

Identification

Overview of Restart

Robert L. Rappaport, Michael J. Spler

Purpose

The Traffic Controller multiplexes (time-shares) the system's processor resources by allotting each process a certain time-quantum which is kept in the hardware's timer-register; when the time allotment is depleted, a process interrupt is generated which results in a call to the Traffic Controller's entrypoint restart. Restart allows the currently running process to reschedule itself for future running and to give the processor away.

Discussion

A running process always has a current time-allotment; Normally, calls to the Traffic Controller do not affect ~~it~~ it (the only exception being a call to block where the 'interaction_switch' is set to 'on'.) Whenever a process willingly abandons its processor (wait, block, quit), its remaining time-allotment is saved in that process' Active Process Table (APT) entry, to be restored when the process is made to run again.

This ~~technique~~ technique insures that the scheduling algorithm remains ~~independent~~ independent of, and unaffected by, the rest of the Traffic Controller's functions.

When the process runs out of processor time, a fault is generated which is transformed (by software) into a process-interrupt; the interrupt diverts the execution of the currently-running process into the process-interrupt handler which in turn calls restart.

Arrival in restart means, that the process has now completely exhausted its last ~~an~~ processor-time allotment, and that it should 'out of consideration

for other processes' abandon the processor and allow some other process to run. In order to do this, the process should perform the following steps:

1. Do its own accounting, add the last amount of used-up time to its processor-usage meter and look to see whether or not it still has processor-time resources left. If it has no more time left, the process must initiate its automatic-logout ~~from~~ mechanism.
2. Reschedule itself. This operation involves the assignment of a level number in the redy-list for this process, and the computation of a new time-allotment.
3. Put itself on the ready-list (declare itself to be ready to run.)
4. Give the processor away to the process which is topmost on the ready list (and which might, conceivably, be this very same process.)

The scheduler is described in section BJ.5.1, the metering module is described in section xx.xx.xx.