Internet Protocol Address Management Using the FastPath

Introduction

The **FastPath** supports 3 alternative configuration schemes: AppleTalk-only routing, K-Star (KIP Style Addressing and Routing) and IP Subnetting. For those sites which require Internet Protocol (IP) routing, Kinetics recommends the use of K-Star. However, some sites will find it more convenient to use the IP subnetting approach.

This document is directed to network administrators responsible for maintaining **FastPath** Gateways and networks of Macintoshes in an IP network. We present the basic knowledge needed to plan and implement the integration of AppleTalk and IP networks.

This document is intended to make the task of configuring your network using IP subnetting easier.

The Basics

It is helpful to understand the similarities and differences between addressing in the AppleTalk domain and in the IP domain.

- Both IP addresses and AppleTalk addresses are made up of a network number and a host number (AppleTalk calls the host number a "node" number).
- IP uses fixed network numbers and static assignment of host addresses.
- AppleTalk also uses fixed network numbers, configured into AppleTalk bridges, but dynamically assigns node numbers at system startup time.[1]

Usually, a LocalTalk/IP (LocalTalk/Internet Protocol) network must be added to an established IP network. Besides K-Star, Kinetics offers two approaches for connecting a LocalTalk/IP network to the existing IP domain: ***Subnetting*** and ***Fixed Routing.***

- ***Subnetting*** allows the LocalTalk/IP network to be treated as a subnet, or a sub-section of an IP network. This approach does not require any reconfiguration of established IP hosts.

- ***Fixed routing*** allows the LocalTalk/IP network to be different from the existing Ethernet IP network, but requires additional configuration of established IP hosts.

A detailed discussion of these two approaches will be presented after a general discussion of IP addressing.

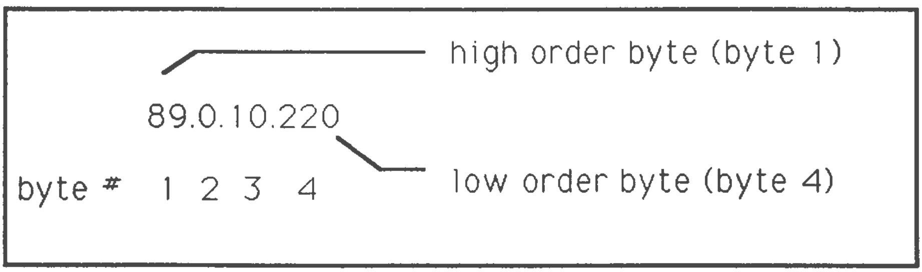
We will be referring to binary, decimal and hexidecimal notation throughout this document. A conversion table is provided in *The FastPath Installation Guide*.

Internet Addressing

An Internet address is made up of four bytes. There are several common ways to write an Internet address, but for our purposes we will use just two: a single hexadecimal number or "dot format" where each of the four bytes is separated by dots and expressed as decimal numbers.

Hexadecimal representation:	59000ADC
Dot representation:	89.0.10.220

Note: We will use the following notational convention: bytes are numbered from high order byte (most significant) to low order byte (least significant), left to right.



IP Network Classes

An Internet address is made up of two portions: the ***network*** portion and the ***host*** portion. The network portion is fixed across a single Internet network. Each host on a given IP network must have a distinct host address.

The ***class*** of the network determines which part of the address represents the network number and which part represents the host number. There are three classes of Internet networks. The class of a network is determined by examining the upper byte (byte 1) of the Internet address.

- ***Class A*** network - an upper byte with decimal values in the range 1 to 127
- ***Class B*** network - values in the range 128 to 191
- ***Class C*** network - values in the range 192 to 254

In summary:

Byte Value		Network Class	Possible # of nets
1 - 127	=	Class A network	127
128 - 191	=	Class B network	16,384
192 - 254	=	Class C network	4,128,768

The network and host portions of the Internet address are determined from the class of the network.

- Class A network- byte 1 is the network portion, and bytes 2-4 are the host portion.
- Class B network- bytes 1-2 are the network portion and bytes 3-4 are the host portion.
- Class C network- bytes 1-3 are the network portion, and byte 4 is the host portion.

Refer to the figure below for a pictorial representation of the classification of networks.

Class A Address

1 - 127	0 - 255	0 - 255	0 - 255

Class B Address

128-191	0 - 255	0 - 255	0 - 255

Class C Address

192-254	0 - 255	0 - 255	0 - 255

= Network Portion   = Host Portion

Important note: In the bytes marked as host portions of the address, at least one of the values must be non-zero and not all of the values can be 255. These are reserved for IP broadcast address.

Examples of Network Parameters for each Class:

Example 1:
89.0.128.3 is a Class A address because
1 <= 89 <= 127.
The network portion of this address is 89 and the host portion is 0.128.3.

Example 2:
128.4.5.6 is a Class B address because
128 <= 128 <= 191.
The network portion of this address is 128.4 and the host portion is 5.6.

Example 3:
195.10.14.10 is a Class C address because
192 <= 195 <= 254.
The network portion of this address is 195.10.14 and the host portion is 10.

Summary of IP Address Assignment Rules:

1. An IP network is defined as *a collection of IP hosts all having the same network number.*

2. For each host on a given IP network, *the host portion of the address is unique.*

The Subnetting Approach to FastPath Routing

One approach to FastPath routing is to use IP subnets. An IP **subnet** is a logical sub-section of a single Internet network[2]. A subnet is some fixed-length portion of the host portion of an IP address. The subnet is used to subdivide a large network into smaller "sub-networks". The **FastPath** uses the "subnet portion" of an IP address to determine when to forward data from the LocalTalk to the Ethernet network, and data from the Ethernet network back to the LocalTalk.

How to define a Subnet

A subnet is defined by selecting a portion of the IP host address as the *subnet net number*, leaving the remaining portion as the *subnet host number*. We specify the subnet portion with a *subnet mask*. Some example subnet calculations will help illustrate this definition:

Example 1: Class A network = 18.192.127.51
If we set the subnet mask to ffff0000, then this IP address will be decoded as subnet host **127.51**, on subnet **192**, major net **18.**

Example 2: Class B network = 137.4.207.163
If we set the subnet mask to ffffff00, then this IP address will be decoded as subnet host **163**, on subnet **207**, major net **137.4.**

Example 3: Class C network = 210.167.3.250
If we set the subnet mask to ffffffc0, then this IP address will be decoded as subnet host **58**, on subnet **3**, major net **210.167.3**.

Understanding Kinetics Subnet Routing

Each FastPath configured as an IP subnet router is assigned two IP addresses, one for the Ethernet interface, and one for the LocalTalk interface. We will refer to these addresses as the **Ethernet IP address** and as the **LocalTalk/IP address**.

When we configure the **FastPath** as a subnet IP router, we define a single subnet, by specifying a subnet mask, which applies to both interfaces. We will refer to the resulting subnets as the **Ethernet subnet** and the **LocalTalk/IP subnet.**

This subnet definition allows the **FastPath** to route IP packets based on their subnet destination, using the following rules:

1. The **FastPath** will route a packet from the LocalTalk to the Ethernet network if the destination IP subnet is **different** from the **FastPath's** LocalTalk/IP subnet.

2. The **FastPath** will route a packet from the Ethernet network to the LocalTalk if the destination IP subnet is the **same** as the **FastPath's** LocalTalk/IP subnet.

Subnet Configuration Rules

In order for the subnet routing scheme to work, the network administrator must carefully follow the following configuration rules:

1. The Ethernet subnet must be **different** from the LocalTalk/IP subnet.

2. If the subnet definition, when applied to established IP hosts on the Ethernet, results in multiple subnets on the Ethernet, (i.e., not all Ethernet IP hosts are on the same Ethernet subnet as the **FastPath**) then *none* of these subnets must conflict with the LocalTalk/IP subnet.

3. The LocalTalk net number must be the **same** as the LocalTalk/IP subnet, and must be *non-zero.*[3]

Note that the second rule, if not adhered to, can result in mis-routing on the Ethernet. The **FastPath** answers Address Resolution Protocol (ARP) requests for addresses that match its LocalTalk/IP subnet, and if the LocalTalk/IP subnet is already in use on the Ethernet, then the **FastPath** will draw traffic for the conflicting subnet away from its true destination.

Rule three is to allow the IP module to create the AppleTalk destination for an IP packet bound for the LocalTalk, independent of the AppleTalk routing module.

Designing a Subnetting Scheme

Having presented the underlying concepts of IP addressing, and the Kinetics subnet routing method, it's time to apply this knowledge and work through some example subnet designs. We present two examples: for Class A using a single FastPath, and a Class C example using two FastPaths.

Once you decide on your subnet design, see the **FastPath Installation Guide, Chapter 3**, for instructions on how to send your IP set-up parameters to the **FastPath.**

Please note: If the hosts on the Ethernet IP network are UNIX Operating System hosts and are using the Berkeley 4.3BSD subnetting scheme, some additional considerations in choosing appropriate IP address values are necessary. Please see the section in this addendum: <u>*Internet Addressing and Berkeley Subnetting*</u>*, for routing in this type of environment.*

Example 1 : Class A Network

In this example, existing IP hosts on the Ethernet are on net **88**. (for example, Host 1 has an IP address of 88.0.0.1, Host 2 has an IP address of 88.0.0.2, etc). The administrator is adding a FastPath and must choose a subnet for the new LocalTalk/IP network.

Because the network is a Class A type, the administrator may choose any part of the lower (rightmost) three bytes (i.e. the host portion) to use as the subnet portion. He or she must make sure that the subnet portion of host IP addresses on the Ethernet are different from the subnet portion of the host IP addresses on the LocalTalk. Since byte 3 of the existing IP network is zero in the established IP address assignments, this is a convenient part of the IP address to use as the subnet portion.

Having chosen byte 3 as the subnet portion, we can specify the parameters for configuring the **FastPath**:

- the subnet mask,
- the Ethernet IP address of the **FastPath**
- the LocalTalk/IP address of the **FastPath**.

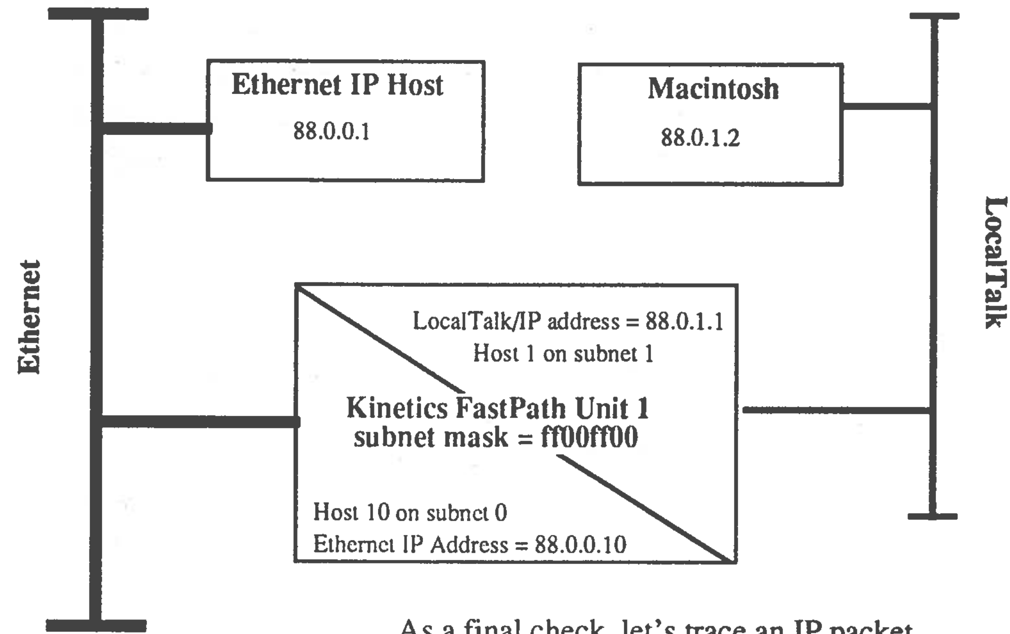
The subnet mask is easy: It will have a value of ff00ff00 because we would like to disregard the lowest eight bits of the IP address (i.e. the lowest byte), and we are only using eight bits for our subnet.

Next is the IP address for the Ethernet side of the

Next is the IP address for the Ethernet side of the **FastPath**. We are guided primarily by the addressing conventions on the existing IP network. The convention on our sample network is to assign addresses of the form 88.0.0.*x*, where *x* is some unused value. In our example, let us assume 88.0.0.10 is unused and let that be our **FastPath's** Ethernet IP address.

Last comes the assignment of the **FastPath's** LocalTalk/IP address. Using the two rules given above, we must assign an IP address of the form 88.0.*y*.*z*. Note that the value of *y* is the subnet portion of the address (i.e. byte 3) and must have a value **different** than zero since the nodes on the existing Ethernet IP network have a subnet equal to zero. After examining the existing Ethernet IP network, we can arbitrarily choose *y* to have the value one. The value of *z* is also arbitrary at this point since no other addresses on our LocalTalk/IP network have been chosen; let us choose *z* to be one. Therefore, our **FastPath's** LocalTalk/IP address is 88.0.1.1.

The following figure summarizes our sample network topology:



As a final check, let's trace an IP packet generated on the LocalTalk/IP net destined for a host on the Ethernet network.

For example, let the destination be address **88.0.0.1**, and let the source be address **88.0.1.2**. The **FastPath** would compare the subnet portion of the destination address, the third byte with a value of zero, to the subnet of the **FastPath's** LocalTalk/ IP address.

Therefore, we are comparing zero with one (i.e. the third byte of 88.0.1.1). Since the two subnet values are **different**, the **FastPath** will route the packet to the Ethernet network where host 88.0.0.1 will receive it.

If the Ethernet host, 88.0.0.1, decides to reply to the LocalTalk host, 88.0.1.2, the **FastPath** will examine the subnet portion of the destination IP address, 88.0.1.2, with a value of one, to the subnet portion of the **FastPath's** LocalTalk/IP address, which is also one, and, since they are the **same**, the **FastPath** will route the packet to the LocalTalk where Macintosh host 88.0.1.2 will receive it. Our **FastPath** configuration data is correct.

Example 2: Class C Network

In this example, we are designing a subnet routing scheme for net **192.1.1.** Established hosts on the Ethernet have addresses of the form 192.1.1.*x*, where *x* is between 1 and 30. At the current time, we have plans to connect only two **FastPaths** for evaluation purposes, and therefore require only two LocalTalk/IP networks.

Since we need a total of three subnets, the mask size must be at least 2 bits. This will give us a total of $2^2 = 4$ subnets, and $2^6 = 64$ subnet hosts[4].

Now that we've decided on our subnet mask size, we need to decide on where to position it in the space available for subnetting. Since the established IP host addresses fall into the range 1-30, we need only convert the range limits to binary to see what portion of the host address space is available for subnetting:

0 0 0 0 0 0 0 1 = 1 (lower)

0 0 0 1 1 1 1 0 = 30 (upper)

We see that the upper 3 bits remain zero throughout the range of current address allocations, and so decide to select **0** as our **Ethernet subnet.** Since we only require a **subnet size of 2 bits, we** decide to use only the upper 2 bits and therefore set the **subnet mask** to **ffffffc0**.

Having selected the mask parameter, we need only assign the **Ethernet IP** and the **LocalTalk/IP** addresses for the **FastPath**, and our configuration will be complete. For the **Ethernet IP** address, we need to assign an address on subnet **0, the Ethernet subnet**. We decide to assign the upper address in the range, which is:

subnet host

0 0 | 1 1 1 1 1 0 = 62 (upper bound of subnet 0)

Note: An address of all ones in the host portion is reserved as the subnet broadcast address.

For the **LocalTalk/IP subnet**, we can choose any of the remaining subnet addresses, so we chose subnet 1. Now we must assign the LocalTalk/IP address on this subnet. We again decide to assign the uppermost address on the subnet:

subnet host

0 1 | 1 1 1 1 1 0 = 126 (upper bound of subnet 1)

To summarize our configuration for the first FastPath:

subnet mask = ffffffc0

Ethernet IP address = 192.1.1.62 (host 62, subnet 0)

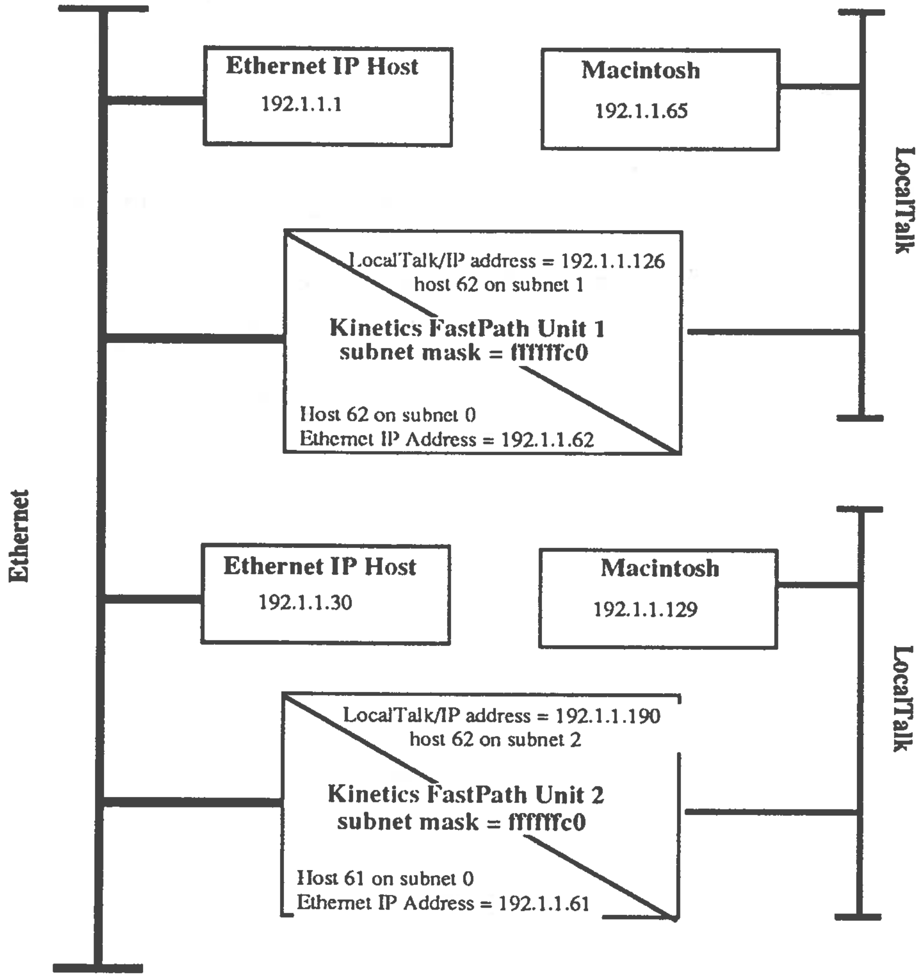
LocalTalk/IP address = 192.1.1.126(host 62, subnet 1)

LocalTalk/IP subnet = 1

FastPath Addendum

Using the same conventions to configure the second **FastPath**, except assigning the **LocalTalk/IP subnet = 2**, *and* assigning a unique **Ethernet IP** address on **Ethernet subnet 0**, we arrive at the following configuration:

subnet mask = ffffffc0
Ethernet IP address = 192.1.1.61 (host 61, subnet 0)
LocalTalk/IP address = 192.1.1.190 (host 62, subnet 2)
LocalTalk/IP subnet = 2



As a final check, let's trace an IP packet generated on the LocalTalk/IP net destined for a host on the Ethernet network.

For example, let the destination be address **192.1.1.30**, and let the source be address **192.1.1.65**. The **FastPath** would compare the subnet portion of the destination address, the upper two bits of byte 4 (= **0**), to the subnet of the **FastPath's** LocalTalk/IP address (= **1**). Therefore, we are comparing zero with one, and since the two subnet values are **different**, the **FastPath** will route the packet to the Ethernet network where host **192.1.1.30** will receive it.

If the Ethernet host, **192.1.1.30**, decides to reply to the LocalTalk host, **192.1.1.65**, the **FastPath** will compare the subnet of the destination (= **1**), with the subnet portion of the **FastPath's** LocalTalk/IP address (also = **1**). Since they are the **same**, the **FastPath** will route the packet to the LocalTalk where Macintosh host **192.1.1.65** will receive it. Our **FastPath** configuration data is correct.

The Fixed Routing Approach To FastPath Routing

The **fixed routing** approach, as its name implies, requires that Ethernet IP hosts maintain explicit routing information in order to communicate with LocalTalk/IP hosts. The advantage of fixed routing over subnetting is that it allows the LocalTalk/IP network to reside on a different Internet network, thus avoiding the complexity and restrictions imposed by subnetting. For example, subnetting becomes difficult when an existing IP network is a Class C network, because the host portion of the Internet address is only one byte. Using fixed routing, we can place the LocalTalk/IP network on a different logical network and may even place it on a different class of network (i.e., on either a Class A or Class B network). The penalty to be paid for avoiding subnetting is that each Ethernet IP host must be provided a method for reaching the LocalTalk/IP network since the LocalTalk/IP network has a different network portion in its Internet address.

There is one subnet restriction that is imposed on Internet addresses on the LocalTalk: the subnet mask operation is still performed on the LocalTalk Internet addresses to determine the LocalTalk net number. This is done to allow the FastPaths to route DDP/IP traffic among themselves, so that only one unit need be designated as the fixed router.

The **FastPath** fixed routing code obeys the following rules:

1. The **FastPath** will route a packet from the LocalTalk to the Ethernet, if the destination IP network is the **same** as the **FastPath's Ethernet IP** network.

2. The **FastPath** will route a packet from Ethernet to the LocalTalk, if the destination IP network is the **same** as the **FastPath's LocalTalk/IP** network *and* if the destination IP subnet is the **same** as the **LocalTalk/IP subnet**.

The easiest way to explain the fixed routing approach is by example. The example below shows how to add a route to a LocalTalk/IP net served by the **FastPath**, using the **UNIX** environment as a model. If you are configuring a host environment other than **UNIX**, you will need to issue a command similar to the **route** command referred to below.

Example: Class C Ethernet network routing to a Class B LocalTalk.

Existing hosts on the network have addresses with a network of 205.128.10 (for example, Host1 has an IP address of 205.128.10.1, Host2 has an IP address of 205.128.10.2, etc). The administrator is adding a **FastPath** and decides that subnetting on byte 4 (i.e., the host portion) of the IP address is impractical. The administrator decides instead to place the new LocalTalk/IP network on an unused Class B network.

There are only two restrictions on choosing the Class B network and the **LocalTalk/IP subnet**: the Class B net number must not conflict with any Class B nets already in use at your site, and the **LocalTalk/IP subnet** must not duplicate any established AppleTalk network numbers.

As with subnet routing, there are three IP parameters used in configuring the **FastPath:**

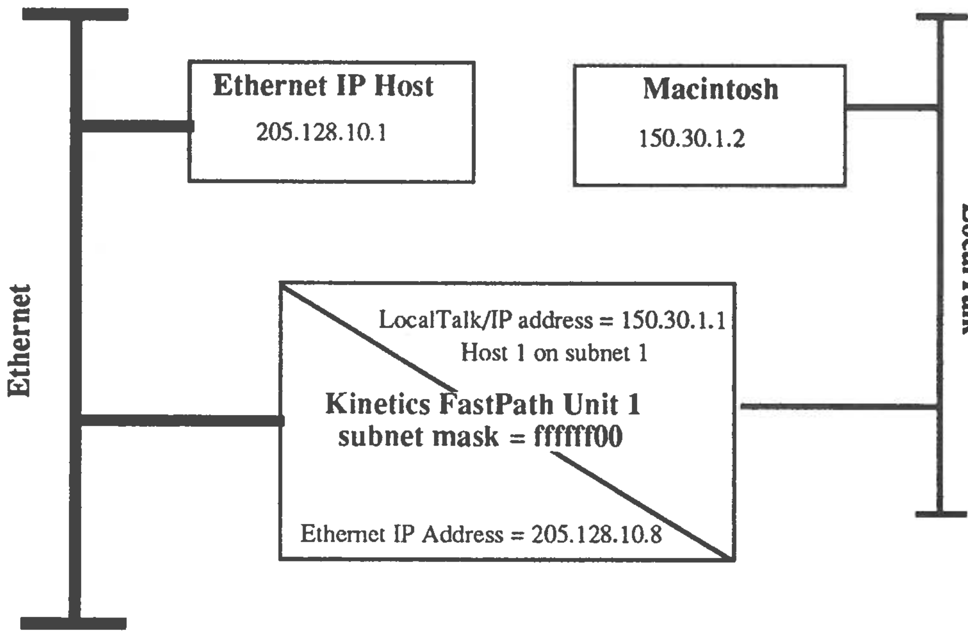
- the subnet mask,
- the **Ethernet IP** address of the **FastPath,**
- the **LocalTalk/IP** address of the **FastPath.**

The subnet parameter is used **only** to determine the LocalTalk network number, so we have flexibility in its choice. In this case, we can choose byte 3 of the LocalTalk/IP address as our subnet. The subnet mask will then be ffffff00 since we would like to disregard the lower 8 bits of the IP address (i.e., the lowest byte) and we are only examining eight bits as our subnet (i.e., the third byte).

Next we choose the IP address for the Ethernet side of the **FastPath**. As with the subnet approach, we are guided primarily by the addressing conventions on the existing IP network. The convention on our example network is to assign addresses of the form 205.128.10.*x* where *x* is some unused value. In our example, let us assume 205.128.10.8 is unused and let that be our **FastPath's Ethernet IP** address.

Last comes the assignment of the **FastPath's LocalTalk/IP** address. The administrator has chosen a Class B network of the form 150.30.*y.z*. Note that the AppleTalk net number must be *y* since this is the **LocalTalk/IP subnet**. In addition, neither *y* or *z* can be 0 or 255, since these addresses are reserved as broadcast addresses. In our example, we choose **150.30.1.1**.

The figure below summarizes our fixed routing configuration:



Final check: trace an IP packet generated on the LocalTalk destined for a host on the Ethernet network. For example, let the destination be address **205.128.10.1,** and the source be address **150.30.1.2**. The **FastPath** would compare the network portion of the destination address, **205.128.10**, to the network portion of the **FastPath's** Ethernet IP address. Since the two network addresses are the **same**, the **FastPath** will route the packet to the Ethernet where host 205.128.10.1 will receive it.

If the Ethernet host, 205.128.10.1 decides to reply to the LocalTalk host, 150.30.1.2, the FastPath will compare the destination IP network, 150.30, to the FastPath's LocalTalk/IP network address, which is also 150.30. Because they are the same, the FastPath will then check that the destination subnet, 1, is the same as the LocalTalk/IP subnet. Since the subnets agree, the FastPath will route the packet to the LocalTalk where host 150.30.1.2 will receive it. Our FastPath configuration data is correct.

Host Configuration for Fixed Routing

The LocalTalk/IP network administrator must explicitly add a route to the host routing table of each IP host on Ethernet that is to allow connections from the LocalTalk/IP network.

On the **UNIX** operating system, this is done with the **route** command. For our example, each host must add a route for our LocalTalk/IP Class B network, **150.30**, and specify that packets are sent via the **FastPath**, whose Ethernet IP address is **205.128.10.8**. The appropriate **route** command is:

```
route add 150.30. 205.128.10.8  3
```

The last argument (3) is an indication of the distance through the **FastPath** to the LocalTalk/IP network. Details on **route** command usage can be found in the *Unix Programmer's Manual.*

Internet Addressing and Berkeley Subnetting

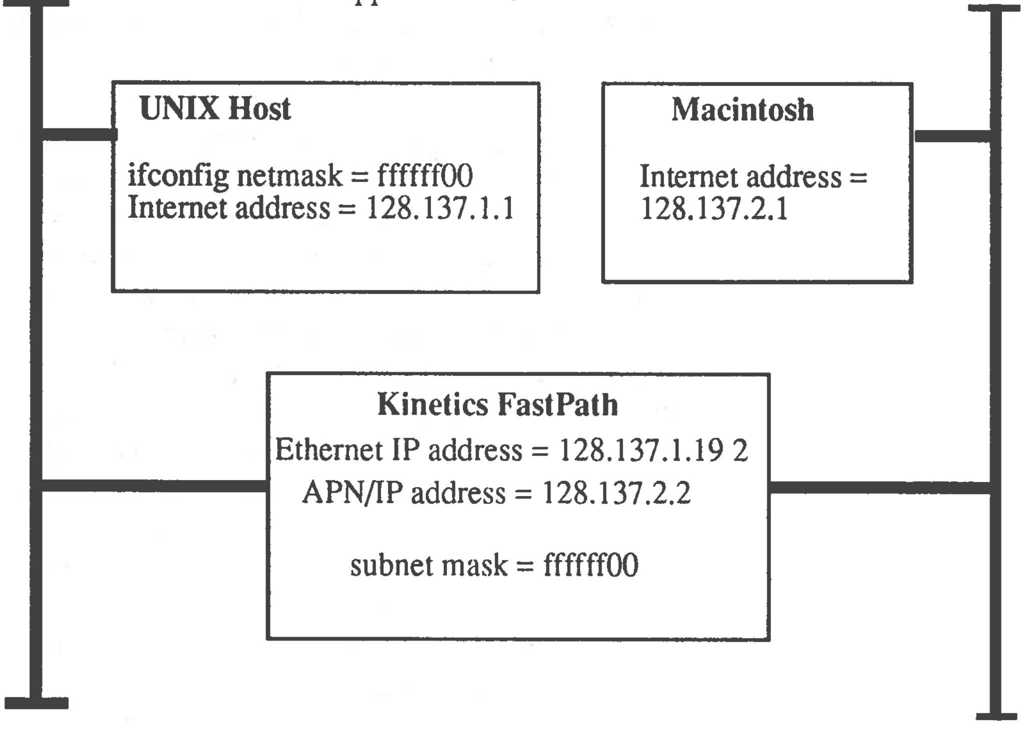
A subnet is defined to be a logical subsection of a single Internet network. The "Internet Standard Subsetting Procedure" (NIC RFC950) describes a method for implementation of subnets and is the basis for Berkeley's 4.3BSD UNIX subnetting scheme. Since the Kinetics Fastpath IP gateway program uses the same subnetting scheme, a method for intergrating an AppleTalk network subnet into a 4.3BSD subnet is necessary. This note describes two methods and an example of each.

Adding subnetting to a 4.3BSD Internetwork is via the **ifconfig** utility. The "netmask" option of ifconfig allows the system administrator to specify a subnet field within the internet address given to a particular network interface. The netmask is given as a hex number that specifies a mask to apply to outgoing internet addresses to determine if the address is on a known subnet. This mask includes both the major net and subnet fields of an internet address. For example, if the host's internet address is given as 89.10.128.1 and a subnet is to be based on the second byte of the network's address, the netmask would be given as ffff0000. The upper two bytes of ffff mask the major net number (89 in this case since we are a Class A network) and the subnet field (10 in this case). The host will route packets destined for those hosts that exactly match the subnet of the source host, that is:

((destination address) & netmask) = ((source address) & netmask))

or will route packets destined for nets that appear in its routing table.

The first method involves use of the 4.3BSD utility **route**. **Route** allows the system administrator to manually add an AppleTalk subnet to a host's routing tables. The administrator will have to explicitly add a routing table entry to allow the FastPath to route to a separate AppleTalk subnet. The idea is that each host acts as a subnet gateway for each AppleTalk net.



In the example, the UNIX System Administrator should run the following command:

route add net 128.137.2 128.137.1.192 3

This adds the AppleTalk subnet (128.137.2) to the host's routing tables and tells the routing mechanism that the FastPath is a gateway from the UNIX subnet (128.137.1) to the AppleTalk net.

The second method of resolution of subnet routing treats an AppleTalk network as residing on the same 4.3BSD subnet but on different FastPath subnets. If we use the same example as pictured in the previous method but change the UNIX host's **ifconfig** net-mask to fffff000, then the 4.3BSD subnet is based on the two bytes of major net and upper nibble of the third byte for the subnet number. No other change is required. From the UNIX host's point of view, the AppleTalk side of the FastPath is on the same subnet (128.137.0) so the host will route to the LocalTalk subnet. From the FastPath's point of view, the UNIX subnet (i.e.,1) is different than the LocalTalk subnet (i.e.,2) so the FastPath will route to the UNIX subnet.

Footnotes

[1]See "Inside AppleTalk", Apple Computer, Inc., July 1986.

[2]Mogul, J., Postel, J., "Internet Standard Subnettting Procedure", RFC-950, Network Information Center, SRI International, August 1985.

[3]In general, AppleTalk net numbers can range from 1 to 65,535.

[4]In general, the total number of subnets available is determined by the formula 2^m, where *m* is the subnet mask size. The number of subnet hosts available is 2^{8-m} for Class C nets, 2^{16-m} for Class B nets, and 2^{24-m} for Class A nets. However, Kinetics restricts the number of Class A subnets to 2^{16-m}. The reason for this is the subnet size cannot exceed the maximum AppleTalk net size.