



Tech Info Library

AppleTalk: Worldwide Network Issues

This article last reviewed: 28 January 1991

TOPIC -----

I have three questions involving AppleTalk on a worldwide network at a top investment bank:

- 1) Are file accesses across the net controlled by AppleShare, or is something else controlling the file access?
- 2) When the packet-switched network re-routes lines, we can experience delays of 5 minutes or more. The connections re-route, but usually we seem to lose our network file connection. What happens to an AppleShare request when its response is delayed for that long? We saw (using a datascope on the line) that if the request for servers timed out, the response was ignored. Does a similar situation take place in the case of a read?
- 3) Does the modification to the GNRL resource affect only the initial call? If the server becomes available quickly, does that shorten the time-out that AppleShare will use for all server access while the server is mounted?

DISCUSSION -----

- 1) AppleShare (AFP calls) controls all access to the remote files, including the handling any time-out values associated with the function of accessing a file. By "controlled", we assume you mean any function that pertains to successfully opening, reading, or writing to a file stored on an AFP server. There are many AppleTalk protocol functions associated with AFP transactions that are not directly considered a controlling aspect of manipulating a remote file. These functions do not control access to the individual files so much as they control access to the media being using to transport the AFP calls.

If you are using standard file system calls and if the file you're trying to access is located on an AFP server, all local file system calls such as PBOpen, PBRead, FSRead, etc. are filtered out and transformed into equivalent AFP calls.

- 2) The 5-minute delay is going to present a real problem to AppleShare, or

to be more precise to the ASP protocol. The session timer for an AppleShare file server connection is set to 2 minutes. Once this timer expires, the session will terminate, and you will lose your server connection. The retry count for the read request is set to infinite, so a read request never really times out.

The session timer is restarted each time a tickle packet is received by either the workstation or the server. Tickle packets are sent out every 60 seconds by both the server and the workstation as part of the ASP protocol, to allow a way of detecting that the other end of the connection has gone down. There is no easy way to reset this timer value.

- 3) If you modify the GNRL resource in the AppleShare chooser document, it will affect every NBP lookup that is done from the Chooser. The Chooser uses these values to determine the NBP lookup interval and retry values for the current NBP transaction. The default of 0705 tells the chooser to send 5 NBP lookup requests at an interval of 7/8 second. This process is repeated in an infinite loop until the user closes the Chooser.

Chooser Event Flow Example:

```
User opens chooser and selects the AppleShare CDEV
GNRL resource -4096 loaded = (5002)
NBP look up mechanism started
```

NBP Loop:

```
Get NBP ID for this transaction
```

```
(Note: All NBP request and replies for this loop will use
this ID)
```

```
Send first look up (NBP ID = "New")
```

```
Collect and display responses from the NBP look up ID "New"
Wait 10.6 seconds
```

```
Send second look up (NBP ID still = "New")
```

```
Collect and display responses from the NBP look up ID "New"
Wait 10.6 seconds
```

```
Discard all buffers and data associated with NBP ID "New"
```

```
(Note: If a response is received for NBP lookup ID = "New"
after this point the reply data would be discarded and the
device would not be added to the list in the Chooser.)
```

```
Do some other misc. cleanup (approximate time, 1 second)
```

```
goto NBP Loop
```

```
End NBP Loop:
```

With the retry timer set to such a large value, the multiple retry count is really not necessary. On the other hand, it doesn't hurt anything either, and it effectively increases the time you'll wait for NBP replies to over 20 seconds for the current transaction. The idea behind

the retry count is to send several lookup requests in quick succession (default is less than 1 second) in case there are devices which cannot respond, because they are busy or because the previous packet never reached them.

The reason that increasing the interval timer helps in the case of remote servers is directly tied to the way the NBP mechanism works. The chooser maintains only 1 NBP lookup request at a time, tracking all replies to that request by way of the NBP ID mechanism. Replies that are received that do not match the current request ID are discarded. The request ID is maintained only for the current NBP request; the interval and retry counters for this request can be tuned via the GNRL resource. In other words, if you set the retry counter to 10 and the interval timer to 50, the NBP ID would be maintained for 10 requests at an interval of 10.6 seconds. The GNRL resource is documented in "Inside Macintosh, Volume 4," page 216.

Copyright 1991 Apple Computer, Inc.

Keywords: <None>

=====
This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 6715