19

September 30, 1971

COMPUTER EVALUATION AND SELECTION: A CASE STUDY AND A METHODOLOGY

C. Gordon Bell
Professor, Computer Science and
Electrical Engineering
Carnegie-Mellon University
Pittsburgh, Pa. 15213

John W. McCredie
Assistant Professor, Computer Science
Carnegie-Mellon University
Pittsburgh, Pa. 15213

Submitted to the Spring Joint Computer Conference, 1972

Inquiries on the paper should be addressed to:

Professor C. Gordon Bell
Department of Computer Science
Carnegie-Mellon University
Pittsburgh, Pa. 15213
Telephone: 412-683-7000

ABSTRACT

This paper presents a taxonomy of important decision variables and a methodology for handling them in a computer selection and changeover decision. A case study from a university illustrates the problem environment and presents the background from which the quantitative methodology evolved. The reader is invited to review an actual decision process and evaluate our proposed way of handling conflicting objectives and constraints of many diverse user groups.

INTRODUCTION

One of the most traumatic decisions required of an organization doing computation is the problem of deciding when to change to a different facility, and how to affect this changeover. As technology evolves and manufacturers introduce new computers, and as budgets and goals change, organizations select new computers. These changes directly affect careers of many people and involve large capital expenditures. Often the choices must be made within a framework of short time limits, little information and intense personal interactions. This paper presents a taxonomy of some of the decision variables, and a methodology for handling them so that uncertainty may be reduced and the quality of the resulting decision improved. The methodology is an outgrowth of an eighteen month computer decision deliberation at Carnegie-Mellon University (CMU). A case study of this decision illustrates many of the problems inherent in such a process and indicates why the methodology could be an important tool in future decisions. We believe that the systematic approach outlined would have increased communication and led to a less chaotic decision environment.

Recently there have been articles published on computer measurement and evaluation.* Techniques such as hardware and software monitoring, repeatable user scripts, mathematical models, simulations, and benchmark user profiles provide a wide range of tools with which to compare alternative systems. Bell and Newell (1971) present a comprehensive methodology for representing and comparing computer hardware structures. The

---

*For example, the Association for Computing Machinery recently sponsored the SIGOPS Workshop on System Performance Evaluation (1971). A similar workshop co-sponsored by the IEEE and Argonne National Laboratory was held in October, 1971.

administrators, user representative committees, colleges, departments, etc. with ultimate responsibility for major computer policy decisions are often not familiar with the vast amount of technical literature on the subject. Nor does it appear to them that it is even possible that these decisions can be subjected to such analysis. Users are often split into groups which lobby for their own solution to the problem. Even within a well structured decision environment, the activities of these competing groups may appear chaotic to those who must make the final decision. The following section sketches the important events of a decision at CMU.

CASE STUDY OF EVENTS AT CARNEGIE-MELLON UNIVERSITY

Two standard case study methods are: (a) a detached, scientific, observer having access to personnel and their communications during a decision follows, records, and analyzes all important events; (b) a non-participating historian-analyst gathers all relevant data allowing significant time to buffer individuals from the actual events.

The analysis of the following sections has neither of these two properties. Both authors participated in the decision process in an active way. Detailed notes of meetings were not gathered, but public memoranda document many important events. Since the actual decision was announced within the last year there is no historical perspective.

The following format will be used to describe the important events concerning the computer decision at CMU:

<approximate date>-<event>: <editorial comment about this event>

12/68 - Computer Users Planning Committee formed: The administration appointed this committee to draft a policy for computation. This committee discussed a computing objective function and tried to outline a number of constraints.

4/69 - Computer Users Planning Committee presents Computing Policy document to the University: One of the major recommendations of this document was that a permanent University Computing Council (UCC) be formed to advise the administration on computing affairs. A three member subgroup from UCC was recommended, called the Computing Board, to consult with the Computation Center director on a daily basis.

5/69 - Computer Science Department decision to no longer support one of the two large computers on the campus for its major research project: The research goals of the Computer Science Department changed, and they decided to withdraw financial support of this machine in June, 1970.

9/69 - University Computation Council (UCC) appointed: The UCC was a group of about 20 faculty and two students representing different segments of the University.

10/69 - Computing Board formed (three faculty members).

10/69 - UCC formulates basic problems: Subcommittees were formed to study computation budget, user services, evaluation procedures, and basic policy. The single most important task of UCC was to recommend what equipment configuration would serve as the general campus facility after the Computer Science Department altered the budgetary framework.

12/69 - UCC attempts to measure user demand and determine budgetary constraints: Subcommittees had difficulty gathering statistics on current operations and determining what budgetary constraints the University will place upon computing resources in the future.

12/69 - UCC gathers a number of equipment proposals for different configurations: Vendors provided alternative configurations and UCC members proposed others.

2/70 - Director of Computation Center resigns.

3/70 - Proposals reduced to two different alternatives.

4/70 - UCC makes recommendation and University accepts: The decision called for the continued rental, for a trial period of one year, of the large general purpose computer being phased out by the Computer Science Department. UCC was unwilling to recommend purchase of this machine without an extensive test period as a general campus facility, but it did not specify tests or performance levels.

4/70 - Administration appoints new Director of Computation Center from current staff.

6/70 - Director and Computer Board make plans for revised configuration: To make the configuration satisfy a number of user requirements, the management of the Center proposed changes to the equipment configuration of the systems and to operational policies. The University signed a contract with a vendor who was to provide a subsystem having substantially better performance charachteristics than the rented equipment. The contract had a one year minimum lease provision.

9/70 - New UCC appointed and Computer Board disbanded: This UCC had a number of new members. At its first meeting it decided that the three member faculty Computer Board was no longer necessary.

10/70 - Vendor failed to install major sybsystem and new contract signed with another vendor: The subsystem was not installed on schedule, and after a number of delays the contract was terminated. A contract with another vendor was signed (about 1/3 of total system cost). This contract provided for a minimum lease of two years if the sybsystem was not purchased.

12/70 - New components added to system: The second vendor installed the subsystem, and a number of other components were added to improve system performance.

1/71 - UCC approves two committees to evaluate performance for March deadline: The first was a User Committee that was to study computing constraints and objectives. This committee never functioned. The second was a Computer Selection committee that was to evaluate the present situation and recommend whether the current configuration had performed well enough that the University should now purchase it. If the system was not to be retained, an alternative would be required.

2/71 - Selection Committee holds meetings: Interested groups made verbal and written proposals to the Selection Committee. One of the important unknown variables was user changeover cost.

3/71 - Attempts to measure user changeover costs: A number of groups tried to measure this cost.

3/71 - UCC Meeting: At this meeting the Selection Committee made its recommendation, and interested groups reported their views. Individual UCC members were polled. The groups reached no consensus.

3/71 - Administration announces decision to purchase machine presently on rental and to commit the University to this configuration for an extended period.

In the next section we present a general formulation of the computer selection and changeover problem and an approach to its resolution.

A COMPUTER SELECTION METHODOLOGY

The following formalization of the changeover problem will serve as a common descriptive base throughout the paper. Stating the problem in a standard optimization framework forces analysts to face important issues at the outset. The form of the objective function and constraint set may change with time, and it is certain to vary among different interest groups. An important achievement for any organization is the creation and circulation of an explicit constraint set and a realistic approximization to the organization's computation objective function. A formal attempt to view the problem in common terms throughout the decision hierarchy could reduce noise and increase communication among the many groups and subgroups active in the decision process. However, this type of formulation will probably not lead to an algorithmic solution. The data are inherently imprecise, and many features will remain subjective. Present efforts in decision theory concerning "fuzzy" constraints and objectives may prove helpful in the future. (Bellman, 1971)

DEFINITION OF TERMS

S_{ijk} ≡ the level of service i which will be provided on system j, in time period k. Examples of typical service components include computation speed, primary memory available to users, file size and security, reliability, number of hours available to users, flexibility, compatibility with other institutions, ease of learning and use, extent of documentation, quality and number of supported languages, power of operating system, response time as a function of request, continuity with previous operations, type of user access, input/output facilities, vendor support, etc.

$\underline{S}$ ≡ the array of service levels S_{ijk}.

$C(S_{ijk})$ ≡ the total cost of providing S_{ijk} of service i on system j in time period k. The cost should include items such as programming support, maintenance, general operations, etc. This figure should also include costs of changing from some mode of operation in period k-1 to any different mode in period k. Examples of such costs are user conversion expenses and potential contractural penalties to vendors.

B_k ≡ the subsidy the organization is willing to provide to the user community in year k. Additional funds must be generated from sources outside of the organization as a result of providing different S_{ijk}.

$EI(S_{ijk})$≡ the external income which may be generated in period k by providing S_{ijk} of service i on system j.

$F(\underline{S})$ ≡ an objective function which maps the space of service alternatives into a space of values.

I ≡ the set of required services.

J ≡ the set of systems under consideration.

R_{ik} ≡ the required minimum level of service i in period k.

FORMULATION OF PROBLEM

Using the previous definitions we may formulate a few illustrative problems. As a first example consider the case in which the organization is able to project firm budgetary guidelines for n future time periods. The situation could be represented in the following problem in which we maximize an overall measure of service subject to budget and minimum service constraints.

(1) MAXIMIZE $F(\underline{S})$

$$\text{subject to: } \sum_{j}\sum_{i} \{C(S_{ijk}) - EI(S_{ijk})\} \leq B_k$$

$$\text{for } k = 1, 2, \ldots. n$$

$$\sum_{j} S_{ijk} \geq R_{ik}$$

$$\text{for } k = 1, 2, \ldots. n \text{ and } i \in I$$

$$S_{ijk} \geq 0 \text{ for all } i, j, k$$

An alternative approach would be to attempt to minimize the time discounted value of the organization's computing subsidy while still meeting service constraints. Formulation (2) is an example of this type of decision where r is the appropriate internal discount rate.

$$\text{(2) MINIMIZE } \sum_k \sum_j \sum_i \frac{\{C(S_{ijk}) - EI(S_{ijk})\}}{(1+r)^k}$$

$$\text{Subject to: } \sum_j S_{ijk} \geq R_{ik}$$

$$\text{for } k = 1, 2, \ldots n \text{ and } i \text{ EI}$$

$$S_{ijk} \geq 0 \text{ for all } i, j, k$$

Other constraints could be added in an obvious fashion. For example, the organization might be regulated to earn not more than some level $EI(S_{ijk})$ from a particular service. Constraints on the subsidy B_k for particular time periods may enter formulation (2) as they did in (1).

The previous formulations allow for a number of different system configurations. Specific services may be purchased from outside; one or many systems may be selected for purchase or lease. Many of the variables may be very hard to obtain, particularly if a proposed system is a new entry in the vendor's equipment line. The virtue of this type of formalism is that policy makers are forced to obtain the most reliable data possible along a number of well defined dimensions. When uncertainty exists in a set of numbers it should be recognized and attempts should be made to measure the variance of the estimates.

EXAMPLE ALTERNATIVES

The last section proposed a general framework in which computer systems can be compared along dimensions such as cost, availability, hardware and software performance, and flexibility. At any point in time, the set, J, of alternative systems under serious consideration, is usually small. To provide service continuity, equipment changeover decisions are often incremental and only a subset of the total equipment configuration is altered in any period. Simple heuristics can usually narrow the search to four or five alternatives. As the data base grows, and the organization begins to understand its constraints and objectives more clearly, additional alternatives may be considered easily. In the following sections we present the type of data that should be considered for each serious alternative. Our intent is to indicate the kinds of quantitative comparisons that are possible. This common data may be used in a number of ways within the decision hierarchy as people place their own value weightings to various dimensions.

The following four alternatives present a wide range of typical potential selections. The first two are well understood in our environment and represent little change from current operations. The last two are untested and risky approaches that appear to have high potential cost savings. Of course uncertainties are greatest in these latter systems. CMU owns a Univac 1108 which is heavily used. Each configuration discussed below is an addition to the basic 1108 system capability which will remain.

THE SYSTEMS

(A) Status Quo

Purchase the large, general purpose, time-sharing system which was at CMU on a rental basis. The machine has been in service at CMU for over four years, but much of that time was dedicated to specific externally funded research projects of the Computer Science Department. Many users are doing useful work on, and are comfortable with, this machine.

(B) Purchase a Less Expensive General Purpose System with Some External Time Purchases

The Computer Science Department at CMU has two such machines for comparison. Hence we had a great deal of operational data, and were able to avoid problems of vendor noise. We could have considered other machines in this cost-performance category, but time constraints and problems of gathering data from outside sources forced us to use system (B) as a representative of this class of machine. If this alternative looked promising, other similar systems could then enter the decision space.

(C) Multiple Minicomputers with External Purchases

This alternative did not receive a great deal of attention in the actual selection procedure because of time constraints. Much of the data is presently unavilable, but this configuration appeared to have lowest cost. Without further exploration one cannot be sure whether the low cost and reasonably high performance figures are a result of uncertainty in the data or an inherent feature of the approach. We encourage others to examine this method of providing a wide range of computing services.

The National Science Foundation sponsored an evaluation in which the IBM 1130 was compared with buying time from commercial timesharing companies at a number of colleges. A configuration using this approach at a larger university could have two minis running Fortran and other languages such as CSMP, circuit analysis packages, etc. for direct use by researchers and students. To satisfy terminal constraints, two minis could be dedicated to time-sharing for either BASIC or Fortran. A final mini could be used solely for file editing, and its output could interface to the internal Univac 1108 and to off-campus systems where time would be purchased. Local firms supply a full range of services to the commercial market.

(D) Purchase Services from External Vendors

All computing needs beyond the basic 1108 service would be purchased from outside sources. In this environment the Computation Center would function as a clearing house for commercial services. The 1108 load would increase since all outside computing would have a direct, observable, cost associated with it. The satisfied 1108 users were quite concerned that this alternative would severely cut back their level of service.

There are a great many other configurations that could serve the needs of CMU. Some were examined, and others were neglected because of time constraints and the lack of an interested group to press the cause of a particular system. For example, with some very large systems we might achieve an order of magnitude improvement in price/performance. The large initial investments generally required to purchase such configurations violated obvious budget constraints since the University is operating under financial pressures. Another interesting alternative that was not examined thoroughly was the introduction of a very large number of minicomputers. For the same cost as alternative (A) we could buy between 40 to 400 minicomputers (at a price of $60,000-$6,000). This alternative might provide more real power and reliability than any of the others, but the construction of such a facility would require more research and development than uncertainty constraints would allow.

THE DIMENSIONS

A general problem solving technique used in situations similar to the CMU decision is: generate a solution; test it against constraints such as cost, time to gather data, data reliability, user transitions, uncertainty, availability, etc.; and finally evaluate it by applying an objective function to the relevant characteristics.

Before a serious comparison should begin, the organization must understand its current computing load. This plea to "understand thyself" may sound trite, but in our experience conventional billing and accounting information is inadequate to accurately characterize the user profile. Load and performance monitoring should be an integral part of every installation. Data from such efforts may not be used continuously, but when the time arrives for a change, it becomes invaluable.

The interesting dimensions for quantitative evaluation may be grouped as follows:

Costs: Capital expenditure; operations; maintenance; user changeover;...

User Machine Performance: Processor capability; availability; terminal access; reliability; interesting hardware features; ....

Software Performance: Operating system features, languages and facilities; user interface; reliability; ....

Uncertainties: Future machine evolution; price changes, load changes;...

Space limitations preclude discussion of each of these dimensions, or even detailed analysis of one of them. Hence we shall illustrate, in some detail, one aspect of one dimension. Consider a comparison between language capabilities of an existing facility (machine A) and a proposed alternative (machine B). The comparison will be over some universe of languages, $\mathcal{L}$ = {Fortran, Algol, Basic, APL,...}, in the context of the usage of the particular installation. For example, define a particular metric, $\mathcal{C}_{\mathcal{L}}(A,B)$, by

$$\mathcal{C}_{\mathcal{L}}(A,B) = \sum_{\ell\in\mathcal{L}} \{\max(T_{\ell}\cdot p_{\ell,j} - c_{\ell,j}) \cdot \sum_{k}(n_{\ell,k}\, I_k)\}$$

where

$\mathcal{L} = \{\ell_1,\ldots,\ell_n\}$ is the universe of language (not all of which need be on both A and B)

$T\ell$ = a measure of the useful of language ℓ

$P_{i,j}$=relative language power; language j on B compared with i on A

$c_{i,j}$=conversion cost measure, programs in language i on A into language j on B

$n_{\ell,k}$=number of users of language ℓ in class k

I_k = relative "importance" of class k users

This particular metric is a sum of terms for each language which includes factors such as: on the proposed system it may be necessary (or advantageous) to switch languages; that there is a one-time cost associated with such a conversion; and that not all languages are equally used nor are users equally "important". We wish to stress the practicality of gathering the necessary data, by benchmark tests, for example.

The following table illustrates a number of important variables which need to be evaluated during any major equipment decision.

Costs

Direct and Operational

- Purchase
- Rental phase out costs
- Contract penalties
- Interest
- Hardware training
- Software training
- Operating expenses
- Changeover
- Outside purchases to satisfy needs
- Systems personnel for maintenance

Performance

- Availability
- Monitor
- CPU power
- Memory space
- Peripheral power
- Terminals
- Batch capability
- File systems
- Languages and Operating System capabilities
- Ease of learning
- Documentation
- Response time
- Recovery time after a crash
- Reliability
- Computability
- Multiprocessor capability

CONCLUSIONS

An analytical framework for a decision of this type will usually have the following benefits: reduce the likelihood that random noise will finally trigger the decision in a completely irrational manner; users will learn to place a value on computation; agreement on the various values, costs and performances; and it establishes a measurement framework by which the computation facilities can be run and evaluated.

In making the decision data gathering should occur calmly. Even for several thousand users it is possible to gather complete information about the user profile; and this analysis should be carried out. Data gathering should be carried out by about three unbiased people (all of whom are knowledgable in computer systems).

Given that a computer is in an environment for a year or more, no matter with what results, changing to some other system (which is not a proper subset of the first system) will be not with the blessing of the users.

Computer selection decisions do admit to quantitative analysis, and the techniques are well known. The necessary data are either available, or methods for collecting them are available. The problem of making a rational selection, once the technical data are organized, is similar to all capital expenditure problems and the same management techniques should be applied.

References

[1] Proceedings of the ACM SIGOPS Workshop on System Performance Evaluation, Association for Computing Machinery, New York, 1971.

[2] Bell, C. G. and Newell, A., Computer Structures: Readings and Examples, McGraw-Hill Book Company, New York, 1971.

[3] Bellman, R. E., and Zadeh, L. A., "Decision Making in a Fuzzy Environment," Management Science, Vol. 17, No. 4, Dec. 1970.