation: Argonne National Laboratory and the Rise of Computer and Computational Science, 1946-1992; Advisor, Dr. Robert Proctor, Stanford University.

Papers and Publications

"The History of Computing at the Consumption Junction," IEEE Annals of the History of Computing, January-March, 2005

"Computer Technology," contribution to the Encyclopedia of the Midwest, "Science, Technology, and Health and Medicine" section (Indiana University Press, forthcoming).

"An American Contribution: Jedidiah Morse, Dobson's *Encyclopaedia* and the Shaping of American Identity," Looking Up: Science and Observation in the Early Modern Period, ed. Jason M. Kelly (Ottowa: Legas Publishing, 2002).

"The Law" vs. "The People": Twelfth Round Table for Law and Semiotics, co-edited with Roberta Kevelson, Ralph Lindgren, and William Pencak (New York: Peter Lang, 1999).

"The Evolution of Machine Intelligence," essay review, Research in Philosophy and Technology, vol. 18 (Stamford: JAI Press, Inc., 1999).

"Two Bibliographic Guides to the History of Computing," book review, Bulletin of Science, Technology, & Society. Vol. 17, nos. 5 and 6, 1997, pp. 307-8.