To: Distribution
From: M.A.Padlipsky
Subject: Meetings 3 & 4
Date: 3/27/72

FLASH!!!! Just before these minutes were input, Webber discovered an index register clobbering bug in the clear associative memory code which would cause the "backing up" of a page and should account for the locking bugs, the bad trailers and the bad SDWs. Just for completeness, though, I'll give the rest of the report as written, in case we haven't really stumbled over the philospher's stone and need to remember the planned traps.

The big news Saturday was that the PML bug had been found. The re-entry trap in the threading subroutine went off some 10 minutes after we came up and that led to the discovery that the routine was after all being used without the global page table lock's being set. It is now set. Score one.

Consensus was to go after locking problems next, particularly because the "page backing up in time" hypothesis would also account for the bad trailers and SDWs. 1) Actions taken: Snyder has been eyeballing page control; special emphasis to be placed on reset_working_set, of which Webber is particularly suspicious. The paging history of the IOAT (most common source of locking problems) will be kept in a ring buffer; both caller and time will be recorded, in order to attempt to get a handle on the "backing up". The quantum will be reduced to induce more pre-page / post-purge action to force the problem. Operations will be instructed to call one of the dump experts when there's a crash so that the secondary storage copy of either the IOAT, str_seg, or involved directory can also be dumped (for comparison with the in-core copy). 2) Actions planned: tracing of ASTE trailer manipulation will be performed; time of paging to be kept in header for comparison with time of change kept in entry can prove "backing up" hypothesis. Jordan's new lock given priority, should go in after Tuesday morning special session; it keeps per-system history of calls to lock.

The other problem actively pursued is null PTWs. All (but one?) generators of null addresses have been tagged. Current segment length is being checked.

Two on-line experiments have been performed. Morris ran two processes sharing a segment (reading and writing) hoping to force a backing up problem; no joy. This could be repeated with calls to put the processes to sleep to insure deactivation of the shared segment. I ran 6-7 million STAC instructions (in one process) just to see if the opcode could be trusted; it could.

The PML traps will be left in for a day or two, in case any other problems crop up.

Shared segments turn out to be post-purged; consensus is that this strategy should be reviewed.