Internet Draft

Ross Callon
DEC

October 1992

Addressing and End Point Identification, For Use with TUBA

Status of the Memo

This document is an Internet Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are working documents valid for a maximum of six months. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a ''working draft'' or ''work in progress''.

Distribution of this memo is unlimited.

Abstract

Addressing is critically tied into routing and scaling in very large Internets. This draft paper discusses how NSAP addresses can be used to allow scaling in a huge Internet, and to allow the flexibility necessary to deal with multiple different dimensions of Internet growth.

This document is a personal contribution of the author as an input to the IETF TUBA working group.

1 Summary

The Internet is approaching a situation in which the current IP address space is no longer adequate for global addressing and routing. There is an urgent need to develop and deploy an approach to addressing and routing which solves these problems and allows scaling to several orders of magnitude larger than the existing Internet [1]. A companion paper [2] describes a simple proposal which provides a long-term solution to Internet addressing, routing, and scaling based on gradual migration from the current Internet Suite (which is based on Internet applications, running over TCP or UDP, running over IP) to an updated suite (based on the same Internet applications, running over TCP or UDP, running over CLNP [2]). This approach is known as "TUBA" (TCP & UDP with Bigger Addresses).

This paper describes a proposal for how addressing and end point identification may be done with TUBA. This allows the ability to scale, as well as the flexibility to deal with multiple different dimensions of Internet growth. For example, the Internet may grow by creation of more backbones and regional networks along the current Internet model, by use of IP and/or CLNP service by large public carriers, and/or by provision of Internet services to a very large number of homes and/or small businesses over telephone networks or similar services.

The main motivation of TUBA is to allow the Internet to scale to a size many orders of magnitude larger than the current Internet. In fact, it is important to be able to scale to as large an Internet as we can conceive may exist (for example, there may someday be a network and hundreds of hosts in every home in the world -- any proposed solution to Internet routing, addressing, and scaling should be capable of easily scaling to this size). The addressing proposal described in this paper makes use of the general scaling concepts described in NSAP Guidelines [reference], with flexibility for other address techniques also built in.

Annex B provides a very rough description of how this proposal allows scaling beyond the largest anticipated size of the worldwide Internet.

2 Overview of OSI NSAPs

<this section needs to be filled out>

- very flexible (some would say too flexible); binary string of variable length up to 20 byte maximum length;

- NSAP is split into "Part which conforms to international standards" (IDP) and "rest" (DSP)

- AFIs; first byte is AFI, indicates format for the part which conforms to international standards. A variety of AFIs are defined (more could easily be defined if anyone could think of a use for a new format).

OSI standards allows for binary and decimal addresses. However, the decimal encodings are obsolete/useless and can be ignored. In practice, only binary representations make sense (and associated external representations, such as writing addresses in hexadecimal with or without separators for human readibility). Old version of standard places variable length restrictions based on AFI. However, this is based on obsolete requirement that addresses be capable of being represented in 40 decimal digits. Given that the decimal representation is obsolete, these length restrictions can be ignored, and have been removed from current versions of the standard. Thus only length restriction is that address has maximum size of 20 bytes (regardless of AFI).

It is important to clarify what hosts can assume about addresses, what routers can assume, and how addresses are actually to be used. (see section xxx below). Addresses will need to have a lot of structure, but most of the structure will be used to allow address summarization (along boundaries which need to be configured) and administration. Thus most of structure is not visible to hosts and router implementations.

There are multiple NSAP formats in use. However, routers need to think of NSAP addresses as a string of bits (or a string of nibbles -- half bytes), where forwarding is based on prefixes. Thus, routers don't know anything about any of the various formats, except perhaps for the configuration code, and knowledge of the size of the ID. Similarly, I would expect aggregation to require substantial manual configuration, such as manually configured summary addresses (surely a router is not going to *automatically* determine that it should aggregate, based on knowledge of the gosips). Thus, adding another format does not impact the router, except perhaps by causing more entries to get added to (or deleted from) the forwarding table.

3 Technical Considerations / Constraints

3.1 End Point Identification

An address performs two functions: It ***identifies*** the system, and it specifies the ***location*** where the system is. The identification of source and destination systems may, for example, be used to demultiplex various network communications. The location of a system may be used as one input to the routing function (to determine how to get a packet delivered to the system).

There are some situations in which it is preferable to perform these two functions independently: For example, if a system moves, then the identification of the system may stay the same, while the location of the system may change. Similarly, if the location of the system is specified hierarchically based on network topology (or based on the geographic location of a private network's attachment to a public service), then a change in network topology (or a change in where the public connection is made) may result in a change in the specification of the location of the system, even though the identity remains constant.

Traditionally (for example, in the 32-bit IP address space) the functions of identification and location are intermingled, so that it is difficult or impossible to change one without changing the other. The current Internet protocol suite generally does not take advantage of the possibility of separating these two functions (ie, enhanced features of the Internet suite, such as mobile host support, had to be developed without consideration of the possibility of separating location and identification information). For this reason it is difficult to accurately predict precisely how valueable this separation will turn out to be. However, specification of the location and the identity of a host system are architecturally separate functions, and therefore it is felt that separation of these functions will turn out to be valuable.

3.2 NSAP Address Standard and Address Structure

TUBA makes use of NSAP addresses which correspond to ISO standards. This includes the requirements of the NSAP address standard [reference], and routing protocols [reference ES-IS and IS-IS]. However, in order to allow the host identification to be separated from the location, the requirement (from IS-IS) that the ID be unique within the area is strengthened, to require that the ID must be globally unique. In order to emphasize that the ID globally identifies the end-point (ie, the conceptual virtual host which is the recipient of a CLNP packet), the globally unique ID will be referred to as the End-point IDentifier (EID).

Also note that (in accordance with IS-IS) the last byte of the NSAP address is refered to as the Selector (SEL) and is used to demultiplex users of the CLNP service within a host. This therefore provides the same function as the Protocol field in the IP header. For TUBA, we will use values corresponding to the assigned Protocol values from assigned numbers for the Selector. This results in an address structure as follows:

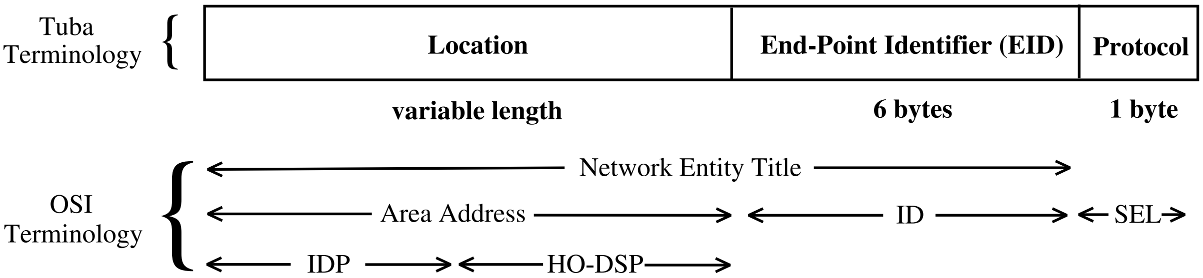

Figure 1 - Basic Structure of TUBA Addresses

The basic structure of TUBA addresses, and the correspondance between TUBA addresses and OSI addresses, is illustrated in figure 1. TUBA splits the addresses into Location, EIDs, and Protocol. Here the location identifies where a system is. The EID identifies the system. Finally the Protocol specifies what user protocol is operating above CLNP (i.e., specifies the protocol whose packets are included in the CLNP Data field).

The OSI term "Network Entity Title" (NET) refers to the entire address, except for the Protocol field. The NET therefore performs the same function as a 32-bit IP address (except longer with more flexibility). The NET is therefore the entity which is returned by DNS name-to-address lookups (in much the same manner than IP DNS lookups return a 32-bit IP address).

IS-IS uses the low order 7 bytes of the address as the identifier and selector. The rest of the address (all of the address except for the last 7 bytes) is known as the "area address", and specifies which area the system is in. Once the destination area is reached, IS-IS then routes directly to the destination host. The IS-IS term "area address" is therefore analoguous to the TUBA term "location", and specifies where the system is.

[NOTE: TUBA requires use of CLNP and ES-IS. Other CLNP-related network layer protocols such as IS-IS are assumed to be used, but are not actually required. If, hypothetically, we were to define a new intra-domain protocol then the TUBA format will remain constant in the sense that the low-order 7 bytes will specify EID and Protocol, and the remaining high-order part of the address would specify the location. However, in this hypothetical case the location might not necessarily correspond to area (for example, if a new protocol routed to subnets, then the "location" field from the TUBA addresses may correspond to "subnet address" from a new routing protocol specification). This allows a graceful transition from IS-IS to a future intra-domain routing protocol, in the hypothetical case that this was required at some point in the future].

NSAP addresses used with TUBA must be valid OSI NSAP addresses. This implies that the first byte of the Location must be an AFI (authority and format identifier), which specifies the format of the remainder of the IDP (ie, for the high-order part of the location). Depending upon the value of the AFI, the location part of the address may vary from a minimum of one byte (containing only an AFI) to a maximum of 13 bytes. Thus the overall TUBA address may vary from a minimum of 8 bytes (which would contain only an AFI, EID, and Protocol field), to a maximum of 20 bytes.

The TCP connection is identified by the EID and Protocol (as well as information carried in the transport level header). Thus the location part of the address is not required for unique identification of the host. However, a complete valid address is required in all packets. Thus, packet forwarding is based on the entire NSAP address, and in general is not based on solely the location nor on solely the EID. [Note: For normal operation with IS-IS and IDRP, the packet is forwarded based on location until the packet arrives at the destination area, and then forwarded based on EID to the destination host].

Future documentation will specify how a host (if it knows a priori the EID portion of its address) can autoconfigure the location part of the address. The same mechanism is also useful for updating the addresses assigned to a particular area or routing domain, by allowing auto-re-configuration of the location part of the address. This is important to allow addresses to be changed when necessary (in the current Internet architecture changing addresses is just too hard). Note, these mechanisms will be based on the address autoconfiguration work done for CLNP, with additional TUBA-specific features (for example, to deal with mobile hosts, to allow TCP connections to remain in operation undisturbed through address changes, etc), The TUBA-specific mechanisms are straightforward, but have not yet been written down in detail due to time constraints.

It is necessary to be able to map from EID to name. This is analoguous to the mapping from 32-bit IP address to name in the current Internet architecture. EIDs therefore must be administered in a manner which facilitates this mapping. In addition, EID assignments must be global, and must conform to Internet requirements (TBD). Initial proposal is described in Annex A.

3.3 Addressing and Scaling

The primary motivation for TUBA is to allow scaling of the Internet architecture to a truely enormous worldwide ubiquitous Internet. This requires that addresses used with TUBA be assigned in a manner which facilitates scaling.

<editor’s note, we need a term for "non-transit network which operates as a customer of the public service providers". For now I will just call these "customer networks".>

The current IP addressing scheme assigns one or more "network numbers" to each customer. There are two immediate problems with this approach:

a) The Internet is running out of some types of addresses (particularly, class B network numbers).

b) Internet routing tables (and routing protocols) require a separate entry for every customer (ie, for each network number). As the internet grows, this implies that the growth in routing tables and routing protocol information grows at least linearly with the number of customers.

The first of these problems is largely a matter of ensuring that the address space (including any hierarchical subdivision of the address space) is large enough so that you don’t run out of addresses to assign. This implies that one needs to be very careful when subdividing and address space, but does not appear to be a significant problem for the NSAP address space due to the large number of addresses available.

The second of these problems require much more care. There are several proposals for how to deal with growth of addressing information:

a) Massive Resources

This approach attempts to deal directly with one or more separate routes for each Internet customer. In the future this potentially will require massive routing tables, massive CPU and memory in routers, and some (unspecified) management tools to make management of the associated information feasible.

b) Provider Based Addresses

This approach makes use of addresses for customer networks which are based on prefixes assigned by provider. Thus, any one particular public service provider would obtain a large block of the address space based on a single prefix from a national or international address authority. Each provider would then allocate a part of this address space (based on a longer prefix) to each customer. This is the approach recommended by RFC 1347.

c) Provider-Subnet Addresses

This approach is not intended for general use by all networks, but rather is intended to deal with one important special case. In particular, in some situations it is necessary to deal with very large numbers of individual systems or small sites connected via telephone networks, public data networks, ISDN, or other public service. For example, such situations may occur in retail sales and home applications. In general, telephone networks, public data networks, and similar networks have their own globally significant address space. In these cases, it will be necessary to map from the global internet address space to the subnet address space used by the provider. This is facilitated with TUBA by allowing such "provider-subnet" addresses to be embedded as part of the "location" field in the TUBA NSAP address.

Provider-subnet addresses are considered important enough that they are discussed separately in section 3.4 below.

d) Geographic Addresses

This approach assigns addresses to customer networks based on the geographic area of their attachment to a public service provider. This approach requires considerable cooperation between public service providers. In particular, it requires that a metro-area be fully connected, (i.e., either once a packet is delivered to any point in a metro area, it can reach any other point in the same metro without leaving the metro, or special mechanisms such as encapsulation are used to allow the packet to be delivered between different providers in the same metro area via intermediate networks).

The topology constraint associated with geographic addressing is just a specific case of the general requirement/assumption of any hierarchical routing scheme: that each region, sub-region, sub-sub-region, etc. be fully connected. There is an analogous case with provider-based addressing: A provider's facilities must be fully connected if they are to be identified by a single prefix.

(Note, this is not an absolute requirement -- you can use tricks like tunneling to link together the pieces of a partitioned region. IS-IS can do that to heal partitioned Level 1 Areas, and the various mobile IP schemes do something similar to allow members of a subnet to roam to arbitrary places in the graph.)

Provider-based addressing and geographic addressing are not necessarily mutually exclusive. Suppose that some providers rely on provider-based addresses (implying that customers of this provider are required to take addresses from that provider's space), and other providers make use of geographic addressing. In this case, those providers which use geographic addressing are required to cooperate and exchange traffic between them, but will have the offsetting advantage that they can advertise "open" addressing (ie, addressing which does not lock the customer into a single provider).

TUBA does not require any particular solution to the hierarchical routing issue. Any combination of Provider based addressing, Geographic addressing, and Provider-Subnet addressing is permitted with TUBA.

<Editor's note: I would like to include a more complete discussion of hierarchical routing possibilities, but do not have time yet>

3.4 Provider-Subnet Addressing to Homes and Small Sites

Imagine a situation in which the financial services department of XYZ corporation has placed point-of-sale terminals in 100,000 small retail stores (we might presume that the stores are independently-owned, and that XYZ corporation has contracted to provide a service to the stores). Alternatively, imagine a situation in which XYZ entertainment corporation has placed entertainment devices in 100,000 homes.

In either cases, let's suppose that access from the central offices of XYZ corporation to the 100,000 individual sites is done over the public telephone network, and that the applications running from XYZ to the sites is done using Internet applications running over TUBA. In this case, it will be necessary to assign NSAP addresses to each of the individual sites, and to facilitate routing between XYZ corporation and the individual sites.

With the existing IP architecture, this problem is very hard to solve. It becomes necessary to map from the 32-bit IP address space to the "subnet point of attachment (SNPA) addresses" (ie, phone numbers) used in each customer site. With IP, this requires a very large mapping table, which is either difficult or impossible to manage.

With TUBA, this problem is solved by embedding the SNPA address in the location part of the NSAP address. This is facilitated by defining a separate AFI for each commonly used type of SNPA address. For example, in the telephone case the NSAP/TUBA address would be as follows:

<AFI><telephone number><EID><Protocol>

In this case the AFI is 1 byte long, the telephone number is variable length (padded to a constant length of xx bytes, corresponding to the maximum length of worldwide telephone numbers), the EID is 6 bytes long, and the protocol is 1 byte long.

Note that with this approach, rather than requiring a mapping table with 100,000 entries, only a single entry is needed in routing tables. This entry would route packets destined to the appropriate type of address to one or more routers with interfaces on the public telephone network. The routers attached to the telephone network would then be able to obtain the correct subnet address (ie, telephone number) by extracting it from the NSAP address. If a finer grained control was required (for example, routing traffic separately based on country, or based on area code), then this would be easy to do by using prefixes applied to the telephone number. This is facilitated since the telephone number space is globally meaningful, and is assigned in a manner which corresponds to the global topology of the telephone network.

A similar approach can be used for public networks using other common address formats. AFI values have been assigned for telex numbers, telephone numbers, X.121 addresses, and E.164 addresses (used by ISDN and some other public networks).

3.5 Compatibility with Current CLNP deployment

TUBA is based on the use of CLNP as a scalable network layer for use with existing Internet applications. However, it must be realized that CLNP will also be used for other purposes. For example, some OSI applications have been deployed which make use of the services provided by CLNP. Similarly, some proprietary applications have also been deployed which make use of CLNP.

It is therefore desireable (although not absolutely necessary) to allow a common address format be used for TUBA, and for other uses of CLNP. The use of a common address format will simplify network configuration, management, and operation.

The identifiers used with TUBA must be compatible with the identifiers used with other CLNP applications, in the sense that the same ID cannot be assigned for one host for use with TUBA and to another host in the same area for use with OSI applications.

Use of common identifiers is also somewhat useful but is not as important. For example, if TUBA makes use of one format for identifiers, and OSI applications make use of a different format, then multi-protocol hosts will have to have two different identifiers assigned to them. However, the requirement that the identifiers used with TUBA must be capable of being used as the index for a DNS identifier-to-name lookup constrains the form of identifier used with TUBA.

Generally, current CLNP applications (including OSI and proprietary applications) make use of two common methods for assigning IDs: Some systems use IEEE 48-bit globally administered unicast IDs, Some use manual configuration of IDs. The latter are not a problem (locally administered IDs for use with OSI applications can be chosen for compatibility with TUBA). However, given that some systems are already using IEEE IDs, we have two choices: (i) Use globally administered unicast IEEE IDs; or (ii) Avoid the 25% of the 48 bit ID space which happens to correspond to the IEEE globally administered unicast IDs.

Given our desire to make ID to name lookups easy, we propose to use the latter approach: All IDs assigned for use with TUBA will use a prefix which avoids collision with the IEEE globally administered space.

The current practice in assignment of NSAP addresses is somewhat more complex (ie, different formats are being used in different situations, and some customer networks are uncertain as to which NSAP format would be the best to use). However there is a strong trend towards use of NSAP addresses which are chosen to allow scaling. In particular, the current trend is towards use of provider-based addresses assigned more-or-less in conformance with RFC 1237. Also, existing NSAP allocations make use of a variety of address lengths, but use a consistent 6 byte ID. The current practice is therefore compatible with TUBA addressing requirements specified in this document.

<Editors note: I would like to discuss use of the Selector: include protocol field as last byte of NSAP>

4 Proposed Address Solution

4.1 What Hosts can Assume about Addresses

- The NSAP address is split into Location (variable length), EID (6 bytes), Protocol (1 byte)

- Location should be treated as flat variable length binary string (in fact will have structure, but structure is not visible to host)

- EID is six byte globally unique identifier. TUBA host can assume that this will always be globally unique, and will identify the other system. DNS will be able to look up name based on EID. However, host cannot make any assumptions about the internal contents of the ID.

- A single host can have multiple values for the location part of the address. It is possible that some future specifications might allow what constitutes a correct location part of the address for any one host to vary over time.

- Logically a host only has one correct value for the EID. Thus, if a real physical host has multiple EIDs assigned to it, it is treated as if it were multiple logical hosts. If the EID changes, then logically you have a different host.

- Protocol field uses same values as IP. Host uses value in destination address. The value in source address MUST be ignored.

4.2 What Routers can Assume about TUBA Addresses

- The NSAP address is split into Location (variable length), EID (6 bytes), Protocol (1 byte)

- Routing is in accordance with other standards

  - For initial use, using existing CLNP-associated routing protocols such as ES-IS, IS-IS and IDRP

  - These use prefixes on location part, on "nibble" boundaries, plus ID in destination area

  - For forwarding loop: Router SHOULD NOT be aware of internal structure of location part of address, except for matching prefixes to location part of address

- Note: During transition period EID will contain IP address. This could potentially be used for router-visible transition methods such as packet translation details outside of the scope of this document.

- Router ignores Protocol (as required by IS-IS and IDRP)

- <need to add summary of what IS-IS assumes about addressing>

4.3 Configuration of NSAP Addresses

Hosts, routers, name servers, and other TUBA systems MUST allow configuration of NSAPs as a simple hexadecimal string without consideration of internal structure.

Systems MAY also allow configuration in other ways (e.g., systems may allow the option of checking to see if a valid AFI is provided, and/or allow the IDP to be entered in a format which is AFI dependent). However, in order to conform with the TUBA proposal it MUST be possible to override this and enter NSAPs as simple hexadecimal strings.

NSAPs may be input as simple hexadecimal strings. Different sub-fields in the NSAP may be separated by dots, but the dots are optionally included only for ease of writing down the NSAP, and do not have semantic meaning (i.e., the dots are ignored by address parsers). The user can put them wherever he/she wants in a configuration file (and they can be inserted in any position on output). For example the following NSAP Addresses are all equivalent:

47000580FFEC00000000000010000080F1005100

47.0005.80FF.EC00.0000.0000.0010.0000.80F1.0051.00

47.0005.80.FFEC00.0000.0000.0010.000080F10051.00

The latter grouping uses the location of dots to group bytes according to administrative fields. In either case, the placement of the dots has no significance other than readability.

4.4 What Address Solution Will be Used

<I need to work on this more>

Currently multiple things in use, will converge over time

Depends upon situation, can use combination of provider based, geographic, or provider-subnet address based.

For geographic base, use different DFI, split rest differently

For subnet-provider-address-based, use X.121 (or...) format

5 References

<Editors note: This list of references needs to be filled in with more complete references>

[1] Overview of ROAD problem (ROAD report or IESG report)

[2] RFC 1347, "TCU and UDP with Bigger Addresses (TUBA), a Proposal for...

[3] RFC 1348, and/or its replacement

[4] NSAP Guidelines

[5] Dave Piscitello's TUBA CLNP Profile

[6] "Protocol for Providing the Connectionless-Mode Network Service", ISO 8473, 1988.

[7] "Supernetting: An Address Assignment and Aggregation Strategy", V.Fuller, T.Li, J.Yu, and K.Varadhan, March 1992.

[8] IP Address Guidelines paper

[9] Steve Deering's Geographic paper

[10] "Extending the IP Internet Through Address Reuse", Paul Tsuchiya, December 1991.

6 Security Considerations

Security issues are not discussed in this memo.

7 Author's Address

Ross Callon
Digital Equipment Corporation
550 King Street, LKG 1-2/A19
Littleton, MA 01460-1289

Phone: 508-486-5009

Email: Callon@bigfut.lkg.dec.com

Annex A A Draft Proposal for EID Assignments

A.1 Constraints

<this will be summarized from the earlier discussion in section 4>

A.2 Proposal

For initial use:

- Fixed 16-bit prefix prepended to IP address
- Prefix chosen to not collide with IDs selected from IEEE globally administered space

For future use:

- Administered by IANA
- Details for further study

Annex B A Rough Analysis of NSAP Scaling

We know that the Internet ***will*** grow a lot, but we don't know ***how*** the Internet will grow. In particular, we don't know what the topology will look like. For example, if the Internet reaches

every home in North America and Europe, how will this public Internet service be provided? Will there be one large public service provider per country, or many smaller providers? If there are many public service providers in each country, how will the providers interconnect? How many providers will there be worldwide and how large will they be? Will the bulk of systems connected to the Internet use "public service provider specific" addresses such as X.121, E.164, or telephone numbers?

This uncertainty about the manner in which the Internet topology will grow leads to a resulting difficulty in determining what future Internet addressing should look like, and in accurately predicting how any particular addressing plan will scale. Fortunately, the NSAP address scheme used with TUBA provides a great deal of flexibility in how addresses are structured. Also, as discussed in section 4, TUBA-capable hosts are required to make ***no*** assumption about the substructure of the location field of the NSAP addresses, and routers similarly should make only very limited assumptions about the location field. This implies that the substructure can be changed (or different structures for the location field used in different parts of the Internet) without effecting existing equipment.

Given this flexibility in NSAP address structure and uncertainly in network topology design, it is difficult to accurately predict precisely how addresses will scale. However, we can give rough "back of the envelope" calculations for several different scenarios.

B.1 Scaling of Provider-Based Addressing

It is proposed that provider based addresses basically look like:

<country*><provider><substructure**><customer><area><EID>

* NOTE: By "country", we really mean country or geographic area. For example, multi-country continental networks (such as a European wide backbone) would specify continent in this field, rather than country.

** NOTE: The "substructure" field is needed because eventually we will either end up with too many providers in order to route on a flat provider space (in one country), or, more likely, end up with too many customers of a single provider in a single country. For example, given more than 10,000,000 companies in the USA, and roughly 100,000,000 homes in the USA, it is likely that eventually there will be one or more providers in the USA which have enough customers to require hierarchical routing between customers of the same provider. This "substructure" field is referred to as "reserved" in the US GOSIP and ANSI address spaces, since initially (so long as the number of providers within a country, and the number of customers of any one provider is small), the "substructure" field does not need to be used.

Note that the some of these subfields in the Provider-based addressing will actually be further subdivided into several sub-sub-fields. For example, the country will be specified using the combination of an AFI (authority and format identifier -- the first byte of the NSAP giving the format of the next field) and DCC (data country code) or ICD (international code designator). Some

NSAP format provide a single byte after the country to indicate the type of the rest of the address (for example, this byte will specify whether the address is provider-based or geographic-based).

For example, with US GOSIP based addressing, the approximate mapping is as follows:

My Term (for rough analysis)	GOSIP term	Field size
Country	AFI and ICD	3 bytes
(not mentioned)	DFI	1 byte
Provider	AA	3 bytes
substructure	reserved	2 bytes
customer	RD	2 bytes
area	area	2 bytes
EID	ID	6 bytes
(not mentioned)	Protocol/SEL	1 byte

Note that the DFI (DSP Format Identifier) in the GOSIP space can be used to identify different address formats, such as the provider based format (as illustrated above) versus a geographic format. The selector/Protocol field is provided by the GOSIP format (and required for TUBA addressing), but is not considered for purposed of analysis of scaling because the Protocol/Selector field is only used for demultiplexing within a host, and is therefore not useful for scaling to a large number of hosts.

Initially, we can probably treat the country and provider as a flat field which specifies the provider. Assuming that dynamic routing in a flat address space allows for roughly up to 10,000 entries at one level (noting that the phone system seems to manage with this size, and that the Internet also is managing routing in a flat space with several thousand network numbers). Our guess is that this may be sufficient indefinitely (there may never be more than 10,000 providers). If this guess is wrong and the number of providers gets large enough that it is not possible to route amongst providers on a flat basis, then we will need to route hierarchically amongst providers.

However, provider identifiers are in fact being handed out geographically, based on country (or at least continent, in the case of multinational nets). Thus, if we needed to, we could first route on country, then on provider, without changing the current address assignments. In practice, if the number of public service providers worldwide gets to be substantially larger than 10,000, then most likely there will be some "major" providers, which will continue to be advertised globally in routing advertisement and tables, as well as some "minor" providers, which can be routed on a per-country basis (i.e., routes to the minor providers will only be maintained within their country of operation).

The RD field allows up to about 10,000 customers per provider (again, routed on a flat basis; in fact, more than 10,000 entries are possible, but we are assuming that 10,000 is a reasonable "order of magnitude" estimate of the comfortable maximum size of a routing table).

Let's suppose that we end up with something like 100 providers in the US (it might be larger, but this number is a sort of "today's guess"). Given that there are millions of companies in the US (most very small), it would appear inevitable that if we assign value for "customer" to each company (even small ones), then we are ***eventually*** going to get more than 10,000 customers for a single provider. Thus there is the substruture field sitting there waiting in just the right spot to allow a single provider to hierarchically subdivide the addresses under it.

The Area, and EID fields are used for routing within a company / campus. For a large company we might have a few hundred areas. If we needed to, we could have a few thousand (the routing algorithms could handle this, but it is dubious that there will be many companies large enough to need it). Each area could have up to a few thousand end systems.

A rough upper bound of the size that can be handled by this scheme can be obtained by multipling together the maximum size at each level. Without using the substructure (reserved) field, and assuming that we would prefer to use flat routing of providers worldwide, we get 10,000 providers (worldwide) with 10,000 customers each with 100 areas per customer with 1000 hosts per area. This implies about 10**8 customers with about 10**13 hosts. If we ignore homes, then this is about the anticipated maximum size of the Internet. However, note that we will probably never actually be able to reach the "upper bound" size, since it is nearly impossible to completely fill in every level of a hierarchical address without having some parts of the address space become exhausted (ie, we will probably end up having to use the reserved field).

If some companies have more than 100 areas that is fine (we could easily go much larger). If we end up with more than 10,000 providers then again this is fine, since there is the option of routing by country/continent first, or of subdividing the providers within a country by using the "sustructure" field. If we end up with fewer providers and more customers per provider, then we will need to use the reserved field to allow hierarchical subdivision of the customers of a provider, this would potentially allow several million customers of each provider, each customer able to have well over 100,000 hosts (again, each customer could have up to several thousand areas and several thousand hosts per area).

We can similarly put together an upper bound with use of the reserved field. We might assume that we are unlikely to get more than 10,000 major providers worldwide. Thus, although the NSAP address scheme allows for more than this number, it is not likely to actually be used (except perhaps for a number of smaller providers). If we assume 10,000 providers worldwide, with 10,000,000 customers (router hierarchically within each provider using the "substructure" field, and noteing that the NSAP structure could actually handle more than this -- we just don't expect any one provider to require more than this), then we come up with a rough upper bound of 10**11 customers, each with up to several thousand areas, and several thousand hosts per area. Note however that this bound is much larger than the number of potential sites which exist which could want to attach to the Internet, unless we consider individual homes.

B.2 Provider/Geographic Based Addresses to Individual Homes

Routing to individual homes is also well handled. Given that each home is located in a small geographic location, each provider would probably arrange its provider network geographically, use the RD field (or perhaps the reserved field) geographically, and then assign a single area to each house in a geographic area (assuming that we have no more than a few thousand hosts within a single house). This would allow 10,000 geographic areas per provider (using the RD field) and 10,000 houses per geographic area (using the area field), or a maximum of about 100,000,000 houses per provider. If there is a single provider serving a country which has more than 100,000 houses (perhaps the Chinese monopoly PTT), then they will to use the reserved field and arrange the Chinese PTT itself in a hierarchical way, allowing 10,000 top-level things (using the Reverved field), 10,000 next level things (using the RD field), and 10,000 houses per thing (using area field). Thus the Chinese PTT, if it really needed to, could hand out 10**12 addresses to 10**12 houses within China, with each house having a thousand computers in it.

Note, in this example, it is possible that the Chinese PTT might not choose to route CLNP directly, but rather might offer simple telephone service or ISDN service to each home. In this case, provider-subnet addresses would be used, as was discussed earlier in section 3.4.

B.3 Scaling of Geographic Addressing

<this section is for further study>

B.4 Summary

It is hard to anticipate exactly how much fan-out will exist at each level of the hierarchy because we still don't know exactly what providers will exist and how many customers each will have. However, the NSAP address space with the ANSI/GOSIP address structure allows plenty of room and flexibility and scales well beyond the current size of the world.