7 Dec '70

Chapter XXimvlde of the
Continuing Interrupt Hassle

We're all aware of rumblings of discontent with the current interrupt structure and several people have seen fit to propse sweeping alterations in ~~the~~ that structure. ~~currently implemented.~~ I think of the interrupt structure as fulfilling two widely different ~~xxxxxxxxxxxx~~ categories of tasks. First, tasks without which we cannot write the system as currently envisioned. Second, ~~xx~~ if a useful, coherent structure can be evolved to handle system necessities, it would be nice to make it available for non-essential tasks. A poll of available staff gives the following list, to which additions are ardently solicited:

A) SYSTEM NECESSITIES
1) Major/minor panic from TTY
2) Initiation of forced swapout
3) Accounting interrupts (eg, too much ecs time*space)
4) Forced logout for system shutdown
5) Timer interrupts

B) USER TOYS

Arguments that a particular item doesn't belong in list A will in general ~~xx~~ only be heard if the arguments are given in a quiet tone of voice (or in writing), and if they are accompanied by a fairly detailed method of implementing the feature in some other fashion. (Item A1 is a necessity in the sense that without it, the system would be hideous.)

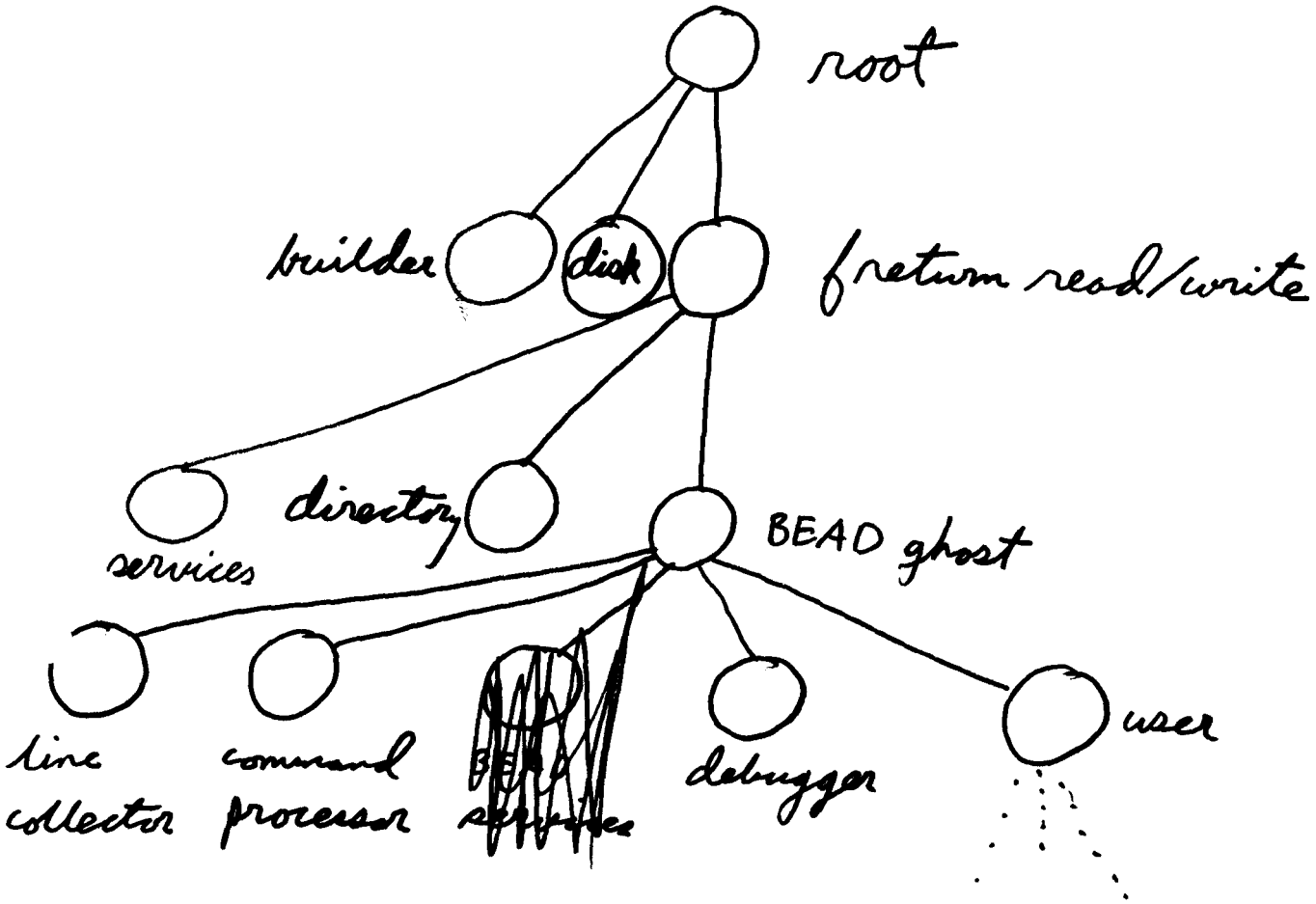
The only one of these which has been tackled in detail, to my knowledge, is A1 with which Howard has been valiantly ~~xx~~ struggling for the last few weeks. He ~~xx~~ claims that the objectives he has specified for cleaning up the call stack and getting to a debugger and suchlike other things cannot be implemented with the current logic. And he has an extensive proposal for a redesign. There is at least one other (partial) proposal in the air, namely , Bruce's "linear interrupt priority" scheme.

I would very much like to avoid the situation where a new implementation of interrupts is coded for Howard which turns out to fail to handle the other cases, so I want to provoke at least minimal discussion of the other system necessities before ~~xxxxxxxxxx~~ coding of second-generation interrupt ~~s~~ is begun.

the system

Projected Process Profile
Disk

7 Dec '70



Notes

1) Bead services may be moved to some of fretum R/W to avoid being interrupted

7 Dec '70

Random Ramblings

1 TTY interrupts are aimed at the BEAD ghost

2 BG runs with interrupts continuously inhibited
   a panics won't take while BG is running & until it ceases being the top of the stack
   b if more than 1 interrupt arrives while BG is on the top of stack, all but the first ill be lost

3 Some system routines are protected from panic interrupts by the priority scheme; since no current proposal extends protection from caller to callee, the disk's potential acces to the line collector, et al, is gravely complicated.

4 System routines below the BG run with interrupt always armed; they protect themselves occassionally by setting the inhibit bit

5 Loops in the directory system can lock out panics for arbitrarily long periods of time (roughly controllable by a parameter specified how and by whom?)

6 The time required by the disk is unknown?, so delays in panics due to the disk are an unknown quantity

7 The forced swapout interrupt must be aimed at the f-return r/w node, or above*

8 Accounting interrupts and system shutdown can probably be aimed at the bead ghost**

9 Howard's algorithms depend heavily on the tree-scan feature of processing call-with-interrupt type interrupts; it would be nice if external interrupt processing were consistent, but it's out of joint with Bruce's linear scheme.

10 It is impossible to imitate the tree structure priority scheme with the linear scheme.

11 If more than 1 interrupt arrives at a given node, only the first is guaranteed to take. Consequently, you aim different kinds of interrupts at one node only at peril of losing interrupts. (So 8 won't work.)

* Howard objects

** VV objects, see 11

7 Dec '70

ECS OPERATION TIMING

Operation	System time, μs
DISPLAY USER CLOCKS	273
WRITE 1 WORD, shape (2,2,2,1)	465
WRITE 8 WORDS, "	1313
WRITE 1 WORD, shape (8)	411
WRITE 8 WORDS, "	414

Tune in next week for MAP FLAPS and SUBPROCESS CALL/RETURN !!