ACME NOTES

Gio Wiederhold

November 11, 1966

A Summary of the ACME System
(Presentation given at the ONR Computer and Psychobiology Conference May 17, 1966 at the U.S. Navy Postgraduate School, Monterey, California)

The ACME computer system at the Stanford Medical School is designed to provide powerful computing to research laboratories in the Medical School. The type of computation service planned is regular batch processing (mainly at night) and real time interactive capability for on line experiments. The remainder of this description will concern itself with the implementation of this latter facility.

In order to accommodate all the laboratories in the Medical School with their widely varying data rates, the system is designed to share the available computer time. The amount of time allotted to a user, however, is not one fixed unit per period, i.e. 200 ms every 10 seconds, but rather the time required by him to acquire and process one data point.

The data can be obtained in a number of different ways (see appendix CN-2, SK-2):

Data may be typed on a typewriter station. All textual and programming information is entered this way.

Slow to medium speed (up to 1000 samples/second) analog signals can be entered via a subsidiary computer which can either scan input voltages at the 1000/second rate or respond to separate interrupt signals given by the experiment or experimenter. The precision of the conversion can be up to 14 bits (.02%) precise. This computer will convert, reformat and preprocess these data and then transmit them to the main computer.

Digital input up to 16 bits wide can be handled in a manner similar to the analog input with the subsidiary computer, in order to serve users that have their own digitizing equipment or whose experiments actually generate digital output.

Users of the system with much higher rate demands, especially those that currently have small computers installed, will be able to connect directly to the main computer via a high speed (up to 125,000 samples per second) parallel data link. In this case they may program the interactive aspects of their experiments themselves, but have the resultant data processed on a larger machine.

In addition IBM research is devloping a device to allow medium to high speed response demand data acquisition via multiflexor channels.

Revision of AS-1 dated May 26, 1966

The IBM 360 Model 50 which we are using as the main computer has been chosen with a configuration that will support this type of multiple user activity. Its main (core) memory size is one million bytes or characters, or 250,000 words or values. The backup storage is an IBM pie file, which stores data on strips of magnetic tape, 2000 of them, which are all individually retrievable within 0.6 seconds. It has a total capacity of 400 million bytes. The subsidiary computer is an IBM 1800 process control computer, connected via a special direct channel to the main computer.

Results may be listed on the system printer, of course, but the emphasis will be on typing out the results on the typewriters in the laboratories.

Data may also be returned in analog or digital form via the 1800 at rates comparable to those of the input capability; generally within a few seconds after the results have been produced inside the Model 50. The digital lines may be used to drive plotters producing graphical summaries of the experiments. Display equipment of various types can be installed in the laboratories and be driven from the analog or digital 1800 outputs.

The small computers can also be used to distribute output via their typewriter, display tubes and plotters, or they may use the results to automatically control the continuation of the experiments.

It is quite obvious that a system of the described scope is not supported by any computer manufacturer. We have therefore designed and written a simple but complete support package including an interactive compiler, a supervisory system, input-output procedures and data acquisition and distribution routines. The system design is such that continuous guidance is provided to the user via the typewriter (see ACME note RC-1 and appendix RU-1).

To make such an approach at all economically possible, we are programming using an IBM compiler (FORTRAN H) to allow us to write the system within the allotted time span. This should also make it possible to later share the results of our software development work with others. The language that our system will compile and which we hope will be acceptable to our non-computer specialist users is a subset of PL-1, a FORTRAN like procedural language, defined by IBM for use on 360 systems. Thus we expect that procedures checked and proven on our interactive compiler can be filed and used on our and other computer libraries under standard systems (see ACME note on PL-1: PL-1).

The input-output system will include file handling and retrieval facilities where all data filed will be automatically labelled with all pertinent information to optimize the usefulness of the collected information. No elaborate buffering procedures will be used to minimize a user's inactive time; the time that he cannot use for computation will be turned over to the next user in the queue.

Since for some experiments the system reaction time will be quite critical, we will have to limit the number of these users in a given period. An attempt to utilize every available computer cycle for this work can only result in system overloading and failure. However, a number of users with non-critical problems, routine processing or information retrieval can balance the system to achieve reasonable total utilization, and we plan also to provide facilities for this type of use as soon as the more critical uses are satisfied.

The project currently uses several different computers around Stanford to check out parts of the system and is preparing to do a simulation study of the queuing algorithm. A small technical group is building prototype interconnection equipment for the various data interfaces.

The planning work was sponsored by a Josiah Macy Foundation planning grant and further funding has been granted from NIH and Macy Foundation. Much credit for ideas and procedures goes to other computer installations and other people, notably project MAC at M.I.T., MEDLAB at the Latter-Day Saints Hospital in Salt Lake City, the University of California at Berkeley Computation Center and ARPA project, U.C. San Francisco medical school, U.C.L.A. Health Sciences, etc., and of course the Computation Center and the Computer Science Department of Stanford itself.