The Student Information Processing Board:

the social and technical impact of an MIT student group

Chian Chuu

Michael Lei

Chiyu Liang

Alida Tei

6.933 Structure of Engineering Revolutions

Professor David Mindell

TA Chen-Pang Yeang

December 14, 2001

Table of Contents

List of Figures

Introduction

Donald MacKenzie's idea of heterogeneous engineering, the development of technology as a result of social and technical efforts,[1] attributes innovation to the developers of technology. However, this theory fails to recognize the importance of the end-users in the success of that technology. It can be argued that the users of technology can bring about technological innovation as well. Student Information Processing Board (SIPB) members were all end-users of computer technology, but had a profound technological effect on the MIT computing environment. SIPB was a volunteer student group at MIT, founded with the vision that computers should be accessible and could become useful for the general populace.

The impact of SIPB on the MIT computing started with their efforts to increase student access to computing resources. Initially, undergraduate students did not have access to computing, despite the developments in time-sharing technology. Time-sharing was a social technology that changed the way users interacted with machines. By making the technology accessible to a greater number of users, SIPB brought about technical and social innovation to the MIT community. Once the user pool included the entire MIT community, SIPB shifted their focus towards making computers useful. By developing applications that would enhance a user's computing experience, SIPB improved the social interactions made possible by technology. Their computing expertise and services allowed average MIT students to interact successfully with computers; thus SIPB members were the bridge between the highly technical community and the average MIT student. The efforts of SIPB to provide access to students and make that access useful blurred the line between a social and technical innovation, helping to bridge the Latourian "Great Divide."[2]

Technology also played an important role in the culture of SIPB, as it is a common interest that ties all of its members together. The culture thrives upon the technological savvy of its members and their passion for computing. However, the role of technology in the story of SIPB does not end with a member's graduation from MIT. Technology enables the network of SIPB alumni to stay in touch with each other and current members. Additionally, many SIPB alumni pursue careers in technology-related fields.

The history of SIPB is a story of the founding, development, and growth of an organization that uses existing technologies. This is distinct and unique from typical sociological

[1] MacKenzie, Donald A.
[2] Latour, Bruno.

and historical accounts of the development of technology. Instead of narrating the project history by describing the dynamics, decisions and actions of the companies, laboratories and individuals who develop technology, we are instead interested in how the technology affects those who use it. The history of SIPB shows how the users of technology brought about significant technological innovation.

Computing at MIT, 1957-1969

The MIT Computation Center was the first centralized computing facility at MIT. The aim of the Computation Center was to promote the usefulness of computation in education and explore its potential in various disciplines. However, access to the computers was limited to research involving machine computation--undergraduates were generally not granted computer time. As interest in computing began to spread, the Institute required more computing power in order to accommodate requests for computer time. As a result of increased demand, MIT replaced the first mainframe IBM 704 with a 709 in 1960, and then the 709 with the 7090 in 1962. However, even as computers got more and more powerful, access was never expanded to include undergraduates.

Time-sharing was the technology that would change the configuration of social computing at MIT, eventually paving the way for undergraduate access. Time-sharing was developed because many researchers were frustrated with the tedious process of running programs. Researchers had to wait up to a day for results, due to the limitations of batch processing. In the 1960's, intelligent time-sharing operating systems were developed, allowing multiple users to operate simultaneously on the same machine. Research in time-sharing resulted in the formation of CTSS (Compatible Time-Sharing System) and MULTICS (MULTiplexed Information and Computing Service). It increased the number of users running applications simultaneously on the same system and changed the way that users interacted with machines.

The Computation Center

"The primary purpose [of the Computation Center] is to demonstrate that such machines are as important a part of the educational equipment of a modern college as are chemistry laboratories, for example. A secondary purpose is to foster research in the use of computing machines, particularly in those fields of application which have not yet been explored or exploited fully."[3]

-- Philip Morse, Director of the MIT Computation Center, February 10, 1960, in a proposal to IBM to upgrade to 7090 from 704.

Despite the good intentions of Philip Morse in the quote above, the educational benefits of the Computation Center was limited to a select few faculty members and graduate students. Undergraduates were never able to enhance their education through use of the Computation Center, due to limited computer time. However, the founding of the Computation Center was the first development at MIT incorporating computing with education. It was founded as a joint effort between MIT and IBM, when IBM provided a 704 machine in March of 1957. "We are pleased to announce the arrival at 8:30 am, March 4, 1957, of a spanking new 43,773 pound IBM Brainchild,"[4] said Morse. Morse was clearly excited about the potential educational benefits of the new computer.

The Computation Center was located in the Compton Labs, in what is now the physics reading room in building 26. Construction of the Compton Labs had already begun when a facility for the Computation Center was needed. Architects building the Compton Labs intended to build it in the fashionable style at the time, which was raised off the ground on stilts. But because the IBM-704 was such a huge machine, extra space had to be provided to accommodate it. Much to the architects' chagrin, MIT forced them to build a ground floor to put the computer in. As a result, the architects built a ground floor, but tried to maintain the illusion of a building on stilts by eliminating windows on the ground floor and painting the walls blue[5].

It took one month to install the IBM-704 because of its sheer size. This system could perform 40,000 additions or subtractions, or 5,000 multiplications or divisions of 10-digit numbers per second. The machine had a magnetic "core" memory up to 32K words; each word in memory could be accessed in 12 millionths of a second[6].

---

[3] Morse. Memorandum to IBM.
[4] Morse. Memorandum to the MIT community.
[5] Corbato interview.
[6] IBM Archives.

Under a special cooperative agreement with IBM, MIT facilities would be "generously offered"[7] an IBM-704 Electronic Data Processing Machine at no cost. The machines were expensive, as IBM produced only eighty 704 and 705 (virtually identical to the 704) computers for commercial or classified operations in the following year (1958). The agreement was reached with the MIT Computation Center as a facility for MIT use, "with the understanding that all the colleges in New England may share in its use."[8] It would provide up to 7 hours per day of machine time to participating colleges other than MIT. In 1957, 27 colleges and universities in New England shared the 704 system. In addition, IBM provided the maintenance staff, machine operators who were constantly working at the Computation Center.

Priority on the 704 was given to students taking courses in machine programming and applications, thesis research and research projects that explored new applications of computation. Such topics included fundamental research in science, engineering, applied business, data processing, tracking of personnel in computer operations, but no classified work. Questionnaires and applications were sent around for "Scheduled usage of 704 Computer Time." In order to be granted time on the machine, potential computer users would have to submit a two-page application in addition to an abstract and proposal. Once time was granted to a user, the user would then also have to compose quarterly progress reports of his research.

Programming on these mainframe computers was conducted with batch processing. To create a program, one would have to write on paper each line to be executed by the computer. A keypunch operator would punch the program onto cards, which would be submitted to the computer facility's personnel that recorded them on a tape. Finally, the tape would run and the results of the output would be sent to a line printer. One would get back the results of their job usually a day later[9]. Batch processing was clearly tedious and time-consuming. Professor Fernando Corbato, Associate Director of the Computation Center, felt that batch processing was a source of "personal frustration" and would not allow "programmers to get back to their machines and to run more ambitious programs."[10] As programs became more sophisticated, queues for getting a job processed became longer and longer. Debugging programs was especially difficult because new punched cards would have to be submitted. According to then - programming student Dave Burmaster ('69), "If you had to debug the program, you turned it in

[7] Morse. Memorandum to IBM.
[8] Morse. Memorandum to IBM.
[9] Burmaster interview.
[10] Corbato interview.

again, a day later, and you found out if it had run. You kept doing this until the week had run out, or the homework assignment was due. There was no debugging, there was no access, no sitting there at the terminal, rewriting the program."[11]

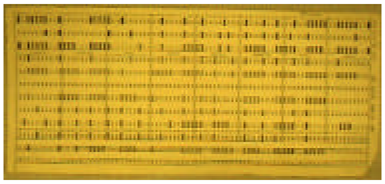


Figure 1: Punch Card

Another educational use of the computing resources occurred in 1958 when the Computation Center started teaching summer classes on computer programming and training on how to use the 704. As a result, interest in programmable computation was piqued. MIT usage of the IBM 704 can be seen in the diagram below, taken from the 1960 report by Herbert Teager regarding MIT computation growth.

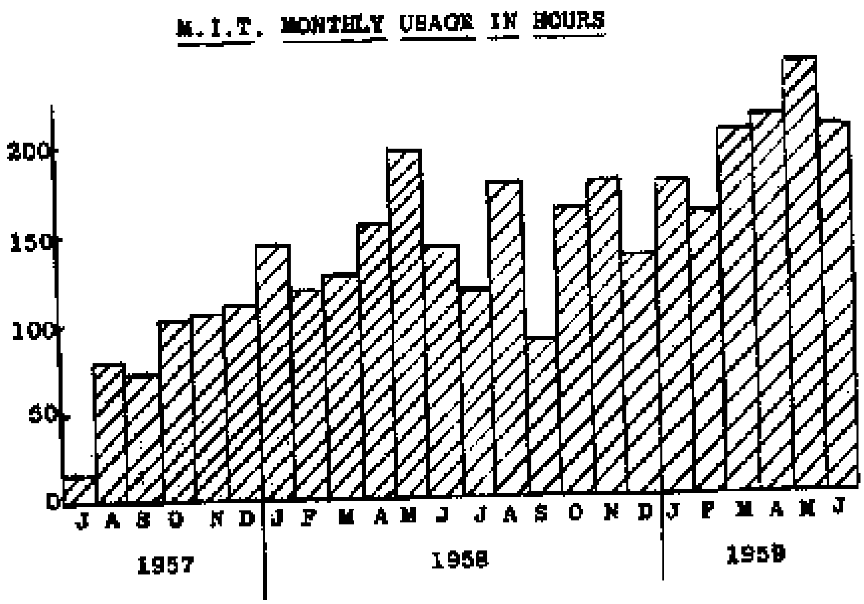


Figure 2: MIT Monthly computer usage (in hours) of the IBM 704. [11]

[11] Burmaster Interview

The use of the IBM 704 increased steadily from July 1957, as can be seen in Figure 2, until it was replaced by the IBM 709 in July of 1960, which was able to accommodate the increasing computation load. The 709 was followed by the IBM 7090, the transistorized version of the 709, in January of 1962.

The 7090 had 32K of core memory to handle the operating system and the input from a typewriter, and disk memory for other programs. It had five times the computing speed of the 709 due to its transistor technology and could execute 210,000 instructions per second. A word in memory could be accessed in 2.4 millionths of a second.

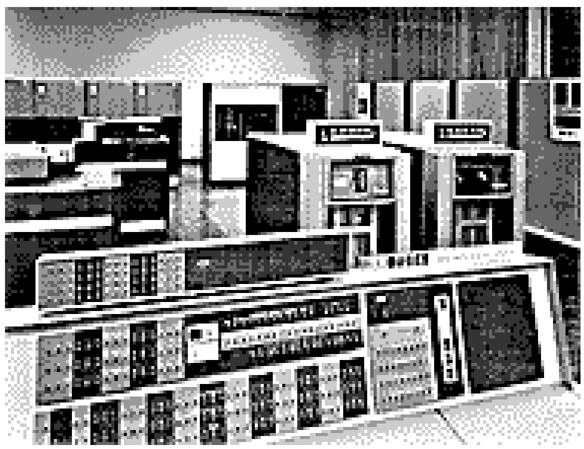


Figure 3: Illustration of IBM 7094

Because of the greater capacity for computation in the 7090 relative to the 704 and 709, monthly usage of the 7090 rapidly exceeded that of the 704[12]. In 1963, MIT upgraded to an IBM 7094 (shown above[13]).

The IBM-7094 was one of the biggest, fastest machines available, able to add floating-point numbers at a speed of about 0.35 MIPS. A standard 7094 had 32K 36-bit words of memory. Its data channels could access memory and run simple programs to do I/O once started by the CPU, and could cause a CPU interrupt when the I/O finished. It could execute 500,000 computations per second and could access a memory word every 2 microseconds. Its cost was around $3.5 million.

[12] Teager, Herbert M, "Summary of 7094 Computer Usage during period Jan 1965-March 1965."
[13] Multicians website.

Thus the Computation Center was able to introduce the idea of applying computers for educational applications, but resources were limited and demand was too high. Even though MIT upgraded mainframe technology, demand would not be satiated until the development of time-sharing systems.

Time-Sharing Systems

Compatible Time-Sharing System (CTSS)

In 1959, MIT Professor John McCarthy theorized that a system could be built to share the resources of a powerful mainframe computer. His theory was based on his research in artificial intelligence, which he had started in the mid-50's. McCarthy proposed a time-sharing system with a time-sharing supervisor, which could intelligently allocate computational resources based on algorithms[14]. "I think the proposal points to the way all computers will be operated in the future, and we have a chance to pioneer a big step forward in the way computers are used."[15] Time-sharing was an essential step towards more interactive computing and McCarthy's vision of intelligent interactions between humans and computers.

McCarthy's time-sharing theory was implemented by Corbato in one of the first working time-sharing systems, CTSS (Compatible Time-Sharing System). "Compatible" meant that the computer could run time-sharing experiments while still providing batch operations, thus allowing the Computation Center to make the transition from batch processing to time-sharing gradually.[16] Time-sharing systems processed many jobs simultaneously instead of dedicating their computing power to one task at a time. The idea of time-sharing systems was a breakthrough because it changed the way that users interacted with machines. Instead of submitting timecards to be processed, users could interact with the machine and receive immediate feedback, thus giving rise to the idea of social computing. Time-sharing systems allowed for dozens of terminals to be scattered around campus, no longer forcing users to bring their punched cards to the central computer facility. The computer could switch among different jobs so fast that each user had the impression of running a real-time, interactive application.[17] Each interaction with the computer was alternated between different users at a rate faster than human reaction time (~0.2 seconds).

---

[14] McCarthy, John.
[15] McCarthy, John.
[16] Abelson, Hal.

In November 1961, Corbato demonstrated a crude, "quick and dirty"[18] prototype system with four terminals, built just to demonstrate the usefulness of time-sharing, was running on an IBM 709; the system was switched to the Computation Center's IBM 7090 in the spring of 1962. This demonstration was effective in proving the usefulness and feasibility of compatible time-sharing systems. It was interestin to note that Corbato had trouble convincing vendors of the benefits of time-sharing, mainly IBM. He described this effort as "like trying to convince General Motors to make airplanes."[19]

Corbato's system had the capability to support four simultaneous users. One user worked in the background system using the 7090's standard monitor system while each of the three other users worked in the foreground system using a flexowriter typewriter, which could punch paper tape or output the contents of paper tape. The typewriters were connected to the Direct Data Connection channel of the 7090. A 60-cycle interrupt clock as well as a trapping system for user-initiated input-output commands were installed on the 7090. The interrupt clock would pause programs as appropriate in order to alternate between users while the trapping system controlled I/O such that each user does not see another user's commands or output[20].

Time-sharing systems introduced the time-sharing supervisor (TSS), the software that would run in the mainframe's core memory. Of the 32,000 words of memory in core storage, the TSS remained in the lower 5,000 words while the remaining 27,000 words are allocated for the four users. The TSS handled all of the commands typed by the user as well as all input-output of the typewriters, thus allowing the supervisor to initiate processing by the mainframe when appropriate. Top priority for the next quantum of computation was given to the next command in the "waiting queue." Once the present calculation was finished, the data was read out to the appropriate dump tape and the new command program was initiated and added to the "working queue." If the waiting queue was empty, then the TSS "executed a simple round-robin of those foreground user programs in the working status queue."[21] If both the working and waiting queues were empty, the TSS ran the background user program until foreground activity was reestablished.

---

17 Abelson, Hal.
18 Corbato interview.
19 Corbato interview.
20 Daggett, et al.
21 Daggett, et al.

Corbato's system implemented hardware and software changes to the MIT mainframe, allowing more users to use the computer. Simultaneous social computing was now possible, paving the way for more sophisticated time-sharing systems like Multics.

MULTICS

The development of Multics, led by Professor Corbato, began in 1965. It was a time-sharing system started as a joint project by MIT Project MAC (Multiple Access Computers), Bell Telephone Laboratories, and General Electric (GE). Initially, it was supposed to be a one to two year project, but it turned out to last four to five years. The main motivation behind CTSS was to prove that time-sharing was feasible, thus a lot of issues that were not addressed were incorporated in the design of Multics. By increasing the number of terminals from four to thirty, Multics allowed for more efficient usage and more simultaneous users, but did not alleviate the problem of limited access[22]. Computing resources were still confined to graduate students, faculty, and certain undergraduates working in Project MAC. However, despite the denial of direct access for undergraduates, Multics was another technological applicaton that had a profound effect on campus.

Multics was a much better time-sharing system than CTSS, because it improved social computing, and addressed security issues, and allowed better file-sharing routines. Like CTSS, Multics also supported multiple simultaneous users, and further improved social computing by allowing for students to share data, programs, and other computer resources with each other in an efficient and secure way. "If two users wanted to run the same program at the same time, Multics would load only a single copy of the program"[23]. Thus two students could work on individual projects while sharing the same resources, enhancing the educational benefits of computers. Security and file-sharing ideas such as virtual memory and memory rings of protection were also employed, as Corbato and Salzer indicate: "Ideas such as virtual memory access to on line storage, parallel process organization, routine but controlled information sharing, dynamic linking of procedures, and high-level language implementation have proven remarkably compatible and complementary."[24] Thus Multics had profound technological impact

---

[22] Multicians Website.
[23] Abelson, Hal.
[24] Clingen, C.T.

due to the technical improvements relative to CTSS as well as the increase in social user interactions.

Similar to the CTSS, convincing commercial vendors of the benefits of Multics was a difficult task. When it came time to select a vendor for the computer that would support Multics, IBM was not interested. Time-sharing was at that time what Clayton Christensen referred to as a "disruptive technology"[25]; IBM was reluctant to introduce time-sharing in the mainframe that was being developed at the time, the 360/65, because of their established success in batch-processing machines. On the other hand, Project MAC was developing a relationship with GE with the help of Professor Joseph Weizenbaum, a former GE employee. GE agreed to built the GE-645, which would become Project MAC's platform for Multics. This business relationship between Project MAC and GE would later influence MIT decisions regarding the Computation Center.

IBM was beginning to get anxious about the new relationship between GE and Project MAC, but they still did not want to pursue time-sharing. In 1967, IBM wanted to bring the 360/7 to MIT. In order to maintain good relations with IBM for future business endeavors, Gordon Brown, the Dean of the School of Engineering, organized for IBM to place their new equipment in a new facility in the Brown building, building 39. Brown also took over the Computation Center, and renamed it Information Processing Services (IPS).

[25] Christensen, Clayton.

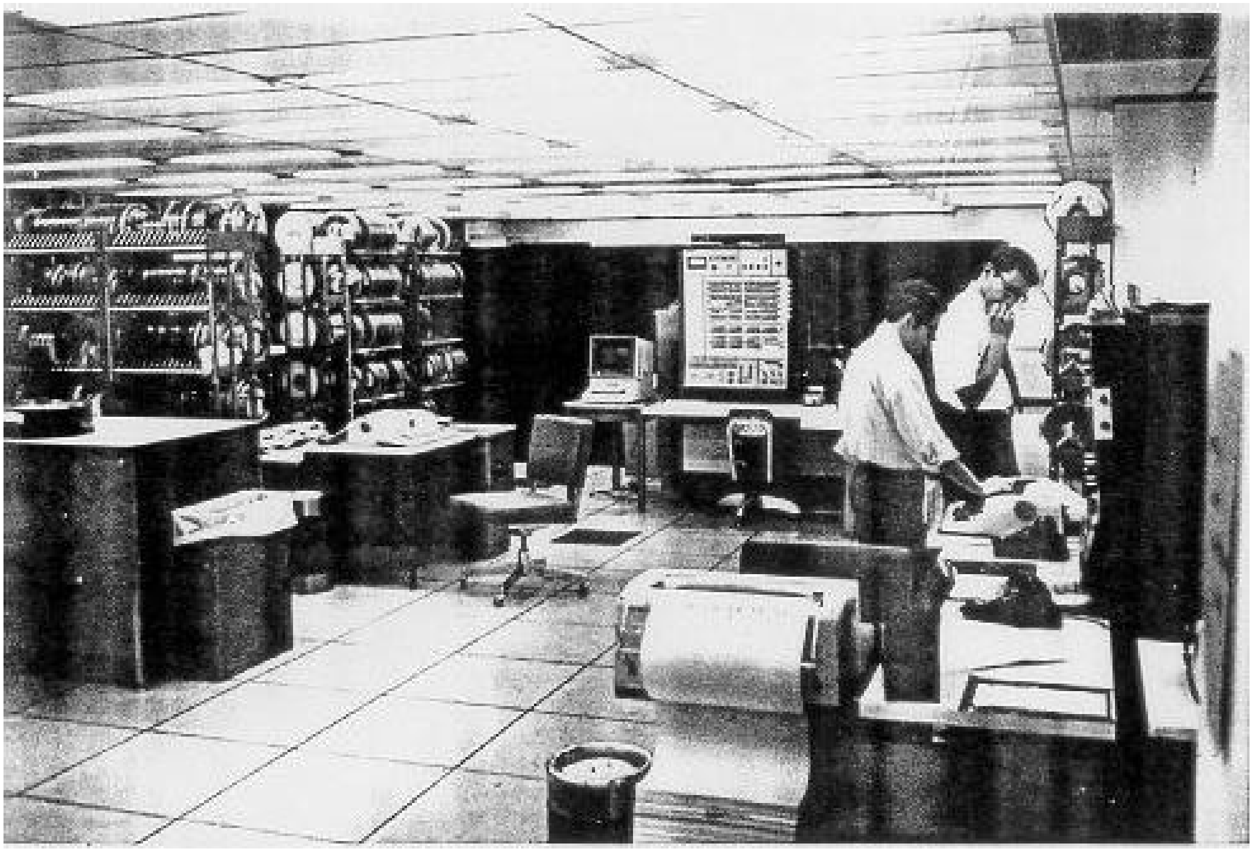


Figure 4: Man Working on IBM 360

A common philosophy about computing soon emerged with the new time-sharing system. The developers of Multics, Project MAC, shared similar visions on the future of computation. Project MAC was founded in 1963 with the visions of Professor Bob Fano and Professor J.C.R. Licklider. Licklider worked on small time-sharing systems with McCarthy on a DEC PDP-1. Fano came to the conclusion that computing was ready to emerge as an academic discipline, and that MIT should start a research laboratory for computation.[26] Both believed that computers had a far greater potential than general-purpose simulation systems, since any logical step could be implemented using them. We will see that this philosophy towards computing was shared by the founding members of SIPB.

An interesting sidenote is that in 1969, Bell researchers working on Multics decided to continue developing some of the Multics ideas, most remarkably the tree-structured file system.

[26] Abelson, Hal.

By cutting corners, they created a working prototype pretty quickly. Another Bell researcher suggested that the operating system be called *UNICS*, a joke that meant *castrated Multics*[27]. Eventually, the Bell crew changed the name of their operating system to UNIX.

A big organizational change also happened within project MAC around 1969, when, under Minsky, the AI team got their own computers (DEC PDP-6, PDP-7 and PDP-10), and team members wrote their own operating system called the *Incompatible Time-Sharing System* (ITS), a direct slap at Corbato's Compatible Time-Sharing System. In 1970, Minsky seceded from Project MAC and created the Artificial Intelligence Laboratory.[28] According to Corbato, they built their own time-sharing system as a response to Multics, but to fulfill their own needs, with no security implementation, only self-policing.[29] This worked fine as long as only a few people used it, the ITS team had a different set of objectives than the Multics', and wanted to have their own style of management. The implications of Minsky's actions only limited accessibility even more, by creating a system designed specifically for the AI Lab members. This action undermined the trend of increased social computing by time-sharing systems, and would be balanced by the actions of SIPB, which was founded at the same time in 1969.

Founding of SIPB, 1969

The technological development of time-sharing provided solid ground for the founding of SIPB. SIPB was able to increase the user pool to include undergraduates because the computing needs of researchers had become well supported.

SIPB was founded not on a niche technology, but rather on a critical void to follow and utilize technology at MIT, to make computers useful and accessible for a wider base of users. The void was a lack of student accessibility to computers, as well as a lack of understanding among the MIT community of the benefits of computers. The history of SIPB revolves around the idea of the relationship between technical and social innovation by the users of technology. Although SIPB did not create time-sharing systems, they caused technological innovation by linking a larger user pool to the computing technology. They are also the bridge between the highly technical community who developed time-sharing and the end users of the technology.

In 1968, many people, especially the founding SIPB members, saw the computational

[27] Abelson, Hal.
[28] Abelson, Hal.
[29] Corbato interview.

powers that were now possible due to the time-sharing systems. However, there was still a need to extend the social reach and technical benefit of these systems to undergraduates. At the time, MIT had the option for students to create their own seminars, and to achieve course credit under the supervision of an MIT faculty member. Under Prof. Merton Kahne, a student group of six to eight members including David E. Burmaster decided to create a seminar discussing the MIT undergraduate experience. Even though the group consisted of undergraduates from different majors, the members had the notion that computers would be incredibly useful. They felt it was important to follow the trend, and to take advantage and fully utilize the developing technology.

There was clearly a need for change. The members of the seminar sought the help of key professors like Minsky, Corbato, and Licklider to find a solution to the problem. Licklider had just been appointed head of Project MAC, and was an advocate for technological progress. On his second meeting in January 1969 with the members of the seminar, Prof. Licklider decided to allott $100,000 from Project MAC to support undergraduate computing. That sum is equivalent to over a million dollars today. Dave Burmaster, who was one of the students present at the meeting, said he "practically fell off his chair."[24] This allocation was then approved by Provost Jerome Wiesner and the head of IPS Richard Mills, thus gaining the support of the MIT administration.

"All in the course of eight days, we had gone from nothing, a dream, to what to do next. Then, of course, reality set in. Dave Burmaster recollected that there was a huge question of where was this money going to come from, how would it be distributed, what computers would we buy?"[30] At that time, the members of the seminar were aware of an IBM 7094 mainframe running CTSS that was going to be decomissioned because it was too slow, and likely to fail. This mainframe allowed thirty simultaneous users. Despite the cooperation of Wiesner, Licklider, and Mills, the members successfully obtained the system, but the high costs of maintenance prevented them from providing access to the system. However, their efforts helped garner support for gaining student access to computing. At that time, many other changes were coming about. The seminar was coming to a close, and many of the students were graduating. The members of the seminar then decided to recruit younger undergraduates to further and continue the cause.

Among the students that were recruited, two of them, Bob Frankston, and Ed Fox, took

[24] Burmaster interview.
[30] Burmaster interview.

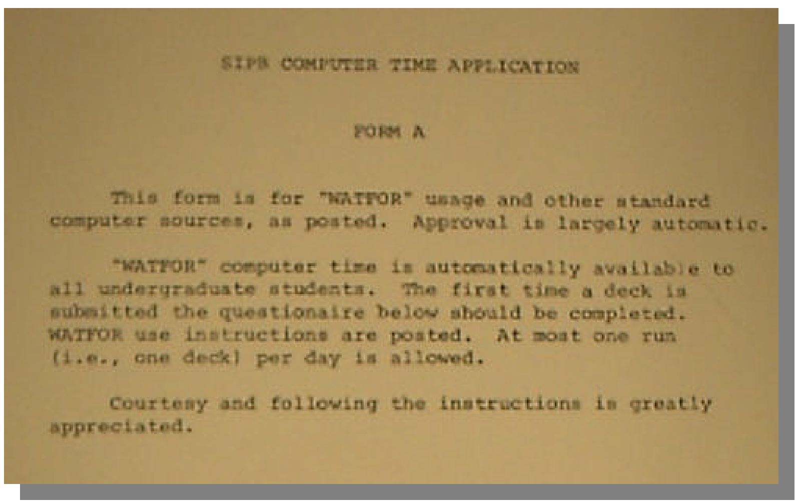
up the cause and founded the volunteer student group called the Student Information Processing Board (SIPB). They were freshmen at the time, but were already recognizing the utilities and capabilities of computers. Frankston helped start the computer curriculum at his high school (Stuyvesant High School), and would later work for Project MAC. Fox was working as a computer consultant in the Computation Center at the time. Fox was also teaching computer classes for high school students in the area and was starting the first MIT student chapter of ACM (Association of Computation Machinery). Based on their past experiences, both were already interested in integrating technical and social aspects of computing technology. They were electrical engineering and computer science majors, and believed firmly that the MIT administration was not doing enough to provide social computing to undergraduates. Thus SIPB took on the role of being the MIT student voice for computing. According to Frankston, the initial goal of SIPB was to convince people that computers were useful both technically and socially and access should be extended to include the entire MIT community.[31]

One of the initial efforts to obtain computing resources failed. SIPB approached Ken Olson, the head of DEC, and asked for a top-end time-sharing system for students.[32] The request was not approved, but another effort proved successful. Working with Provost Wiesner, a plan to distribute computing time to students was developed. Computers in building 39 were owned by MIT, but usage was paid for under federal contracts. If a professor had a research contract through a federal agency, then he would obtain funding from the agency to complete research on those computers. Those computers were busy from 8am to midnight. After midnight till 8am the next morning, the mainframe computer was guarded by staff in a locked room. The idea suggested by Provost Weisner was to allow undergraduate students to submit requests for a certain amount of computer time during the midnight to 8am shift. These requests would have to be tied to an undergraduate thesis, a course, or a personal research program, and had to be approved by the SIPB group. If approved, the student would be given a coupon that would correspond to an allotted amount of computer time.[33] A problem came up with the issue of rates. The government would only pay for computer time at a rate that was not higher than the lowest rate charged others. If the university had given student free access, then the government would have had free access as well. Provost Weisner came up with the idea of asking the government to approve giving the students this computer time for free by exchanging time coupons. The

[31] Frankston interview.
[32] Fox email correspondence

coupons were worth thousands of dollars in computer time, but could not be exchanged for actual money. These coupons were hence referred to as "funny money."[34]

The setup of allowing students to directly use the IPS resources was an incredible breakthrough for student computing and technological progress at MIT. Allowing direct access to these computers opened up opportunities in undergraduate education that had previously been too computationally intensive to perform without computers. SIPB was soon flooded with more requests for computing time than they could handle. A typical amount of time to be allotted for an undergraduate research project was thirty minutes for a semester, and if more time was needed, another request had to be submitted. The effect of this new computing opportunity rippled throughout the MIT community. Professors were amazed by the extensive term projects of students, since they were now able to include more computationally intensive models. "It opened up the faculty to a new point of view,"[25] said Burmaster. The faculty were convinced that it was even more important to further and continue the increase of computing resources.


SIPB COMPUTER TIME APPLICATION

FORM A

This form is for "WATFOR" usage and other standard computer sources, as posted. Approval is largely automatic.

"WATFOR" computer time is automatically available to all undergraduate students. The first time a deck is submitted the questionaire below should be completed. WATFOR use instructions are posted. At most one run (i.e., one deck) per day is allowed.

Courtesy and following the instructions is greatly appreciated.

Figure 5: SIPB Computer Time Application[35]

In addition to being in charge of distributing computer time, SIPB introduced technical software projects to the MIT community. SIPB had to decide which projects to fund, since the high expense of computing limited them from trying everything. The initial student applications that they approved involved the use of SPSP, a statistical package, and projects that involved

[33] Burmaster interview.
[34] Fox email correspondence.
[25] Burmaster interview.
[35] Courtesy of SIPB Office

programming for student classes.[26]

Despite the breakthrough in student access to computing, the time-sharing system was not the final solution. There was simply too high of a demand to be able to provide enough access. Thus the next breakthrough in MIT computing happened with the next wave of technological advancements in the computer industry.

Development of SIPB

This next wave would take place in the late 70's. IBM, DEC, and Apple all flourished, and the move from large mainframes to the age of personal computing had just begun. With the variety of computing options available, many colleges and universities were turning to the Interuniversity Communications Council (EDUCOM), a non-profit computing consulting firm. When Weston Burner became the director of IPS in 1977, he hired EDUCOM to scrutinize and provide recommendations about the current state of computing, as well as IPS structure and services. In the EDUCOM report issued to MIT in July of 1980, the consultants begin the extensive report with: "In today's world of escalating computer demands and scarce resources we would expect to find on any campus some angry administrative computer users who feel that their needs are not being met. What shocked us at MIT was that most administrators we talked with had gone beyond the point of being angry or hostile at the situation and were, instead, demoralized."[36] Many of the faculty felt that IPS was incapable of efficiently providing computing solutions to MIT-according to EDUCOM's report, the faculty tolerated IPS because they felt that Burner was an able director.

However, as a direct result of the EDUCOM report, Corbato was appointed Principal Officer for Information Systems and Computing. Along with Provost Francis Low, Corbato established the Committee on the Computing Environment in November 20, 1980. The committee was charged with providing recommendations to the institute regarding computing goals in the future (specifically in 1990). At this time SIPB had grown in credibility in the minds of MIT administration and MIT realized that student input about the Institute's future computing environment would be valuable to the process. One member of SIPB was chosen to be the student voice on the Committee. The first SIPB representative was the chairman at the time, William York ('82).

[26] Frankston email correspondence.
[36] EDUCOMM Report to MIT

The report from the Committee on the Computing Environment was not submitted to the provost until April 2, 1984. York was followed in successive years by Wendy Rowe ('83) and Ramin D. Zabih ('85). Amongst the committee's most notable recommendations was to construct a single source of campus-wide information systems, to integrate four separate entities: Information Processing Service, Project Athena, Telecommunications, and Purchasing.[37] These recommendations supported the technical transition from giant mainframe computers to the idea of decentralized workstations at MIT. The exponential growth of the affordability of computers, portability, and power made it easier to convince the administration to increase the number and distribution of computers around campus. In 1983, SIPB was able to obtain funding to buy fourteen new video terminals for living groups, thus bringing technical computation to a broader audience. Through its role on the committee, SIPB was able to substantially build on the existing user pool to include even more undergraduates. As a result of their successful effort to distribute workstations around campus and their key role on the Committee on the Computing Environment, SIPB gained a solid credible presence on campus.

The ongoing computer revolution would also affect the technical projects and responsibilities of SIPB. From the first statistical package run on the CTSS machines in 1969, to multi-platform software that was written and supported for the development of Athena, the skillset of SIPB adapted to different technologies. In 1969, the only programming languages were Fortran, PL1, and assembly languages. However, over the course of 32 years, many programming languages have been developed, such as C, Perl, Java, C++, and Lisp. The variety of these languages enables SIPB to tailor towards specific programming applications. In 2001, two SIPB members, Keith Winstein and Marc Horowitz upset the movie industry when they wrote a 6-line program in Perl to crack encrypted DVDs. Previous attempts to write DVD cracking programs were attempted in other programming languages, and were a lot more lines of code, and were not as compact and elegant.[38]

Here is the 6-line Perl code for cracking DVD encryptio program:

```
#!/usr/bin/perl
# 472-byte qrpff, Keith Winstein and Marc Horowitz <sipb-iap-dvd@mit.edu>
# MPEG 2 PS VOB file -> descrambled output on stdout.
# usage: perl -I <k1>:<k2>:<k3>:<k4>:<k5> qrpff
# where k1..k5 are the title key bytes in least to most-significant order
s''$/=\2048;while(<>{G=29;R=142;if((@a=unqT="C*",_)[20]&48){D=89;_=unqb24,qT,
@
```

[37] EDUCOM Report to MIT
[38] Gallery of CSS Descramblers.

```
b=map{ord qB8,unqb8,qT,_^$a[--
D]}@INC;s/...$/1$&/;Q=unqV,qb25,_;H=73;O=$b[4]<<
9|256|$b[3];Q=Q>>8^(P=(E=255)&(Q>>12^Q>>4^Q/8^Q))<<17,O=O>>8^(E&(F=(S=O>>14&7
^O)^S*8^S<<6))<<9,_=(map{U=_%16orE^=R^=110&(S=(unqT,"\xb\ntd\xbz\x14d")[_/16%
8]);E^=(72,@z=(64,72,G^=12*(U-
2?0:S&17)),H^=_%64?12:0,@z)[_%8]}(16..271))[_]^((D
>>=8)+=P+(~F&E))for128..$#a}print+qT,@a}';s/[D-HO-U_]/\$$&/g;s/q/pack+/g;eval
```

SIPB also kept up with the developments in technology and was quick to embrace the powers of the World Wide Web. They quickly took the domain http://www.mit.edu before the MIT administration had thought to do so. Their interest in technology as well as their interest in motivating others to be interested in technology would fuel the motivation for many of their projects, including the SIPB web server. The webpage provided information about MIT, links to MIT student homepages, and provided general information about SIPB.[39] SIPB agreed in 1999 to mirror the content of http://www.mit.edu with the official MIT webpage http://web.mit.edu.[40] In general Internet users would assume the official page would reside at www.mit.edu, and thus it made sense for the change to occur.

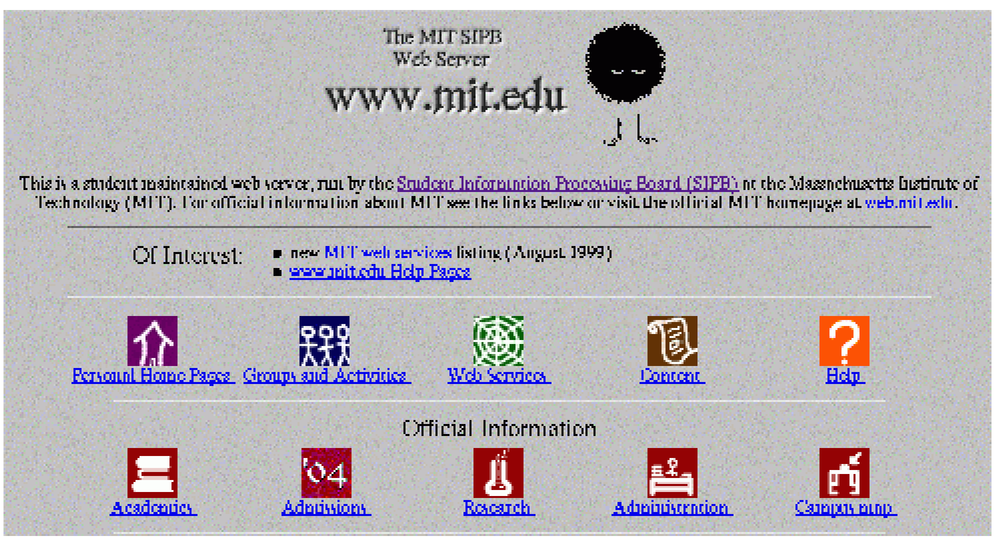


Figure 6: What Internet users saw when they expected official MIT homepage

The creation of Project Athena also affected the development of SIPB. Project Athena, started in 1983, had four initial goals: to develop computer-based learning tools that are usable in multiple educational environments, establish a base of knowledge for future decisions about

[39] MIIT SIPB Home Page [old].

educational computing, to create a computational environment supporting multiple hardware types, and to encourage the sharing of ideas, code, data and experience across MIT.[41] A lot of the goals of Project Athena overlapped with the goals of SIPB. The establishment of Athena as the centralized computing environment on campus altered the role of SIPB. Now the user pool included all members of the MIT community, and thus there was no need for SIPB to continue advocating for student access to computing resources. The changes in computer technology changed the goals of SIPB from the original visions of the co-founders. Now SIPB would focus on the other goal of making computers more useful, by improving the typical MIT user's computing experience. In essence, the goals migrated from expanding the user pool to include undergraduates to developing applications and providing services to enhance the user experience.

Fulfilling this goal involved developing applications that made Athena more user-friendly, and catered to the needs of Athena users. Some of these applications include Xcal (calendar), Xscreensaver (screen saver), Xmcd (clock), Fortune (fortune-telling program), EXMH (email application), and Zwgc (Zephyr Windowgram Client). XMH was the mail handler that was originally provided on Athena, and EXMH was developed by SIPB as an alternative for users. EXMH featured a nicer user interface, and new capabilities such as an address book and a search option. Zwgc is one of the most popular applications of Project Athena, and was initially developed by Mark Eichin, who was a SIPB member and an Athena student employee at the same time. Zwgc is responsible for receiving, formatting, and displaying zephyrs on output devices, and has the functionalities of different text colors, font styles, and cut and paste capability.[42] The Fortune program tells the user a fortune every time the command is typed at the prompt. These changes made the social communication through technical means more fun. With the growing popularity of Athena, students wanted to be able to run Athena from their dorm rooms, so SIPB members wrote a Linux installer for Pcs. The following is a SIPB Fortune program.

```
athena% fortune
Please ignore previous fortune.
```

[40] Benefiel, Anna.
[41] Bruce interview.

Applications such as EXMH and Zwgc play an important role in improving the social interactions that users have through technology. SIPB's development of convenient email and messaging programs are examples of the blurred distinction between technical innovation and social innovation. These applications enhance and improve social interactions between users, while at the same time are technical developments.

Other applications were not related solely to Athena, but still were geared towards enhancing the personal computing experience. Many of these projects were created due to individual interest. Steve Bellovin co-created a global message-board named USENET redistributes news to other universities on the Northeast. The Multics Forum was originally written by Jay Pattin ('83), J. Spencer Love ('77), and Jeff Schiller('79). Richard Tibbetts ('02) took the initiative to create virtual machine (VM) ware dialup and request tracker (RT) machine. By installing VM, the user can simultaneously run both Linux OS and Windows OS on computers, thus providing enormous flexibility to users. The RT program is useful in tracking software bugs as well as user requests. In addition, SIPB members also developed Linux and Windows based OS on Athena. Now, we see Dell machines running Linux OS in addtion to the usual Sun machines running Solaris and the SGI machines running Irix. The Windows-version Athena is code named Pismere and receives full backing of the MIT Information Systems, according to IS Vice President Jim Bruce.[43]

In addition to applications, SIPB also provides many useful services to enrich the MIT computing experience. The office door is always open for walk-in consulting, for students who run into problems on Athena, or who have general computing questions. They are generally very willing to help students at all hours of the night, and are a good source of computing expertise. They also write documentation and references for Athena software, such as "An Inessential Quick Reference to Athena," "How to Choose a Good Password," and "Inessential Zephyr."[44]

The development and direction of SIPB was altered and dictated not only by the emerging computer industry and Project Athena, but also by its eventual integration under the supervision of MIT IS (Information Systems) in 1983. Under the suggestion of then Director of Information Systems Bruce, SIPB became a part of the official computing group at MIT. IS would then fund SIPB (the 2001 annual budget was $38,000). Bruce thought that SIPB could be utilized to carry out the objectives of Athena and IS. The objectives of IS include providing

---

[42] SIPB Zephyr Class
[43] Bruce interview.

better service to people using computing, increasing bandwidth to desktop computing, and building better communication paths to clients to let them know what services IS provides and where to ask for help. SIPB agreed in part due to the overlap of goals between SIPB and IS. SIPB would also continue to be the student's voice for computing by serving as IS's link to the student body. "The SIPB office has become a portal to the student body for IS,"[29] according to Bruce. The presence of SIPB members among the different parts of the MIT computing environment has distributed its social reach on campus. According to a report from SIPB to IS, "SIPB members work in various laboratories and departments around campus, providing an informal network of contacts between independent groups of computer users."[45]

Another reason for incorporating SIPB into IS is for recruiting purposes. A lot of the members of SIPB go on to become employees of IS, Athena student employees, or are simultaneously involved in both groups at once. According to one Athena student employee Calista Tait ('99), "I became an [IS employee] long ago because of having friends who were, and it paying better. Those friends happened to be the same friends that pulled me into SIPB. So I would say I became a watchmaker and a SIPB member for the same reason. The friends I'd made and their involvement."[46] According to full-time IS employee Greg Hudson, " I was motivated to join IS because I had become attached to the environment as a project to work on, both because I'd been a watchmaker for a couple of years and because I had been working on the SIPB-Athena ports. Being a member of SIPB influenced both the decision to become a watchmaker and the decision to become a staff member."[47] The members of SIPB sharing common interest in technology, and making computers useful binds their culture together.

Technology and the SIPB culture

Since its founding, the group has developed a prominent culture that reflects its common interest in technology. The group became known as a collective place for people who shared interests in computers ranging from undergraduates to graduates to alumni. At their meetings every Monday at 7:30pm, there are many people from many different age groups. The skillset

---

[44] SIPB Documents
[29] Bruce interview.
[45] SIPB 10-year report to IS
[46] Tait zephyr correspondence.
[47] Hudson zephyr correspondence.

varies from experts who were the original designers of Athena, to people who join the group to learn new skills. The motivations also vary from people who would go on to pursue MIT student computing as a career by becoming IS employees, to people who just think of computing as a hobby. According to the current Chairman Liana Lareau, "The SIPB culture is mostly based on spending time in the office (or, true to the stereotype, at Mary Chung's enjoying peking ravioli) either hacking on projects or discussing anything from current events (privacy and security are common concerns) to the history of computing.'[30] This statement by Lareau gives us an understanding of the close comraderie between SIPB members. Technology does bind the culture together, but is not the only glue that holds the group together. SIPB members enjoy each other's company. This is evident by the significant time spent socializing in the office, going out to eat together, and playing frisbee together.[48]

Figure 7: SIPB members at daily meeting.

New prospective members are drawn by the common interest in computing. Since its founding, SIPB has accumulated 216 members, and has currently 20 active voting members. SIPB members are distinguished by voting and non-voting status, but according to Tait, membership is forever.[49] SIPB alumni maintain the right to their MIT Athena account, and many of them are active on the SIPB zephyr class instance.

Although SIPB submits publications for the freshman activity mailing and also participates in the activities midway during orientation, the group still relies heavily on students'

[30] Lareau email correspondence. October 23, 2001.
[48] Various SIPB Minutes.
[49] Tait zephyr correspondence.

individual interest in computing when selecting new members.[50] A student who is interested in joining SIPB is called a "prospective," and he or she must have the initiative and interest in the student group to hang out in the office, get to know SIPB members, and complete a project before he or she is voted on to be a full member. "The most effective way of becoming a SIPB member is to become a prospective and hang around the office helping out so much that the members forget you aren't a member and proclaim, 'WHAT? You're not a member? We'll have to fix that!"[51] After a prospective has spent a significant amount of time in the office, they are voted on during one of the weekly meetings. Members ask the prospective various questions about their involvement with SIPB. The questions will often be playful and of a joking sort. The following shows an excerpt from the meeting minutes when Emily Marcus <emarcus> was elected into membership.

```
   golem: Name three office heads and their IP addresses.
   emarcus: I don't know IP addresses.
   golem: Neither do I.
      ...
   fyfer: If you become a member, what would you encourage
           prospectives to do?
   emarcus: Documentation and tours again.  I would encourage
           prospectives to be in the office and come to meetings
           before they do a project.
   gif: Name two things that have previously hung from the
ceiling.
   emarcus: A slinky and bert's hair.
      ...
   jweiss: Why do you want to be a SIPB member?
   emarcus: I don't like being locked out.  I'd like to be here
      for more meetings and have a better sense of what's
going on.  I sometimes punt meetings as a prospective
if there's an election.
   <13-0-0, emarcus is a member>
```

The playful personality of their members is not only present in their questioning of prospective members. Their spontenaity and lightheartedness is evident in the various jokes and musings recorded during their weekly meetings, in a section of the minutes entitled "Other Other." "Other Other," in contrast to just plain "Other," is for non-SIPB related musings and

[50] Lareau email correspondence. December 9, 2001.
[51] SIPB Office Manual.

comments, whereas "Other" is for SIPB-related comments.[52] These comments give us a glimpse of the thoughts and humor of SIPB members. Many of the remarks made during "Other Other" are technology-related, but most are simply funny or interesting observations. A sampling of these remarks are shown below. The members of SIPB clearly not only share their interest in computing technology, but also enjoy each other's company and humor.

```
keithw: Arun, Yak, and I went to the cluster Saturday to help
        celebrate the giga-second.  There was general cheering
        from us but other people didn't participate much.

golem: I got a sexy new National Semiconductor tshirt.

sly: Screaming children should be in their own car on the
     commuter rail.

jhawk: Taxes are due tomorrow.
jmorzins: Being so poor that you get money back from the
            government is fun.

nathanw: Yet another dot-com is advertising in the men's room
      urinals.
```

SIPB has also become known for its hacks. A hack is a clever, benign, "ethical" prank or practical joke, which is both challenging for the perpetrators and amusing to the MIT community.[53] One well-known hack was referred to as the "Great Athena Fuzzball Hack of 1989."[54] The fuzzball bitmap was originally in the Public directory of an MIT undergraduate Anne LaVin ('85),and she also used it as her Athena screensaver icon. It was noticed by SIPB members, and much to her surprise and amusement, became the unofficial mascot of SIPB. Before http://www.mit.edu was taken over by MIT, the SIPB fuzzball was the prominent icon to visitors of the page. In December 1989, users at over 200 public workstations were greeted by the SIPB fuzzball instead of the Athena owl at the login screen. Athena signs around the Institute were also modified to display the grumpy fuzzball . Around 8am Monday morning to 4am Tuesday morning, the Athena staff examined the code responsible for the hack, and determined that the perpetrators had hacked the workstations by hand. The hack was deemed

[52] SIPB Office Manual.
[53] MIT IHTFP Hack Gallery.
[54] MIT IHTFP Hack Gallery: Grumpy Fuzzball.

harmless, and the Athena owl was returned to the login screen at 4am Tuesday morning. Many students commented the fuzzball resembled a burnt-out owl and thought it was fitting to the final week of classes. This hack was technical in nature in that it required the technical expertise to hack into Athena and social because it was humorous and amusing to the MIT community. Even through small hack, SIPB's technological effects are evident.

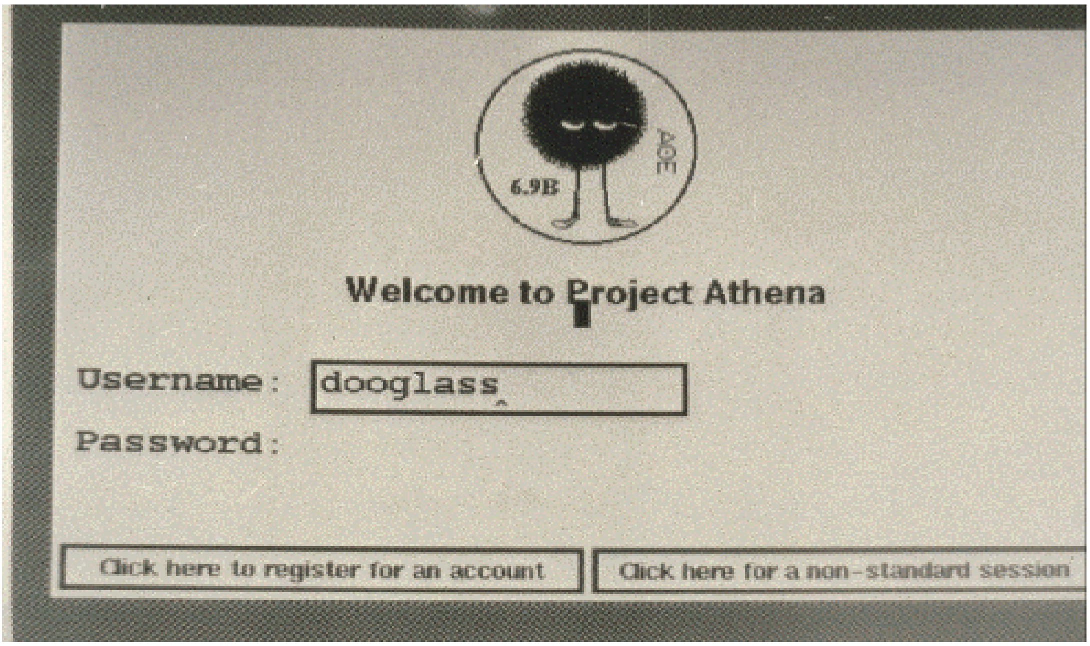


Figure 8: Great Athena Fuzzball Hack of 1989

The DVD decryption incident mentioned earlier also upholds the hacker ethic. The incident exemplifies one of the major beliefs of SIPB members that information should be exchanged freely. General sentiment of the members is that this philosophy is more a "good idea" than an official SIPB policy.[55] "Eventually a philosophy emerged from MIT known as the Hacker Ethic. The one and all-holy central tenet was this: information should be free. Hackers believed in free information the way hippies believed in free love."[56] Thus most of their applications are open source, and accessible for curious MIT users in the SIPB locker /mit/sipb/bin.

SIPB members also show their interest in technology by making it an integral part of their lives. During our phone interview with Frankston, he tried to utilize and incorporate

[55] SIPB Zephyr Class.
[56] Cyberpunk Archives.

technology to the fullest--he persuaded us to log on and chat with him using an online messaging program and was disappointed when he found out that we did not have a webcam. The simultaneous video, text and voice interaction was clearly natural to Frankston's lifestyle. Frankston also revealed to us that he created a searchable database of the documents and items in his garage, a further testament to the integration of technology in his life. SIPB members have had technical-related license plates saying "MULTIX," "FOOBAR," "USENIX", "DISCUSS," "AMBAR1," which are currently proudly displayed in the SIPB office. These are all examples which reflect the role of technology in the everyday life of SIPB members.

Figure 9: License Plates in the SIPB Office.[57]

SIPB culture is unique and fun, while at the same time reflecting the members' collective interest in technology. From the jokes and musings shared in the office to hacks and the general lifestyle of SIPB members, the influence of technology is evident.

Shortcomings of SIPB

Despite the many accomplishments of SIPB since its founding 32 years ago, there are ways in which SIPB is not fulfilling their goals.

In their Constitution, the SIPB Statement of Purpose includes these goals:[58]

1. to provide students with improved access to computational facilities
2. to serve as an advocate for the MIT community to the faculty and the administration in computer-related topics
3. to function as an information center for members of the MIT Community interested in topics in computer science or computing
4. to conduct, manage and support experiments run by students to advance the techniques of

[57] Courtesy of SIPB Office.
[58] Constitution of the SIPB.

organization, planning, and administration of access to computers

5. to initiate, develop, and provide computer services deemed valuable to the MIT community

According to our research, SIPB has successfully accomplished and followed through with most of their goals, namely providing students with improved access by increasing computing resources, functioning as a source of computing information with their documentation and walk-in consulting, supporting student projects, and providing valuable computer services. However, it is not clear that they fulfill the goal of being "an advocate for the MIT Community to the faculty and administration." SIPB admits that there are many common misconceptions about them. These include the notion that SIPB is simply a helpdesk and not a developer of new applications, and that SIPB is an elitist computer group, contributing a lack of understanding about their identity. Additionally, SIPB acknowledges that many members of the MIT community do not even know that SIPB exists.[59] If the MIT community that SIPB professes to represent does not know of its existence and holds these misconceptions, then it is doubtful that SIPB actually advocates the needs of the MIT community. Although SIPB regards themselves to be the voice of the MIT community about computing, they somehow need to address and resolve these discrepancies in order to truly be the community's voice.

Even though SIPB services and actions address most of the goals outlined in their constitution, SIPB could do more to truly provide for the entire MIT community. Currently, SIPB depends too much on the individual students' motivation to be interested in computing; students without UNIX experience must find documentation on their own or sign up for Athena minicourses to learn how to use Athena, and those who are familiar with UNIX must take the initiative to explore the SIPB locker to discover useful Athena applications. Those students who lack this motivation are not experiencing the social and technical usefulness of computing that SIPB is trying to promote. SIPB does "initiate, develop, and provide" computer services, but neglects to motivate the interest in computing to those who do not already possess it.

[59] Lareau email correspondence. 9 December 2001.

Where Have They Gone?

The importance of technology and the role of SIPB in the lives of its members does not end with graduation from MIT. Many of the members of SIPB have gone on to careers in computer science and computing, for example, the co-founders. Fox became a Professor of Computer Science at Virginia Polytechnic Institute and Frankston implemented the first spreadsheet application, VisiCalc. Other more recently graduated members, such as Tait and Jered Floyd have worked for computing-related companies.

The network of past and present SIPB members is sustained with the help of technology. SIPB members enjoy full-access Athena accounts for life and thus are able to easily interact with current members through the SIPB mailing list as well as the SIPB zephyr class. Computer interaction, however, is not the only way that SIPB members keep in touch. In 1994 SIPB held its 25-year reunion. In attendance were members dating all the way back to SIPB's founding as well as their most newly elected members.

Figure 10: SIPB 25th year reunion[60]

This network of people built since 1969 have affected social and technical change in the computing environment at MIT.

Conclusion

SIPB brought about technological change in the state of MIT computing ever since its founding in 1969. Prior to 1969, MIT recognized the need for student computing in an educational setting and established the Computation Center, which allowed for research opportunities that had not previously been possible. These opportunities, however, were available only to the limited user pool of graduate students and faculty that could obtain limited computer time. The development of time-sharing systems allowed the user pool to be expanded, but still limited to graduate students and faculty. Nevertheless, time-sharing systems were the

[60] Courtesy of SIPB Office.

technology that changed the manner in which users interacted with computers.

The advent of time-sharing helped to satiate the existing demand for computer time, allowing SIPB to advocate undergraduate access to computing resources. The founding members of SIPB recognized the educational benefits of computing and the need to expand the user pool of interactive computing to include undergraduates. Even though SIPB did not create the technology of time-sharing, they were able to bring about technological change by making computing resources accessible to a new group of users.

Along the stages of its development, SIPB needed to adapt to new computing technology by shifting their focus from increasing student access to promoting the usefulness of computers. Computing technology in the 70's was evolving from large mainframe computers to dispersed workstations, and MIT's support of the changing technology made it easier for SIPB to obtain more resources for more students in the form of workstations scattered around campus. In 1983, MIT's Project Athena opened computing to all of the MIT community, thus eliminating need for SIPB to continue their initial cause. SIPB's new focus became to improve the typical MIT user's computing experience. Applications such as Xzewd and EXMH enhanced and improved social interactions through technical means, and are examples of the technological impact of SIPB on the MIT computing environment.

Common interest in technology is a characteristic that binds all SIPB members together. The network that is formed by this common interest ties together SIPB alumni, current members and new prospectives. The culture that emerged from interest in technology is often technology-related, once again showing that social and technical factors cannot be easily distinguished.

From its founding days until the present, SIPB worked towards social and technical change in the area of computing at MIT. The technological accomplishments included expanding the user pool of MIT mainframe computing facilities and enhancing the user's computational experience on Athena. The lasting effect of SIPB's work is evident on campus. Many students use SIPB applications in their personal Athena customizations, and compared with the three mainframe computers on campus in 1983, MIT is now equipped with more than 30,000 network devices and 300 servers on campus. According to Bruce, "This development and success is due in part to SIPB. It is the foundation of SIPB that changed dramatically how MIT and IS view computing on campus."[27]

[27] Bruce interview.

Sources

Abelson, Hal, Ed. and Simson Garfinkel. Architects of the Information Society: Thirty Five Years of the Laboratory for Computer Science at MIT. Cambridge, MA: MIT Press, 1999.

"Application Form for Computer Usage." 10. April 1957

Benefiel, Anna. "SIPB turns over Website to Institute." The Tech, Vol. 119, Number 34. August 28, 1999. <http://the-tech.mit.edu/V119/N34/34athena.34n.html>.

Bruce, James. Personal Interview. 27 November 2001.

Burmaster, Dave. Personal Interview. 19 November 2001.

Christensen, Clayton M. The Innovator's Dilemma. Boston, MA: Harvard Business School Press, 1997.

Clingen, C. T., Fernando J. Corbato, Jerry H. Saltzer. "Mutics-The First Seven Years." <http://web.mit.edu/Saltzer/www/publications/f7y/f7y.html>.

The Constitution of the Student Information Processing Board of the Undergraduate Association of the Massachusetts Institute of Technology. December 1999. 5 December 2001 <http://web.mit.edu/afs/sipb/admin/text/constitution/constitution.ps>

Corbato, Fernando J. Personal Interview. 26. November 2001.

Cyberpunk Archives--hackers. 2 December 2001 <http://www.joabj.com/stories/hacker.html>.

Daggett, Marjorie Merwin, Robert C. Daley, and Fernando J. Corbato. "An experimental Time-Sharing System." SJCC, May 1962.

DVD Decryption Code. 2 December 2001. <http://www-2.cs.cmu.edu/~dst/DeCSS/Gallery/qrpff.pl>.

EDUCOMM: "EDUCOMM Report to MIT," July 1980.

Frankston, Bob. Bob Frankston's Web Page. 2 December 2001 <http://www.frankston.com>.

Frankston, Robert. Personal Interview. 24. October 2001.

Frankston, Bob. "RE: 6.933 SIPB Questions." Email to the author. 22 October 2001.

Fox, Ed. "RE: 6.933 SIPB Questions." Email to the author. 22 October 2001.

Greenia, Mark W. Computers and Computing: A Chronology Of the Men and Machines that Made Computer History. Sacramento, CA: Lexikon Services Publishers, 1990.

Gallery of CSS Descramblers. 2 December 2001. <http://www-2.cs.cmu.edu/~dst/DeCSS/Gallery/>.

Hudson, Greg. SIPB Zephyr class correspondence. 11 December 2001.

IBM Archives. 30 November 2001 . <http://www-1.ibm.com/ibm/history/reference/glossary_7.html>.

John McCarthy. MIT. 11 December 2001 <http://www.digitalcentury.com/encyclo/update/mccarthy.html>.

Lareau, Liana. "Re: Question." Email to the author. 9 December 2001.

Lareau, Liana. "Re: 6.933 Interview Questions about SIPB." Email to the author. 23 October 2001.

Lareau, Liana. Personal Interview. 29 October 2001.

Latour, Bruno. Science in action : how to follow scientists and engineers through society. Cambridge, MA: Harvard University Press, 1987.

Lavin, Anne. "The Fuzzball." MIT. 2 December 2001 <http://www.mit.edu:8001/fuzzball.html>.

Levy, Steven. Hackers : heroes of the computer revolution. New York: Dell Publishing, 1984.

MacKenzie, Donald A. Inventing accuracy : a historical sociology of nuclear missile guidance. Cambridge, MA: MIT Press, 1990.

Martin, Nick. Personal Interview. 7 November 2001.

McCarthy, John. "Subject: A Time Sharing Operator Program for our Projected IBM 709." Memorandum to P.M. Morse Proposing Time Sharing. 1 January 1969.

MIT IHTFP Hack Gallery. MIT. 2 December 2001 <http://hacks.mit.edu>.

MIT IHTFP Hack Gallery: Grumpy Fuzzball. 11 December 1989. MIT. 2 December 2001 <http://hacks.mit.edu/Hacks/by_year/1989/grumpy_fuzzball>.

MIT SIPB Home Page [Old]. 2 December2001 <http://www.mit.edu/sipb-homepage.html>.

Moreau, R. The Computer Comes of Age: the People, the Hardware, and the Software. MIT Press: Cambridge, MA, 1981.

Morse, Philip M. Memorandum to Dr. C.R. DeCarlo of IBM Corporation. 10 February 1960.

Morse, Philip M. Memorandum to MIT Community. 4 March 1957.

Moses, Joel et al. "Report of the Committee on the Computing Environment." 2. April 1984.

Multicians Website. 29 October 2001. <www.multicians.org>.

SIPB Documents. 5 December 2001. <http://www.mit.edu/sipb/docs.html>.

SIPB Office Manual2. MIT. 15 October 1998. 2 December 2001. <http://web.mit.edu/afs/sipb.mit.edu/project/doc/office-manual/HTML/manual/>.

SIPB Members and Prospective. 5 December 2001. <http://web.mit.edu/afs/sipb/admin/text/members/members_and_prospectives>

SIPB Minutes. 7 November 2001. <http://www.mit.edu/afs/sipb/admin/minutes/HISTORY/>.

SIPB Ten-Year Report to IS (1982-1992). June 1993.

Tait, Calista. SIPB class zephyr correspondence. 11 December 2001.

Teager, Herbert M. "MIT Computation Growth." 15 January 1960.

Teager, Herb. "Summary of 7094 Computer Usage during period Jan 1965-March 1965." May 1965.

Turkle, Sherry. Life on the Screen: Identity in the Age of the Internet. New York: Simon & Schuster, 1995.

Waldrop, Mitchell. "The Dream Machine: J.C.R. Licklider and the Revolution That Made Computing Personal." New York: Penguin Group, 2001.