THE TECHNICAL INFORMATION PROJECT OF THE MASSACHUSETTS INSTITUTE OF TECHNOLOGY*

Project MAC at MIT has been mentioned as one of the newer developments in the computer art. In Project MAC, a man in Michigan can sit at a teletype machine and interrogate a computer, or in general behave as if the computer were next door to him. He can program, compute, or do what programmers call debugging or cleaning up a program. He can also, as of a few months ago, type a request such as "Compile a recent bibliography on laser physics" or "What's new in plasma physics?" It is this application of Project MAC that will be described here.

When the man in Michigan formulates his question, when he says "I want the latest bibliography on lasers," he does not get just a printout of a previously compiled bibliography. He gets the results of the actual putting together of a bibliography based on thousands and tens of thousands of documents in the computer's store, since it is a dynamic system. This then is a function that has been grafted on the MAC computer system at MIT, and it is necessary to understand the technological environment of Project MAC.

Project MAC was originally a very ambitious notion; some of the more venturesome people at MIT thought of it as "Machine Aided Cognition," which is a very revolutionary notion, but it is more realistically described as "Multiple Access Computer." TIP, the other acronym which will be used here, stands for Technical Information Program; it is that part of MAC which is concerned with non-computational uses of the computer, particularly for the manipulation, storage, and retrieval of all manner of technical information.

---

*The following paper is based on the tape of an unscripted talk given by M. M. Kessler (Associate Director of Libraries, and Director of the Technical Information Project at the Massachusetts Institute of Technology). Mr. Kessler has kindly granted permission for the paper to be printed in the version below, with the proviso that it represents the editor's understanding of his remarks; neither Mr. Kessler nor M.I.T. is to be held responsible for the precise form in which they appear here.

The essential idea of MAC is that a computer will be utilized as is a public utility. Just as there is not a little dynamo for each room or each building, so a computer for each department or each project is not needed. One central computer can serve the entire community. Now whether that community is the nation, or a region, or a university is not quite clear. It most certainly is not the nation, but it most certainly is not just one company either. At this time we might think of a "public utility" computer as being of the magnitude that might very well serve a region. Now obviously, in order for that to be realized there must be a remote access facility. And so the remote access idea is very much a part of the MAC thinking, and it means that you can approach the computer and use it, without any limitations, from a remote location. As of this time there are two possibilities. There is the telephone wire approach which is cheap and ubiquitous; all you have to do is attach a proper piece of equipment at the end of a telephone, which at this time is a standard teletype machine. A more sophisticated approach is to use a more broad band type of communication, such as television and data transfer circuits; but for any sort of human interaction with the computer, the telephone system is fast enough, accurate enough, and available enough so that we need think only in terms of telephone connections to the computer. And so the approach in and out of the computer, or the input - output machinery so to speak, is a standard teletype machine connected to the computer by means of telephone wire. Obviously remote access is a very important aspect of MAC.

The other important aspect of the public utility notion of a computer is time-sharing. What it means essentially is that people do not line up serially to use the computer, so that the tenth person has to wait until the ninth is finished, and the ninth has to wait until the eighth is finished, and so on. Instead there is a long trough, and they are all using it at the same time. MAC can accommodate thirty users at the same time; it can be time-shared by thirty people. Strictly speaking they do not time-share it, and they do use it in rotation. However the rotation is so fast that it is a matter of seconds, and the printout absorbs the cycle time because the computer works much faster than either a man can type or the teletype can type back to him. This means that, while one person is typing, if he hesitates at all or if his attention wanders, somebody else is using the computer. The internal machinery is such that the computer remembers the state of the last operation, and when it comes back to that operation, it picks up and continues another few milliseconds, and so on. It is not literally simultaneous, but virtually so. There are plans to increase this facility from thirty time-shared users to possibly one hundred and fifty.

Besides the notion of it being remote and of it being time-shared, there is the concept of being on line. And by "on line" is

meant that a dialogue can take place between the person and the machine. That is, you do not bring in the work, give it to an operator, and then come back the next day to find that one of your cards was misplaced. When you are on line, if you make a mistake the computer immediately tells you that it is a mistake, and if you want to correct something you can do it immediately. There is a back and forth real time interaction between the person and the computer.

A subtle psychological factor is involved here. When you work with a batch processing system that is not on line, and you bring some work in and are told to come back the next day, and you come back then and it is ready, you are very happy indeed that it is ready twenty-four hours later. But if you sit at a time-shared console and the answer comes back two minutes later instead of two or three seconds later, there is a great deal of impatience and annoyance. You don't mind waiting twenty-four hours, but you do mind waiting two minutes instead of two seconds. This is not entirely irrational because if you know that the answer is coming back in twenty-four hours you can do other things, but if you expect it in two seconds, and it comes back in two minutes, you just sit there and stew. This is an interesting human response, and there is apparently a critical time delay.

There are many other such things involved, of which this is an example, because we are dealing here perhaps for the first time with a real interactive process between man and an intelligent-like machine. At least it is an interaction between a man and a machine by way of man's intelligence. You are sometimes tempted to kick at your car or radio and take an anthropomorphic attitude towards them. This response is even more understandable where the machinery simulates some intelligence, and you expect more from it than it actually gives you; the psychological strain can sometimes be considerable when you realize that it is a poor dumb beast, and not even that but a dumb machine. This aspect of engineering or of research is extremely important in our type of application because we are thinking of using the computer in an intellectual manipulation and we really do not know what the answer to the problem will be. It looks pretty good if the machine can be made much bigger, much faster, and much cheaper. This is very important because if the machine is not cheap, you resent its not only wasting your time but wasting $5 every minute or so, and that gets to be a very serious resentment. These are very important considerations, and they do color the use of the machine and its application in this area.

Fortunately, the computer people tell us that computer technology can only go one way; it will all become faster, cheaper, and smaller. It does seem that things are moving in this direction, so that we can look forward to these problems being solved. However from the beginning of computer technology (which is not more than

ten to fifteen years old) people have expected great things from the computer in the way of library and information application. As of today, if we froze the technology and took a snapshot of the situation, we must admit that the application or the contribution of computers to our type of problem has been rather trivial. Here and there a list is made up, or a little bookkeeping is done, or a circulation system is handled, but on the whole the computer has not yet made its impact on library and information sciences. However, the inception of the computer industry was only about twelve or fifteen years ago at the most, and the next ten or twelve years will really show a difference.

To proceed with the description of the MAC system, there is a computer, and the computer consists of two parts, the processing unit (with circuitry to do all the various computer manipulations), and the memory unit where all manner of information is stored. In the MAC computer, the memory is divided into several parts. There is first of all a part of the memory that has to do with the internal workings of the computer, e.g., the addition, subtraction, or whatever else you want to do with it; this concerns only the computer engineers, the maintenance people, and those whose responsibility it is to improve and develop the system. This is not visible to anybody; as far as the user is concerned, it is a black box that is closed, and he need not know what is inside. There is another part of the computer memory that contains within it various facilities which are available to the user. These involve the various computer languages, such as Fortran, MAD, or whichever it is, and this part may contain worked-out programs, so that if you want to perform a given operation you do not have to develop the program, you just call for it. This then is a library of publicly available routines, programs, and data.

The rest of the memory is divided into many compartments. Each compartment is accessible to only one user or subscriber, and out of each of these, schematically and not to dwell on the actual circuitry, come one hundred and fifty telephone connections, each going to a console, to a teletype machine. Thus there are one hundred and fifty locations where teletype machines are installed, and that location in Michigan is one of them. There are in fact about twenty or so scattered over the country; the rest are around MIT. Each user has at his disposal a common library of programs, languages, facilities, and so on. He also has at his disposal a private little library, his own office library, which he can put things into, take things out of, work on, and do all kinds of things that are available to him only. Within this range of memory, the strictest privacy is maintained. Great care has been taken so that user A can in no way get to user B's memory unless user B makes that facility available.

It is possible for any one of the users to take part of his private memory and declare it public, and that in every sense is a publication. As a matter of fact we have an editorial board and referees, and we

go through the entire procedure of publication before any user's private file is made available to others as a public facility. To give a routine example, suppose that somebody develops a new way of integrating an equation or a new way of solving a problem, and he thinks that it may be of use to others; he then submits it to the editorial board who examines it and approves of it for the public file. As soon as that is done, it is available to the entire community of users.

It is with regard to this sort of scheme that we have developed TIP which stands for Technical Information Program. A rather large slice of machine memory has been used for quite some time as our own private experimental slice, but about ten months ago it became sufficiently developed to be offered to the public. What used to be a private memory slice has now joined the public domain, and TIP is now available to all of the 150 consoles that have access to MAC. There are 150 consoles, but only 30 can use the computer simultaneously. In fact, since more than one person has access to any one of the 150 machines, there are some 500 people who at one time or another use Project MAC.

So much for the computer structure with which we work. Let us now look at TIP in more detail. When we first started thinking about this application, it was clear that we could not build an entire Library of Congress or MIT library system into a computer, turn the switch on, and then have it work. We had to have a model of the system, and the model had to be realistic in the following sense. First of all, it had to be scalable, so that if it worked, it could be scaled by a factor of 10 or 100; otherwise it would be a toy and not a model. The other requirement was that the model be big enough to be capable of functioning in a real environment and not be just an analytic type of model. If it was to be a model library, it had to be a big enough library so that people who want to use libraries would actually be motivated to come to this thing and use it. Even though they do not care about this as an area of research, they should care about this as a service. In other words the model had to have critical size. It had to be small enough so as not to tax the experimental facilities of the situation, and yet it had to be big enough not to be a toy but to be of serious interest to workers.

As a result, we had to limit our literature or holdings, and we picked physics as an area in which to work, more or less by chance because I am a physicist; but it was a happy choice because physics is a very well disciplined literature. More than that, we picked the journal literature of physics—no reports, no books, only physics journal articles. We process now twenty-seven journals, roughly 1200 articles per month. This corresponds to about 60 percent of the physics literature which ends up in Physics Abstracts. For each article in each journal as it comes in, we put into the computer memory the following information: what we call the identification, that is

the journal, volume, and page of the article; the title; the name of the author; the author's location or institutional connection; and whatever citations or bibliographical references are in the article either as footnotes or in the body of the text. There is no indexing or key word identification of any sort, and this is a calculated risk. One of the important considerations was that the system or model be scalable, and one of the most difficult things to scale upward is human intelligence. We wanted to see how far we could go with an input that is purely clerical. What we record for each article requires no judgment whatsoever, no assignment of key words, or indexing terms—it is purely clerical. Indeed this work is done by a girl who sits at the teletype machine (we do not use IBM cards) and types directly onto the computer memory the indicated information—identification, title, author, location, and bibliography.

The question of course is whether this is sufficient for a legitimate system. We went through a long series of experiments and we are satisfied now that it is sufficient, at least to begin with, and so we are not doing more. However the system is flexible, so that if at a later date we want to add other things, anything at all, we can do it by just typing in the identification data. The system is open ended, and if more information is needed, it can easily be put in.

This data then is on the computer memory disc in a format that is immediately available to any one of the 150 users. In other words, there is no loading of tapes or loading of cards; the information is there 24 hours a day, except during the times when the computer is off the air. It takes a bit of computer memory to do that but it is not prohibitive. This information is organized on the memory disc much as it would be organized on a library shelf. Think of twenty-seven journals, bound into volumes, and located all in one place; in our case we have files, and for this purpose each volume is a file. And within the file, within the volume, the information is organized by way of page numbers. This is the most primitive approach to file organization, and it is certainly not the optimal approach. It is now in the process of being changed to a more reasonable approach from the machine point of view because it is not the best approach for large scale searches. As our library gets larger, we will change this file structure, but as of now it is a serial file.

We then had to develop a set of words, a search language, because we set ourselves the design criterion that the people who want to use MAC for real purposes must not be asked to do programming. They are librarians, writers, working scientists—including physicists—but they are not programmers, and they are not to be asked to write their own programs. We had to develop a language that was close enough to English to make communication with the computer comfortable.

The language we have developed is a very comfortable sort of semi-English. What you do is to sit at the typewriter, log in, and identify yourself so that the MAC system accepts you as a legitimate user. You then type the word "TIP," which informs the computer that it is about to be used as a library and not as a computer. There are many other facilities which the computer has. For example, civil engineers have developed a road intersection program and bridge network program, and there are all kinds of biological and psychological programs; somehow the computer has to be informed of which aspect of its personality is about to be called into play, so to speak. When you type "TIP," that indicates that you are interested in the library part of the computer. The computer will come back and probably say "Ready," or some such thing; when it says "Wait," something is wrong. When it is ready, you might then say, "Search Physical Review, Volume 136 to 140," which is a typical search command. Or you might say, "Search Physical Review, latest issue," or "Search all latest," which means search the latest issue of all the twenty-seven journals. In other words, there is quite a variety of research statements you can make, each of which essentially means, "Take this designated literature off the shelf."

Then you type some request, like "Find title nuclear," which means "Find every title in this literature range that has the word nuclear in it." Here again there is a wide variety of possibilities. You can say "Find title nuclear energy," in other words you can use phrases, or you can say "Find nuclear energy," and in that case it will find every title which has the words "nuclear energy" regardless of their order, for example, "Production of energy in nuclear engines." If you want the words "nuclear energy" to be exactly as stated, you put an asterisk between them and this means that they must be joined. There is a wide variety of other manipulations; you can say "and/or but not," for example, "Find title nuclear but not spectroscopy." In other words, you want not the whole set of literature having to do with nuclear spectroscopy, but you do want other nuclear literature. You can say "Find nuclear and author such and such;" you can mix any of these things with the logical possibilities worked in. And of course you can find any one of these things separately. You can say "Find citation (and name the journal, volume, page)," and that means "Search this literature and find every article that cites this paper." Or you can say "Find author Smith," or any combination of these by way of "and/or but not," and so on.

By these directions you take the volume off the shelf, and look through it. Having found what you want, say all the articles with the word "nuclear" in their titles, what do you do with it? We have to make some sort of output statement, and a common form of output is printing. So you say "Print title, author, and page no." In other words, you direct the computer, having found all the papers with "nuclear"

in the title, to print the complete title, and the author, and whatever else you wish to ask for. You can also say "Save" because if you print, you erase what you have found as soon as it is printed out, but you may want to save the information for later work. Suppose you have discovered 230 articles with the word "nuclear" in their titles, you have created a new file, a new list. You have only to name it somehow if you wish to save it; so you say "Save file," and follow it by a name. For example, you might call it "Nuclear titles" or "List 1" or "Jones' favorite subject"; you must give it some name because you may want to come back to it later. Once you have done that, you can at another time say "Search" and instead of searching Physical Review, you can now search the file that you have made. This is a very important consideration.

The main point is that in this type of organization and language structure there is the possibility for searching by author, by any word in the title, or by what is known as citation index; all of these are available at any time and in many mixtures. You could do one, then the other, and so on and back and forth, and you can save that information and then look at it again later from another point of view. There are several safety features built in. For example, if you request an item that is not in the library, e.g., the Journal of Gestalt Psychology, the computer will say "The Journal of Gestalt Psychology is not in the TIP library." If you make mistakes in spelling and things of that sort, there are provisions for erasing and correcting them. If as you work, or as the computer works back at you, the telephone rings and you want to go away for a while, you can stop and then come back and type "Start," and it will start again where it left off, even if it is three hours later. We now have a graduate student working on a teaching program, so that the computer itself will teach people how to use the computer. This is of particular interest in a case where a non-programming population is involved. That is, we would like to provide for the situation in which a user logs in and says "I want to use TIP." He should know that much. Then the computer comes back and says, "Do you know how to do it or do you not?" If he says "No," a teaching program comes up point by point with illustrations. This, of course, is very closely related to the whole man-machine problem.

A second very useful search technique has been developed, something that we call bibliographic coupling. Let us say there are two or three articles which I know I am interested in. They may be my own papers, or they may be the papers of a friend who is an expert in lasers, and I know that I am interested in his work. Now I want to be able to conduct a literature search and find others like it. This can be done, provided that you identify the criterion of likeness. You may say to the computer "Find others like it," and by that you mean find other papers which share something with this paper. They may share a variety of things; they may share common words in the

title, or they may share the fact that they were all produced in or came from the same laboratory. These are useful but rather limited applications. What we found to be extremely useful is to say "Find other papers like these" where the criterion of likeness is the bibliography, or the list of citations in a given paper. Then we are saying in effect, "Examine these two papers that I know are of interest to me; observe the citations in these papers, then examine some given range of literature and find other papers that have similar items in their citations." We call this "bibliographic coupling," because the resultant papers are coupled not with words, but by virtue of sharing certain references. For example, if I write a paper with twenty items in my bibliography, and somebody else in Japan writes a paper with twenty items in his bibliography, and if it happens that ten are the same in both papers; then the probability is very high that these papers are related. We have done many hundreds of such experiments and find this to be an extremely powerful search tool. So the share group of programs, added to the find group of programs described previously, is very important. We can say "Share title" with the named paper, or "Share author" and so on; but of all those shared programs, the shared bibliography is the most potent search tool.

In the hierarchy of search procedures, we have the find list of programs and the share list of programs. Beyond that we have another even more sophisticated search procedure. (By the way, the number of papers on disc in the TIP library is now about 35,000 and growing at the rate of 1200 per month; this is pretty close to four years of literature.) Let us say we want to produce a bibliography on lasers. There are various strategies one can use. You can say "Search all," that means "Search the entire library," or you can say "Search 1965 papers" or anything of that sort.

Then you can say "Find title containing the word laser," and if you put a plus in front of laser, the computer will find anything where the suffix is laser, e.g., super-laser or anything of that sort; and sometimes it is convenient to do this. Then for output, you can say "Print and save," and call the saved file "Laser titles." In a typical case, this will result in about 230 papers. These are 230 papers out of those 35,000 which have the word laser in the title.

This by no means exhausts the laser literature obviously, because there are many papers that have to do with "lasers" which do not have the word laser in the title. But certainly those that do have "laser" in the title are important, and we have now compiled and saved this first list of all papers that have the word "laser" in their title. The next step is to say "Search this new list of laser titles"; we do not search the entire literature of 35,000 now, we search only 230 papers. Actually the direction is "Read laser titles," by which is meant that the computer is asked to list all the references in these papers in the order of their frequency of occurrence; out of 230 papers, there will

be perhaps 3000 references—slightly more than 10 per paper. Most of them of course, are referred to once and never again. Some are referred to twice, and so on; these 3000 references then will be listed in the order of their frequency of appearance in this group of papers, and we end up with a list that says "Paper 1 in this list of references appeared 17 times"; "Paper 2, 15 times," and so on. To do all this will take 5 or 6 seconds.

Of course, one might take just a slice of this and ask for a list of all papers cited more than five times, and go look at those. But what is even more interesting is to say "Let us go back to the literature, search all of it again, and find all of those papers that do not have "laser" in the title, but nevertheless cite these same references." For example, someone may have used a synonym for laser, namely optical maser, or some other term. This second complete search then gives you another sample of relevant articles. You look at this new set, and ask, "Here is a set of papers concerned with lasers, and the coupling is very good, although they do not have the word laser in their titles. What are the most frequent words in their titles?" In other words, "What are the most frequent and common actual substitutes for the word laser?" Perhaps you come out with two or three terms, such as optical maser or optical amplifier; then you go back through the literature, and search all titles for those words. This goes on back and forth, and each time you form a new list; there are in fact list merging programs that say "Go back to the original list, and if this new article is not there, add it; if it is a duplication, do not." The list which started with 230 articles may add 50 in the next process, and 30 in the next; you continue searching as long as you wish, until you no longer add anything. Then you must be at the end.

This sort of procedure lends itself to many different kinds of manipulation, and we call them strategies. And this particular strategy just described, in which you start with a word and then go to frequency distribution of references, and back to the word and so on, is called Strategy A (since it was the first strategy which worked well for us). You can sit at the teletype machine now and say "Execute Strategy A on lasers;" you do not have to go through all of the steps just described, and the computer will proceed to perform each step in the whole process. This is a high order of programming which is extremely potent.

The system is now in use by something like 75 people per week. In other words, 75 people log in and ask for TIP to become available to them. We have written a monitor system, so that everybody who is using the system gets recorded as who he is, how long he is using it, what questions he asks and so on; and we are collecting user experience now through this monitor program. It is also being used by librarians in a much more self-conscious way. The system has been

in use about six months in this public fashion; we have had it in use experimentally for over a year now, but it was only about six months ago that it went public, so to speak.