

Xerox Meeting Notes:

The following are notes on a meeting held, on 15-FEB-72, at Xerox PARC, attended principally by Bill Paxton, Charles Irby, Peter Deutsch, Jim Mitchel, Bill English, and Butler Lampson, to discuss:

PARC's statement of work regarding IMLAC level protocol for running NLS over the NET, and

a possible PDP-10 -- PDP-11 configuration to run NLS for ARC and the relationship between this configuration and xNLS on a mini-computer.

Regarding the Imlac level protocol, PARC would just as soon not have been given the responsibility for developing this protocol. Larry Roberts apparently saw an opportunity to get this done with minimal hassle and took advantage of the situation. They would gladly give the responsibility for developing and publishing this protocol to us since its initial use will be for running DNLS over the ARPANET. We agreed that it would be a joint effort anyway, but that it was unfortunate that Larry Roberts had seen fit to do things this way, since its initial use involves NLS which is principally an ARC responsibility.

Regarding the possibility of a PDP10-PDP11 configuration at ARC and its relationship to xNLS:

I have again raised the issue of supporting terminals with small computers which would do the command interaction with the users, sending fully specified command requests to the PDP10, which would have its scheduling mechanisms set to favor heavy compute jobs.

The interface between the PDP10 and PDP11's would involve message sending and confirmation, much like the ARPANET. (In fact, one of the ways in which they could be connected together is through the ARPANET. Note that there could be more than one PDP11, and that they would not have to be local.) The PDP11 would run a simple operating system (which we would probably NOT write) with no swapping and no file system and would run jobs for NLS users.

The NLS running on the ten would specify which job the eleven should run. This allows us freedom to run several versions of NLS concurrently. A set of PDP11's would run all of our terminals. A net user could connect to a PDP11 and be a tty or to a PDP10 and be PDP11-like. The interface looks simple and clean, with no new language development.

Writing the program to run in the PDP11's, and separating the interactive command specification portion of NLS from the non-interactive file manipulation portion would be quite straight forward, since NLS is already organized in that way for the most part.

The interface consists of two sets of procedures, between which an arbitrary protocol could be added.

The measurement and analysis of our system, which will be taking place over the next couple of months, should indicate whether or not using a small computer to do the interaction with the users is the appropriate thing to do. I think it is reasonable at this time to look into the feasibility of doing such a thing. In considering the possibilities, the following questions occurred to me.

Which small computer would be best for such a task?

Which have reasonable operating systems?

Which have reasonable languages?

What about the availability, reliability, and maintenance?

How might the machine be connected to the PDP10?

Through the ARPANET?

Through the I/O bus?

Through the memory bus?

What about being compatible with the PARC choice for an XNLS computer?

Have they made a choice yet?

If it is up in the air, will our needs help them decide?

How does this relate to XNLS on a small computer?

When will MPL be available on a small computer?

Does it seem desirable for XNLS to adopt the same approach?

Try putting the interactive portion in a small machine between the XNLS computer and the display system or in the display system itself, if there is excess resource available.

When will XNLS be operational?

Should we wait until then?

What about the PDP11 BLISS which compiles on a PDP10 -- being developed at Carnegie (I think)?

What about operating systems that now exist for PDP11's and NOVA's, such as the one developed at U of Ill for the PDP11.

What about the energy DEC is pouring into the PDP11?

What about a net interface?

Will PARC build their own?

What about the one used at U of Ill for the PDP11?

Do we or PARC intend to support MPL on other machines?

Other network sites might be interested in buying an NLS front end.

Would this be in competition with XNLS?

Would the use of some other language be better, since we wouldn't have to maintain the language?

Or, would the use of MPL be better because it would encourage people to experiment with the NLS front end by adding new modules or replacing old ones?

The discussion at PARC did not really suggest answers most of these questions.

Since nls now consists of two modules, interactive and non-interactive, with two sets of procedures representing the two sides of the interface, how should we proceed?

We could use a PDP11, connected to ten through the I/O bus or through the network, use bliss and possibly the operating system and net interface developed at U of Ill.

We could choose to wait for NLS in MPL, with MPL and an MPL operating system running on a small machine -- all necessary for XNLS.

If it looks like it will be a year or longer before we could do this, I would probably choose to proceed sooner than that (we are severely overloaded, our user community is growing, and I do not think we can expect to get more than 10% to 15% more out of the PDP10).

This seems mostly to be a question of timing and energy required to do it without MPL and PARC help.

The following tentative schedule for next few months, regarding MPL and NLS

Rewrite MPL in MPL: mid FEB to mid MARCH

Design data typing facility for MPL: mid FEB to mid MAR

Implement data type facility: mid MAR to ?

Start design for NLS: mid MAR to ?

CHI, 22-FEB-72 10:06

<GJOURNAL>9219.NLS;1 4

start implementing NLS in MPL: JUL to ?

MPL on mini: ?

MPL operating system: ?