Copy to: Daley
Clingen
Saltzer
from Corbato

Oct. 30, 1968

Performance issue

Segment size = 65K ?

1. All seg have same max length, hence remove issue from user interface; elimin SETML
2. Temp. Probs. of FL/I, Ftn, PoP's, flush(!); lisp; GE marketing; Gecos compat., PL/I
3. Can still have expensive way of getting 256K seg w/ 4 contiguous ASTE ??
4. AST can be in 2 parallel arrays ~~segments~~ w/ page tables in one and indexing, not alloc + free
5. segment ~~page~~ fault time speeds up because no allocation of page tables
6. ss " " " " " no " " ASTE
7. Space of AST shrinks by maybe a factor of ~~(2)~~ 2
   a. alloc + free headers vanish ~~[illegible]~~ 4 words/entry
   b. Information drops out of ASTE (4 words/entry)
   c. No longer any breakage
8. Speed of handling AST goes up because of better AST structure
9. Reduced statistical effects because only descriptor segments in any sort of pool
10. AST hashtable ~~[illegible]~~ disappears in explicit form and becomes integral to AST w/ more space saving
11. Estimate half of code in segment control eliminated
12. Speeds up directory control primitives indirectly via speed up AST manip. and in fact anyone who touches AST speeds up
13. Directory control can now tie together ~~up~~ branch + file map and avoid alloc. + free
14. page ~~table~~ table creation time reduced by avoiding Do loop over 64 entries setting pointer to ASTE
15. Can put page tables in a common segment to save space.