A Description of a Cooperative Venture in the Production of an Automatic Coding System*

Wesley S. Melahn

The RAND Corporation, Santa Monica, Calif.

The exact origin of the idea of a co-operative undertaking to produce an automatic coding system is hard to determine as we look back at the history of the Project for the Advancement of Coding Techniques (PACT). For years, people in service organizations, like the computing groups in which many of us work, have been looking for ways to persuade fellow organizations to "co-operate". Achieving co-operation in this case consists of convincing a fellow worker that he should supply you, gratis, with a set of board wiring diagrams or some similar piece of know-how in the hope that some day you may have something the lender can use. West Coast groups, being younger and less experienced than many of our eastern neighbors, have tended to become experts at this kind of co-operation. The purpose of this paper is not to discuss the merits of this technique, but to point out that PACT is the product of a different sort of co-operative undertaking.

The joint venture, PACT, got under way during November of 1954. That was the time when several organizations on the West Coast were taking a fairly critical look at their achievements and future needs in computing equipment. Several of them had decided to turn back the first large digital computer that they had rented in favor of a bigger, faster, and more flexible machine. Several facts emerged clearly during these self-examinations. Problem check-out was expensive and time-consuming. In some cases, check-out time was running as high as 40% of production time. Estimates of the cost of writing and checking a program put the cost at something like $2 to $10 per instruction. Far too many of the problems that were being run were inefficiently and awkwardly put together, resulting in an excess of machine time for execution. Experienced programmers were hard to find. Nearly every computing organization was expanding.

One of the suggested ways of relieving the programmer shortage, and at the same time improving the quality of the program that gets to the machine, was to employ more automatic coding; that is, to let the machine do a larger share of the work of preparing a problem for machine computation. However, the more sophisticated problem preparation programs were by no means an obvious clear-cut answer to the problem of what to do about coding. Every West Coast organization had already made considerable investment in utility programs. Everybody had an interpretive system of one kind or another that was aimed at making problems easier to code. These interpreters helped some by reducing preparation time, but the cost in machine time required to execute the programs was high. Everybody had a stack of library programs for helping to solve the

* Presented at the meeting of the Association, Sept. 14–16, 1955.

problems, or parts of problems, that were known to repeat often. Everybody used some kind of an assembly program for putting together their programs. These assemblers had grown in scope and usefulness to the point where many of them could justifiably be called compilers. Co-operation, the first kind described, was taking place on an ever-increasing scale, but with success that was limited by the incompatibility of the systems in operation. Incompatibility was showing up in another way too. The replacement of the 701's that several West Coast computing groups were planning would make virtually all of their machine language library obsolete and would require many problems to be recoded. In view of these sobering facts, there was reason for considerable restraint on the part of computer group managements to approving investment of time and effort in production of more elaborate programs for permitting the machine to aid the coder in the preparation of programs. On the other hand, there was an ever-increasing need to do something about the high cost of producing answers.

Jack Strong and Frank Wagner, from North American Aviation, deserve a great deal of credit for suggesting that a co-operative effort aimed at obtaining a more elaborate automatic coding system would be feasible. They pointed out to many of us that a truly co-operative effort could cut the cost to each organization and substantially reduce the elapsed time to completion of the program. It was largely through their efforts that enough enthusiasm was generated in the Los Angeles area to warrant calling a meeting of several computing groups to discuss the proposed joint effort. PACT, as the organization later came to be known, was formed, substantially as suggested by Strong and Wagner, at a meeting which took place at the Douglas Aircraft Company's El Segundo plant on November 16, 1954.

The organizations represented at the founders' meeting were: Douglas (El Segundo), Douglas (Santa Monica), IBM, North American Aviation, Ramo-Wooldridge, and The RAND Corporation. Representatives of these organizations agreed to set up two committees to undertake the joint project. A Policy Committee composed of the supervisors, or designated representatives, from each installation was formed to direct the work of a second committee. The Working Committee was composed of at least one representative, working nearly full time, from each co-operating organization. It was agreed that other organizations in the area would be invited to join on either a full-time working basis, in which case they would enjoy full and equal voting rights with the founders, or on a part-time basis. The RAND Corporation was selected as a satisfactory neutral spot at which to meet and work on the project.

The Policy Committee, as the founders called themselves, instructed the Working Committee to consider six months as a target period in which to study the problem, recommend a solution and code an experimental program for 701. This deadline was suggested in order to allow a few months' trial in actual use in each installation before some of them discarded their 701's in favor of 704's. It was hoped that the 701 experience would serve as a guide in setting up a coding system for the 704.

The full-time working members of the project during the first six months when PACT I was produced, were supplied by: Douglas (El Segundo), Douglas (Long Beach), Douglas (Santa Monica), Lockheed (Burbank), North American Aviation, N.O.T.S. (Inyokern) and The RAND Corporation. IBM and Ramo-Wooldridge made part-time contributions of manpower. Several of the member organizations have consistently supplied more than one man to the Working Committee; and all of the representatives consulted freely with fellow-workers in their home organizations. PACT I, as the first experimental program is called, in a very real way is the joint product of many people's efforts.

Soon after the Working Committee began to meet regularly, it became apparent that it had more problems than the Policy Committee had recognized. The most serious of these was language. People really didn't know their neighbors' systems well enough to be able to talk about them intelligently, much less to understand the subtle pressures that made small points seem important enough to argue about. After a few weeks of mutual education, the Committee raised its sights a little and inquired concerning goings-on elsewhere. A quick look at the UNIVAC A-2 compiler, at Whirlwind programs, at University of Michigan programs, and at the plans for FORTRAN, convinced the Committee that a satisfactory solution to its problem was not available in the computer literature.

The Committee was seeking a modest program that would attempt to remedy some of the serious faults of their current systems, rather than a new system that would revolutionize coding. They felt that much good technique had been learned by their organizations, and that this should be retained in any new coding systems.

The most serious shortcomings of the systems in use seemed to the Committee to be:

1) The high machine bias of the language used to describe the calculations and manipulations that the program was to perform.

2) The large amount of labor involved in writing down a code.

3) The frequency of errors committed by coders doing routine things that they know how to do.

The machine bias of the language was most evident in the operation codes. The symbolic address schemes that the various groups had adopted provided slightly more flexibility than the absolute machine addresses, but the operation codes were still exactly those provided by the hardware of the machine plus a few closed subroutines. The use of the machine operation code required each coder to learn all the quirks and peculiarities of the machine before he could code a problem. It required that the rules be learned again every time the computing machine was changed. Further, the entire library was made obsolete by a change of machines. The training of coders was slowed down, and errors were caused by the fact that the expression of ideas in machine language is not usually direct and to the point.

The labor involved in writing down the code was due to several factors. A number of the frequently used manipulations required lengthy awkward machine language expressions to describe them. The symbols adopted by the assembly program designers were long and often cumbersome. The inflexible parallel input to the computer caused an unyielding rigid format to be adopted for inputs. The inability to salvage bits and pieces of old problems contributed to the writing required for each new problem. Far too little of the day-to-day work could be conveniently put in the library for later use. Often the only useful remainder of a problem was the know-how acquired by the coder.

Errors in everybody's codes were far too frequent. The open shop and the closed shop people, surprisingly enough, found that they agreed concerning the types of coding errors that caused them the most trouble. Most of the errors, something like 40% of those reported in error studies, turned up in the logical or housekeeping operations, building loops, tallying, etc. The arithmetic errors accounted for only about 15% of the total reported, and most of these had to do with scaling.

The various suggestions of Working Committee members for correcting these difficulties were subjected to the critical scrutiny of their neighbors. The pressure of a deadline and the enthusiasm of the Committee members did not always insure that full credit was given to the source of ideas nor that unpopular ideas were rejected with gentleness. Looking back, it is somewhat surprising that the Committee not only survived these weeks of discussion but met regularly four days a week with virtually no absenteeism. By the end of January 1955, the Committee had selected a few compatible ideas which they proposed to try out. PACT I received the approval of the Policy Committee in late January 1955. It was coded for 701 by the Working Committee between February and June 1955.

The PACT I program attempts to make coding easier without sacrificing speed of execution. It is a pre-execution processing type of program. The PACT language description of the calculation and cues for storage assignments are processed to produce a fixed point machine language program that runs at machine rate during execution of the calculation. The aim was not to produce a best code by optimizing some feature, but to produce a code that would compare favorably with a hand-written code. The language in which the description of the calculation is written was selected with a view to making it simple and effective. An attempt was made to divorce the operations from a particular general purpose machine and to aim them at a selected group of problems. The class of things that could be used as factors was broadened to include numbers, single variables, systematic arrays up to dimension two, and results of previous steps in the calculation. Directness of expression was the aim wherever possible. Reduction of labor was achieved by turning over some of the routine clerical jobs to the program, by making the symbols short, and by making frequently used expressions concise. Other papers in this series will describe the features of PACT-I in detail and will discuss the coding techniques employed in the program.

The first few months of experience seem to indicate that the co-operating computer groups will be handsomely repaid for the small investment in PACT I by the savings in coding and machine time. Perhaps the greatest dividends will come from the demonstration that co-operative undertakings by groups with diversified interests can succeed and can speed up the development of the art of machine computation.

APPENDIX I

PACT Policy Committee Members Who Approved PACT I 1/28/55

Regular Members

Douglas Aircraft Company, Inc., El Segundo, California	W. Schlieser W. Dobrusky
Douglas Aircraft Company, Inc., Long Beach, California	J. Viall
Douglas Aircraft Company, Inc., Santa Monica, California	J. Lowe R. Middlekauff
Lockheed Aircraft Corporation, Burbank, California	L. Amaya
Naval Ordnance Test Station, China Lake, California	B. Oldfield
North American Aviation, Inc., Los Angeles, California	J. Strong F. Wagner
The RAND Corporation, Santa Monica, California	P. Armer W. Melahn

Part Time Members

International Business Machines, Santa Monica, California	W. McClelland
Lockheed Aircraft Corporation, Van Nuys, California	R. Bemer
Ramo-Wooldridge Corporation, Inglewood, California	W. Bauer J. Mercel

APPENDIX II

PACT Working Committee

Regular Members

Douglas Aircraft Company, Inc., El Segundo, California	
H. G. Martin	Subroutine linkages
Douglas Aircraft Company, Inc., Long Beach, California	
R. P. Bacon November 15, 1954 to June 1955	Subroutine operations
T. Littlejohn June 1955 to date	Subroutine operations
Douglas Aircraft Company, Inc., Santa Monica, California	
C. L. Baker	Card reading, final listing
Lockheed Aircraft Corporation, Burbank, California	
R. C. Luke	Storage assignment
Naval Ordnance Test Station, China Lake, California	
B. G. Oldfield	Operation expansion
R. C. Miller, Jr. Chairman, May 17, 1955 to date	Operation expansion
R. G. Selfridge	Operation expansion
North American Aviation, Inc., Los Angeles, California	
O. R. Mock	Duplicate expansion, temporary storage, final assembly and punching
F. R. Anderson	Manual
The RAND Corporation, Santa Monica, California	
W. S. Melahn Chairman, November 15, 1954 to May 17, 1955	Loop expansion

J. I. Schwartz	Loop expansion
G. S. Hempstead	Loop expansion
I. Greenwald	PACT Primer
J. Derr	Storage assignment

Part Time Members

International Business Machines, Santa Monica, California	
W. McClelland	Formulation
Ramo-Wooldridge Corporation, Inglewood, California	
J. Mercel	Formulation
T. Tack	Formulation
The RAND Corporation, Santa Monica, California	
J. Shaw	Formulation