

XEROX

BUSINESS SYSTEMS

Systems Development Department

To: Bill Kennedy Date: August 9, 1978

From: H. C. Lauer Org: SDD/SD

Subject: 32 Bit D0 System Identification Number Filed: [Iris] <Lauer > D0SystemIdentifications.memo

Reference: 32 Bit D0 System Identification Number, dated August 8, 1978 from Bill Kennedy to Bob Bosse

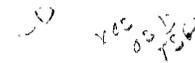
I did agree with you that the 32-bit number stored in the EPROM of a D0 could be called the "System Identification Number." I did *not* agree to calling it a "SIN." Pilot will not use the term "SIN" nor will we learn to recognize it. I suggest that if you wish to abbreviate "system identification number," try "SystemID." (Actually, we use the term "ProcessorID" internally.)

With that minor point out of the way, let me comment on the last paragraph of your memo. Pilot regards the System Identification Number as private. No client is permitted to know it. Furthermore, even if a client can discover it, there are no Pilot facilities that accept this number as parameter. All clients deal with Pilot and with other clients on other system elements using File Capabilities and Network Addresses (both of which are derived from the System Identification Number). Note also that Pilot may substitute a fake system identification number in lieu of the one obtained from the EPROM. This occurs, for example, when OIS communication takes place over the Ethernet. This decision is invisible to clients.

Even at initialization time, system elements introduce themselves to each other without a client having to specify system identification numbers. Pilot provides special facilities to establish high-level communication among clients for the first time. Thus it is never necessary that a Xerox representative or customer installation manager has to know the system identification number of any particular system element. Above all, *no client program of Pilot is permitted to ask him or tell him (or any other user) this piece of information.*

The reason we impose this restriction, and the reason we are so adamant about it, is that client programmers who try to make use of the System Identification Number are programming at too low a level. In particular, they would be binding their designs and code to implementation details and decisions which Pilot reserves the right to change without notice. We provide, instead, the higher-level concepts such as Network Addresses and uniform higher-level mechanisms to cope with invalidating and reissuing them as a result of changing EPROMs, moving resources, changing software, etc. These concepts and mechanism form the framework upon which Clearinghouses, Star Directories, and other client- and user-oriented resource naming and management.

We strongly recommend that you do not, *as a matter of routine*, make humans aware of the System Identification Number in your diagnostics which run outside of Pilot. We also recommend you avoid designing your control systems based on the need or desire to know it and/or type it into any program. (Obviously, some knowledge of the System Identification Number is necessary in those special cases where a fault is suspected to involve it, but these are few and far between.)


c: Bob Bosse, T. Kelly, R. Metcalfe, R. Pierce, H. Scharmann, C. Thacker, J. White, W. Lynch