New Technical Notes
Macintosh
®
Developer Support

Name Binding Protocol Q&As

Networking

Revised by: Developer Support Center May 1993
Written by: Developer Support Center October 1990

This Technical Note contains a collection of Q&As relating to a specific topic—questions you've sent the Developer Support Center (DSC) along with answers from the DSC engineers. While DSC engineers have checked the Q&A content for accuracy, the Q&A Technical Notes don't have the editing and organization of other Technical Notes. The Q&A function is to get new technical information and updates to you quickly, saving the polish for when the information migrates into reference manuals.

Q&As are now included with Technical Notes to make access to technical updates easier for you. If you have comments or suggestions about Q&A content or distribution, please let us know by sending an AppleLink to DEVFEEDBACK. Apple Partners may send technical questions about Q&A content to DEVSUPPORT for resolution.

Where to find AppleTalk Name Binding Protocol code sample

Date Written: 10/30/90
Last reviewed: 2/20/91

Do you have AppleTalk sample code that uses NBP (Name Binding Protocol) to check for a name and type on the network and, if not found, register the name and type on the network?

___

The sample you've requested is available on the Macintosh System 7 CD.

Version tests before using NBP's RemoveName under A/UX

Date Written: 11/1/90
Last reviewed: 2/20/91

When I was running my program under A/UX 2.0, the AppleTalk NBP (Name Binding Protocol" routine RemoveName seems to do nothing. Beta manuals hint that LaserWriter printing works but other parts of AppleTalk are absent or iffy. Am I right in assuming that RegisterName works but RemoveName doesn't? Is it safe to check for AppleTalk version 0 for this purpose? When will RemoveName work under A/UX?

We recommend checking the A/UX version instead of the AppleTalk version. Good recommendations about how to do that are found in the Macintosh Technical Note “A/UX

System Calls from Macintosh Software." If you use Gestalt you could also check if the release level is 2.0. Now that this problem is fixed in 2.0.1, you can check for release level and enable your functionality.

Macintosh NBPLookup badUnitErr conditions

Date Written: 11/14/90
Last reviewed: 2/8/91

Why am I getting a badUnitErr as a result code from NBPLookup? This error is not listed as a result code for NBPLookup, nor anywhere in the AppleTalk Manager.

___

You’re probably getting the badUnitErr result because you don’t have the AppleTalk drivers open. Since AppleTalk calls are made to a device driver (the AppleTalk driver), you can get back general operating system errors from the driver.

See the Macintosh Technical Note “Opening AppleTalk” for a description of how to open the AppleTalk drivers.

AppleTalk 56 NBP reply packet entity field checks

Date Written: 6/14/91
Last reviewed: 10/15/91

As stated in *Inside AppleTalk,* each entity field in an NBP reply packet can be a maximum of 32 characters. In previous versions of AppleTalk, no range checking was done on entity field lengths, and often names that were too long were used, which eventually caused a crash somewhere else in the protocol stack.

Likewise, a null character in the object field of an entity has never been OK; it just happened to work because no range checking was done. (The only place that a null can legally be used in an NBP tuple is in the zone name field of an NBP LkUp, FwdReq, or BrRq packet; see page 7-17 of *Inside Appletalk* for more information.)

Now, under Appletalk 56, a range check is done on each entity field to ensure that the length is legal (1<=length<=32). When it finds a packet that contains a “corrupt” entity name, that packet is discarded.

Name Binding Protocol (NBP) interval and count fields

Date Written: 9/13/91
Last reviewed: 10/8/91

Is there a limit on the values set in the Name Binding Protocol (NBP) interval and count fields when used with PLookupName and PConfirmName calls? How do the interval and count work? If a device is not on the network and I send a PLookupName with interval = 20 and count = 20, will I wait 400 seconds before PLookupName returns?

___

Since the interval and count parameters for NBP calls are both 1-byte, the values used are limited to the range of 0-255 ($00-$FF). Here’s what the values do:

- interval = retransmit interval in 8-tick units. This value is used to set up a VBL task. A value of 0 should not be used because that would mean the VBL task would never be executed and would be removed from the VBL queue.

- count = total number of transmit attempts. Each time the interval timer expires, this value is decremented by 1. When it reaches 0, the NBP call completes. So if a value of 0 is used, the packet will be retransmitted 255 times (or transmitted 256 times).

Three things can happen to make the LookupName, RegisterName, or ConfirmName calls complete:

- PKillNBP can be called to abort one of the calls (see *Inside Macintosh* Volume V, page 519).

- maxToGet matches are returned or the return buffer is filled. Here's how this works: Each time an NBP lookup reply (LkUp-Reply) packet is received, an attempt is made to add all the NBP tuples found in that LkUp-Reply packet to the return buffer. If all the tuples cannot be added to the buffer because there isn't enough room, the call completes with as many tuples as could fit and the numGotten field will contain the number of matches in the buffer. If all the tuples from the LkUp-Reply packet are added to the buffer, numGotten (the number of matches in the buffer) is compared to the value passed in the maxToGet field. If numGotten is greater than or equal to maxToGet, the call completes and the numGotten field will contain the number of matches in the buffer. Since the buffer can fill before maxToGet matches are received and since LkUp-Reply packets can return multiple tuples, you may get more or fewer matches than you asked for with maxToGet.

- The count is decremented to 0. You can use this equation to determine how long the call would take to complete this way:

```
IF count = 0 THEN count := 256;
TimeToCompleteInTicks := count * interval * 8;
```

The RegisterName and ConfirmName calls always complete after they receive the first LkUp-Reply packet to their request, so you could look at them as always having a maxToGet of 1 (maxToGet is not one of the parameters for those two calls).

AppleTalk PLookupName interval and retry values

Date Written: 1/6/92
Last reviewed: 1/27/92

What are recommended values for retry interval and retry count when using the AppleTalk NBP call PLookupName on a complicated internet?

___

You might want to start with the NBP retry interval and retry count values Apple uses for its Chooser PRER and RDEV device resource files. The Chooser grabs these values from the PRER's or RDEV's GNRL resource -4096:

Device	Interval	Count
LaserWriter	$0B	$05
AppleTalk ImageWriter	$07	$02

AppleShare	$07	$05
If no GNRL resource	$0F	$03

Apple’s engineering teams found these values to work well in most situations.

The count value should be based on how likely it is for the device to miss NBP lookup requests. For example, the AppleTalk ImageWriter has a dedicated processor on the LocalTalk option card just to handle AppleTalk, so its count value is low; most Macintosh models and LaserWriter printers depend on their 680x0 processor to handle AppleTalk along with everything else in the system (the Macintosh IIfx and Macintosh Quadra models are exceptions to this), so their count value is higher.

The interval value should be based on the speed of the network and how many devices of this type you expect there to be on the network. On a network with very slow connections (for example, one using a modem bridge), or in cases where there are so many devices of a particular type that lots of collisions occur during lookups, the interval value should be increased.

Apple puts these values in a resource because not all networks and devices are alike. You should do the same (put your interval and count in a resource so that it can be configured).

Don’t access the MPW NBP EntityName field directly

Date Written: 1/1/91
Last reviewed: 1/1/91

When I fill in the fields of MPW’s Name Binding Protocol (NBP) EntityName structure, AppleTalk doesn’t recognize the entity, even though I know it’s out there. What’s going on?

The real definition of EntityName is 3 PACKED strings of any length (32 is an example). No offsets for Asm are specified since each string address must be calculated by adding the length byte to the last string pointer. In Pascal, String(32) will be 34 bytes long since fields never start on an odd byte unless they are only one byte long, so this will generate correct-looking interfaces for Pascal and C, but the interfaces will not be the same. This is OK since they are not used.

The point is that you should never try to access the fields of the EntityName field directly. The only reason the type is defined at all is so that you can allocate EntityName variables that will hold the largest possible EntityName. To fill in an EntityName record, you call the NBPSetEntity routine.