INTERDEPARTMENTAL

MASSACHUSETTS INSTITUTE OF TECHNOLOGY CAMBRIDGE, MASS. 02139

from the office of

TO: G. F. Clancy
F. J. Corbato
R. C. Daley
E. L. Glaser
R. M Graham
K. J. Martin
R. L. Rappaport
J. H. Saltzer ✓
G. G. Schroeder
T. H. Van Vleck
J. Osanna (BTL)
P. G. Neumann
B. A. Tague
V. A. Vyssotsky

A. Bensousson (GE)
H. Frick
L. Lambert

C. Jones (DOD)
D. Johnson

FROM: Robert R. Fenichel

DATE: October 21, 1966

SUBJECT: New Transactor Overview

I have attached a new version of BT.0; your comments are welcome. Several changes from the old version (August 12, 1966) should be noted:

(a) Certain old features have been removed. In particular, the complaint service is an attached front end to mail (BO.8), and the resource transfer facility is also out in System Administration (BO.5.05)

(b) The former document's planned reservations are now the responsibility of user-owned demons.

(c) This new document is at a level of much greater generality (vagueness) than the previous version. Back-up documents (BT.1.0, BT.2.0, etc.) will contain underoverviews.

(d) The BT portion of BTABLE needs utter revision.

(e) The notion of service streams has been removed from the Transactor.

Identification

Transactor Overview

Robert R. Fenichel, Leo J. Lambert, Karolyn J. Martin

Purpose

It is pleasant to think that the Law of Large Numbers, buttressed from time to time by the Law of Supply and Demand, will be sufficient regulation for MULTICS. The Transactor, described in this section, exists only because a self-regulation MULTICS is neither feasible nor desirable.

Metamorphic Description

Given stochastically-determined service of some kind, the user who wants better service will clearly have to pay for it. Not only does he pay for his larger share of the facilities, but he also pays for the administrative costs of giving him distinguished treatment.

On the road, for example, traffic speed is more of less codeterminate with the spacing between cars. But the driver who wishes to move much faster than his peers needs more than a little extra headway; he may need a whole lane. Strikingly, the same is true if he wants to more much more slowly than his peers.

The Transactor is the MULTICS facility for Sterling Moss and for Sterling Moss' grandmother. In addition, the Transactor is the agency which predicts traffic densities, reserves lane-pairs for overwidth housetrailers, and throws some drivers off the road in order to relieve congestion.

General Description

1. The Load-Leveler (BT.1)

On a minute-to-minute basis, MULTICS must decide whether additional users will be permitted to log in. In extreme cases, MULTICS must avoid general degradation of service by logging out some users and suspending operation of some absentee processes. An algorithm is needed.

The Load-Leveler is that protion of the Transactor which arranges for the system to be busy, but not too busy.

2. Advisors of Look-Behind Algorithms (BT.2)

Many MULTICS scheduling and allocation procedures are based upon look-behind algorithms. While these algorithms succeed in supplying equitable service to typical users, some special users may wish to advise the schedulers of special requirements.

2.1 Multilevel Storage Manager Advisor (BT.2.1)

A file may be needed in a hurry, even though the file has not recently been active. Anticipating such a need, a user may arrange through the Transactor to have the file in question held in relatively fast-access secondary storage.

Conversely, an impecunious user may arrange through the Transactor to have some or all of his files held on the cheapest device in the hierarchy.

2.2 Basic File System Advisor (BT.2.2)

A user may wish to recommend to page control that some segment should be unpaged, or even latched into core. Similarly, he may wish to suggest to segment control that a certain group of segments always be loaded and unloaded together.

The Transactor will provide an ear for such suggestions.

2.3 CPU Response Advisor (BT.2.3)

CPU response is at least 4-dimensional. That is, the system's response requires two curves of description:

(2.3.1) There is a frequency distribution of Quantity-of-Service-Requested. In the case of CTSS, for example, this curve is roughly normal; it is positively skewed with a mean of about half a second.

(2.3.2) The ratio of elapsed-time to CPU-time-requested may be plotted against CPU-time-requested. For short requests, overhead causes the ratio to blow up. Long requests, different policies have different results:

(a) In an airline-reservation system, requests might accumulate priority as they wait so that a long-waiting request eventually takes over the machine. On the curve, the right-hand asymptote is 1.

(b) Straight round-robin scheduling will give a right-hand asymptote equal to n, where n is the number of active processes.

(c) Multiple-queue scheduling (like that of CTSS, for example) will cause the ratio to rise without limit for large requests

One can imagine a user's demand for special CPU response. In terms of the curves, this user has a peculiar 2.3.1 curve and he wants a 2.3.2 curve to fit. For example,

(a) The user knows that he will present one 8-hour job sometime during the coming year. He will want it back in 12 hours.

(b) The user's apparatus produces a 50-μ request every 60 ms. Each request must be processed in 75μ.

In either of these cases, the user is adding less than 0.1 °/o to the system's load. But if (a) is offered, the system must probably own a spare CPU, and if (b) is offered, extra core is needed.

Because any offer of special response will have system-wide repercussions, special response is an installation-dependent facility in MULTICS.

A more reasonable user demand is for better response, where better (unlike special) is never precisely defined. In terms of the curves 2.3.1 and 2.3.2, a user seeking better response is only trying to depress 2.3.2 more or less uniformly.

Through the Transactor, a user will be able to choose one of a few (say, 2) levels of better response. Each of these levels will be associated with a value of some scheduling-algorithm-dependent parameter, such as "PB" in CTSS.

2.4 The I/O System Adviser (BT.2.4)

A user may wish to advise the I/O System just as he advises the Scheduler. The Transactor will allow the user to choose one of a few (say, 2) levels of preferred queue treatment.

3. The Reserver (BT.3)

Time-sharing is intended to eliminate user awareness of conflicting, simultaneous demands for unique equipment. This elimination is generally accomplished by commutating that equipment among conflicting users according to a rapid schedule.

Such commutation is not possible with such devices as teletypewriters, tape drives, display units, and unit record equipment. These devices must be allotted for "job-length" periods which may be three or four orders of magnitude longer than the periods of the commutation schedule used for CPU time, disk time, etc.

In order to allow rapid, unscheduled access to non-commutatable devices, then, very large pools of these devices are needed to account for normal variation in demand. To avoid the cost of maintaining such pools, MULTICS will, where feasible, substitute scheduled access for demand access.

Scheduled access will be encouraged for other services which would require vast overcapacity if offered on a demand basis. Better response, for example, may be available only (or at least primarily) through a machine-administered sign-up sheet.

A popular view is that the reserver is the same sort of beast as the CPU scheduler (BJ.4), except bigger. This view is erroneous in one major respect.

The Scheduler, on the one hand, deals primarily with unremarkable users. Not only are the external specifications of the Scheduler constant while policy is changing, but even changes in policy do not affect most of the Scheduler's users.

The Reserver, on the other hand, is assigned by definition to dealing with irregularly-shaped users. Even the grossest specification of the Reserver will carry policy assumptions, and most of the Reserver's policies will be immediately perceptible to the Reserver's users.

For this reason, it seems prudent to list here those prejudices and considerations which are most responsible for the present specifications of the Reserver. Where quantitative terms (e.g., "month") are used, they should naturally be taken as examples of parameter values.

(a) A reservation for time today is more securely made last week than yesterday.

(b) On the other hand, the system can't make reservations in ignorance of its own capacity. A reservation put in last year may not be meaningful.

(c) The chronological priority of (a), moreover, is not the only kind of priority.

(d) Notwithstanding (c), the notion of confirmation is unassailable. A user with a confirmed reservation is in a state of grace far beyond all priority.

(e) In particular, it is natural to wonder what happens after a system crash has caused reserved or other service to be unavailable. The users so deprived do have a claim against the system.

This claim is not sufficient to cause other confirmed reservations to be automatically pushed back.

(f) In the normal case, a user who wishes to reserve two hours of tape-time does not much care ("ANYTIME Thursday") when those two hours come. That is, a reservation request will normally include a calendar period (when to reserve) and a much shorter elapsed-time subperiod (how much to reserve). The calendar period may have a polarity ("Early as possible on Thursday"), but the user will generally allow the system maximum leeway.

(g) When a reservation is confirmed, the user may wish to freeze his reservation to a precise subperiod of the calendar period.

(h) Reservations will commonly be coordinated sets ("2 tapes and a printer at the same time); the Reserver will generally deal in terms of reservation groups.

(i) The general mechanism will depend upon a division of the future into near (this month and next) and far. A reservation group for the near future will be immediately considered and rejected or confirmed. The notion of priority is not relevant.

(j) A reservation group for the far future (say, January) will be noted in one of the January Reservation Files. These several files correspond to the several levels of priority recognized by the Reserver; regulation is by BFS access controls, and may probably be brought.

On the first of December, these files are examined in order and confirmations are made. Rejections are handled by mail and by awakening designated user processes.

D) The Estimator

Through the Transactor, the user has several different ways of altering the packaging in which system services come to him. Those services are unchanged, of course, and the user may wonder if it pays to send an Air Mail letter from Cambridge to Medford.

The estimator is build around a simple time-series model of the system. Through the Estimator, the user can ask such questions as:

"How much will secondary storage cost me next month? How much would it cost me if I restricted all of my files to disc residence?"

"If I come by at 1400 on April 19, 1968, will I be able to log in?"

"If I started a compute-bound process at 0200 on December 24, 1968, how much CPU time would it have used by 0300? What if I were buying better response plan # 2 at that time?"

"Will my attemped reservation of 2 peripheral oil wells for April 1, 1970 be confirmed?"