

Inter-Office Memorandum

To	Distributed Computing	Date	August 8, 1976
From	David Boggs	Location	Palo Alto
Subject	Pup Network Directory Update Protocol	Organization	Parc

XEROX

Filed on: <Pup>NetDirUpdate.bravo

This memo documents the protocol by which name lookup servers update their local copies of the network directory.

Disclaimer

This version of the protocol assumes that the network directory is maintained at galactic central, from which all servers get copies. This scheme is cheap and easy to implement for small galaxies, but as civilization spreads: (1) it will become cumbersome: to add or change a listing, one must interact with galactic central which may be many light-years away in space with a consequent many year delay in time before the listing propagates out to the periphery (besides, galactic central will be a huge bureaucracy, slow in responding to requests) (2) the directory will get very large: everyone in the whole galaxy will be listed in each copy of the directory, which will strain the capacity of small servers to store and search in reasonable time. In short, *it is centralized*.

The eventual solution is, of course, to *distribute* the network directory. A server's network directory should be a *cache* containing only those names most likely to be referenced. If a request arrives for a name not in the cache, the request should be passed on to other name lookup servers.

The Protocol

When a Pup of type `NetDirVersion` arrives at the miscellaneous services socket, the version number in the Pup should be compared with the version number of the local network directory, and one of the following actions taken:

If the local directory version number is greater than the version number in the incoming Pup, a reply of type `NetDirVersion` should be generated with the local network directory version number as data.

If the local directory version number is equal to the version number in the incoming Pup, do nothing.

If the local directory version number is less than the version number in the incoming Pup, probe the inter-network to find the server with the latest version, (the version advertised in the incoming Pup is newer than your version, but it may not be the newest) then ask him to send it to you. See below for a suggested implementation to probe for newer versions.

When a Pup of type `SendNetDir` arrives at the miscellaneous services socket, open an EFTP Socket with the foreign port set to the port specified in the incoming Pup, and send the local network directory.

Periodically (every 6 hours or so) probe the inter-network for newer versions of the network directory.

Probing for newer versions

Open a level 1 Pup socket and broadcast a Pup of type **NetDirVersion** to miscellaneous services on all directly connected networks. Remember the address of the host which responds with the highest version number. If the highest version number is less than or equal to the version number of the local directory, give up. If a newer version is located, start an EFTP receiver (do not use the default EFTP receive socket), and send a Pup of type **SendNetDir** with the EFTP socket as data to the host which responded with the highest version number. If the EFTP receiver times out, the **SendNetDir** Pup should be retransmitted a few times before giving up. If a file is successfully received, it should be installed as the new network directory.

Packet types and formats

NetDirVersion: Pup type #240

Data: 2 bytes containing a 16 bit version number.

SendNetDir: Pup type #241

Data: 6 bytes containing a port to which the recipient should EFTP his copy of the network directory.