IPv6 Deployment on the 6bone

Bob Fink
Becca Nitzan

ESnet – Energy Sciences Network

Lawrence Berkeley National Laboratory

LISA'97

31 October 1997

What is IPv6?

- The next generation Internet Protocol project at the IETF, called IPng, led to the development of a new set of standards to replace IPv4...the result is IPv6
- Major goals/features for IPv6 are:
  - Expanded addressing capabilities
  - Autoconfiguration (real plug and play)
  - Easy/graceful deployment/transition from IPv4
  - Support for security
  - Streamlined header format for hi-performance
  - Header support for options/extensions
  - Better quality of service features

IP version numbers (if you care :-)

- 0 – 3 unassigned
- 4 IP as we know it today - IPv4
- 5 Stream Protocol - ST - (never an IPng)
- 6 SIP -> SIPP –> IPv6
- 7 IPv7 –> TP/ IX –> CATNIP (died)
- 8 Pip (later joined with SIP)
- 9 TUBA (died)
- 10 – 15 unassigned

What is the 6bone?

An international IPv6 testbed network to:

- **Test IPv6 implementations and standards**
- **Tryout IPv6 transition strategies**
- **Get early applications/operations experience**
- **Motivate implementors, users and ISPs to try IPv6**
- **Start the transition**

Why do we call it the 6bone?

- The Multicast Backbone (called the MBONE) has provided a way to transport IP Multicast packets over the Internet, using tunneling, prior to production routers supporting native IP Multicast
- Like the MBONE, until IPv6 is implemented in site and ISP routers, there is no way to move IPv6 packets
- Thus an international tunneling environment to move IPv6 packets over the existing IPv4 Internet infrastructure is needed
- ...and of course Steve Deering invented ip multicast <u>and</u> had lots to do with ipv6...

...thus a 6bone!

Is this the only testing to be done?

- Besides the 6bone two other arenas have yielded excellent interoperability testing results:
  - trade shows like Interop, etc.
  - Univ. of New Hampshire compliance and interoperability lab
- The turn around time on some design changes, like the AGGR address format change described here later, were done and tested in UNH in record time:

  early April IETF (Memphis) for concept design
  mid-May for first I-D
  June/early July for implementation
  late July for UNH testing (and not just AGGR)
  15 companies came to test, and all pretty much worked

Standards of current interest

- Addressing (Aggregatable Global Unicast)
- DNS changes to facilitate above and site renumbering
- Router Renumbering (changing routing prefixes)
- Routing (Integrated BGP4)
- APIs
- Applications (what needs to be fixed for IPv6)
- Security (both authentication and encryption)
- all the usual that make up an implementation that we need to keep working on, and keep working :-)

What does it take to make a 6bone

- Dual IPv4 and IPv6 “stacks” on your “client” host
- DNS that supports IPv6 “AAAA” records
- IPv6 encapsulated (tunneled) in IPv4 packets
- Dual stack backbone routers that know IPv6 routing
- A registry of sites to helps us know who is who
- A mail list, various tools and statistics, available on the web at:

http://www.6bone.net

Some statistics

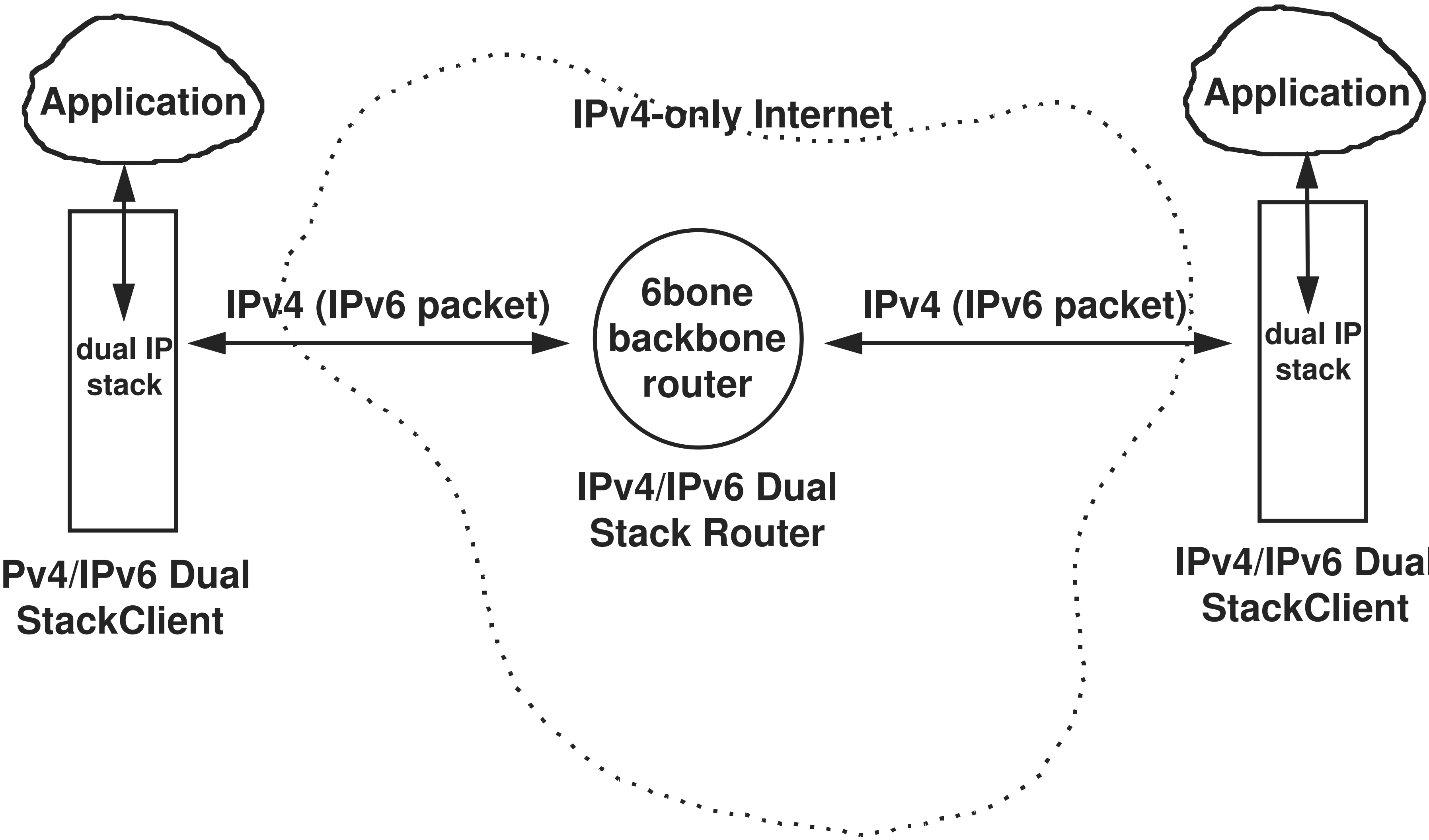
- The 6bone came online in mid-July 1996 with links from Denmark (UNI-C) to France (G6) to Japan (WIDE)
- There are now ~180 sites in 29 countries participating
- There are 20 sites acting as 6bone backbone routers
- The backbone uses mostly Integrated BGP4+, some RIPng, a little IDRPv6 and a few static routes
- There is a 6bone Registry with every participating site registered with a formalized “RIPE” syntax

29 countries on the 6bone

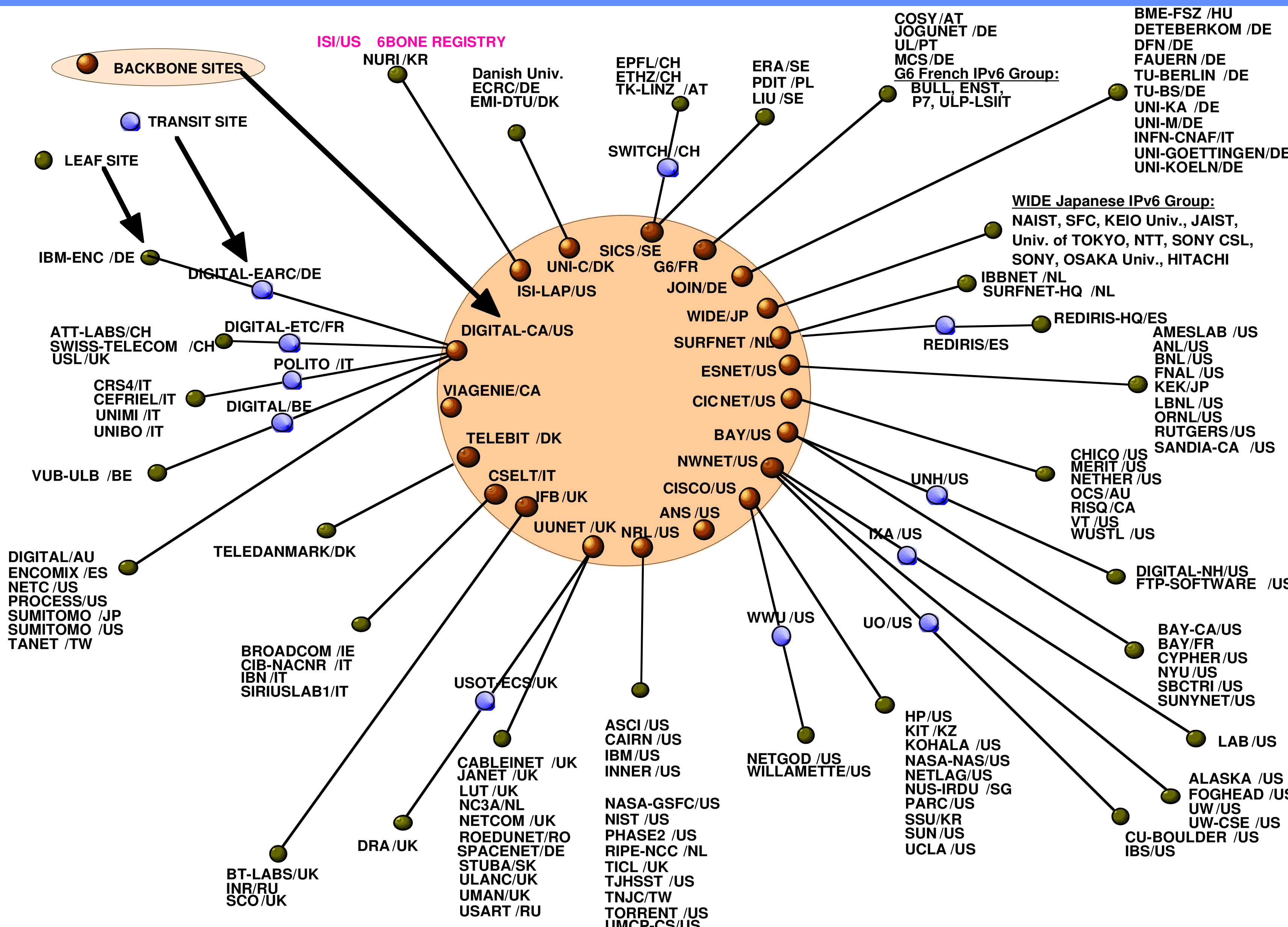
AT - Austria
AU - Australia
BE - Belgium
BG - Bulgaria
CA - Canada
CH - Switzerland
CN - China
DE - Germany
DK - Denmark
ES - Spain
FI - Finland
FR - France
HU - Hungary
IE - Ireland

IT - Italy
JP - Japan
KR - Korea
KZ - Kazakhstan
NL - Netherlands
PL - Poland
PT - Portugal
RO - Romania
RU - Russian Federation
SE - Sweden
SG - Singapore
SK - Slovakia
TW - Taiwan
UK - United Kingdom
US - United States

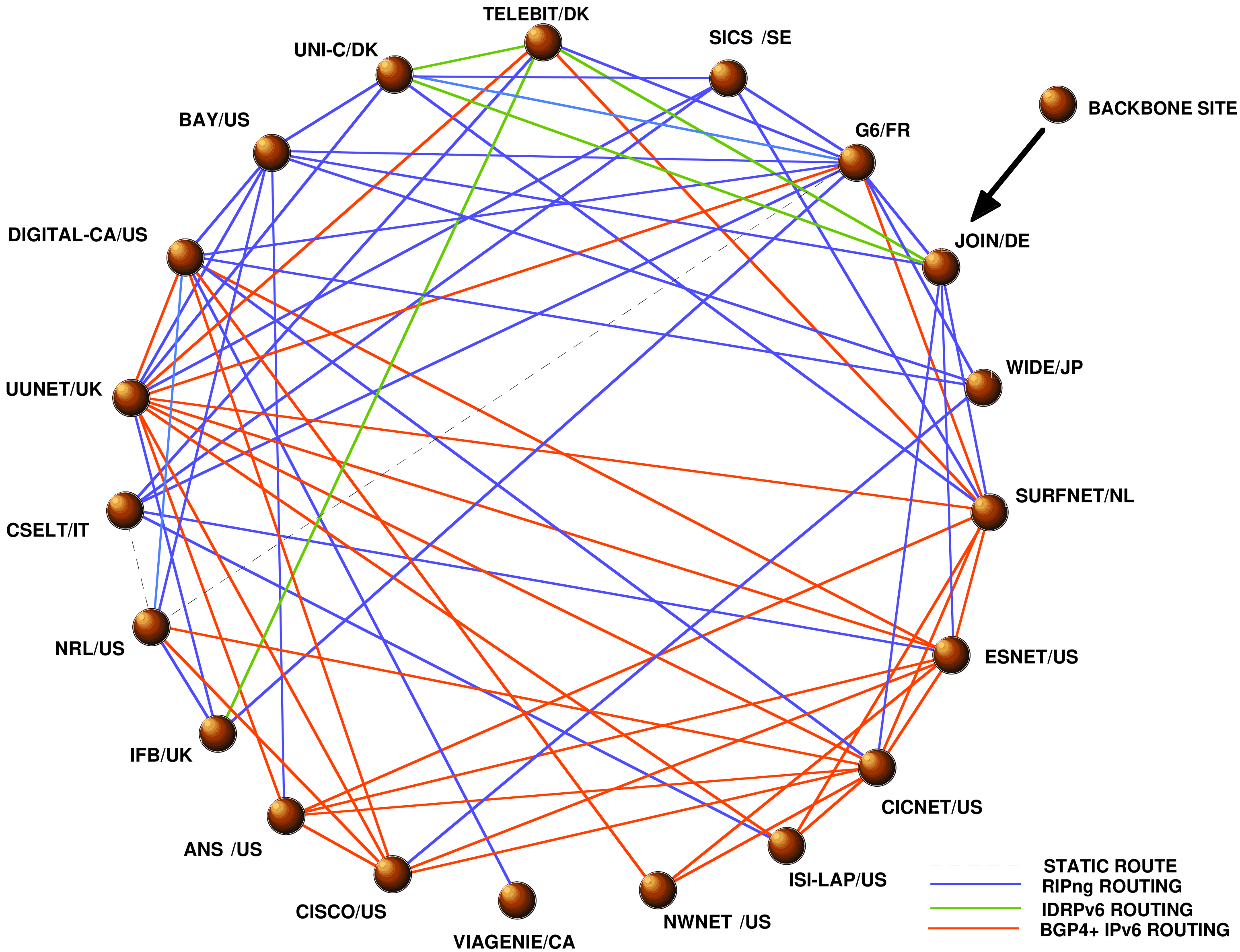
A Simple View of the 6bone



6bone Overview Diagram



6bone Backbone Diagram



Peering in the 6bone backbone

- We have learned several lessons very rapidly:
  - keep it simple, and keep it well organized
  - avoid RIPng use except in well controlled circumstances, probably entirely limited to local site use
  - don’t really expect IDRPv6 to move into the real world of ISPs very rapidly
  - build on what works now (for ISPs), thus BGP4
  - getting consensus isn’t always a talking game, getting three interoperable BGP4+ implementations to work on the 6bone did the trick

Implementations

- we estimate over 40 unique host and router implementations of IPv6 are complete, or under development, that we know of
- on the 6bone we have tested:
  - 14 host implementations
  - 13 router implementations
- it's harder to know how many are truly unique, but most of them are NOT just copies of some unix implementation
- so...lots of unique creative talent is going into IPv6

host implementations in use on the 6bone are:

- Digital OpenVMS
- FTP Software Windows95
- IBM AIX
- Linux
- Sun Solaris
- NRL for BSD
- WIDE ZETA for BSD
- Digital Unix
- Hitachi NR60
- Inria BSD
- HP-UX (SICS)
- UNH for BSD
- WIDE Hydrangea for BSD
- WIDE v6d

router implementations in use on the 6bone are:

Bay

Digital

Hitachi NR60

Linux

NRL for BSD

WIDE Hydrangea for BSD

WIDE ZETA for BSD

Cisco

Fujitsu LR550

Inria BSD

Merit MRT

Telebit

WIDE v6d

A new Unicast Addressing Format

- Besides the inevitable “running out of IPv4 addresses” problem, the current IPv4-based Internet routing infrastructure is suffering from route complexity explosion
- The more routes, different ISPs, “flaps” and whatever, means that route computation time skyrockets
- Is there an answer to this?
- IPv6 will constrain and address this problem with the new “Aggregatable” Global Unicast Format [AGGR]

Aggregatable Unicast Addressing

3	13	32	16	64
001	TLA	NLA	SLA	Interface ID

TLA = Top-Level Aggregation IDs are assigned to ISPs and Exchanges that act in a default-free way with a routing table entry for every active TLA ID (this constrains the routing complexity)

NLA = Next Level Aggregation IDs are assigned *by TLAs* to create a multi-level hierarchy underneath it as the ISP chooses (i.e., multiple NLA levels may exist for ISPs with more ISPs...to the end site)

SLA = Site Level Aggregation IDs are used to create local addressing hierarchy (e.g., a flat subnet space)

New EUI-64 Interface ID

Interface IDs are used to indentify interfaces on a link.

They follow the newer IEEE EUI-64 format (which has an extended IEEE 48-bit MAC address embedded in it)

$\text{ccccccFFFEeeeeee}_{16}$ (an EUI-48 extension)

...this helps automatic configuration while allowing expansion for future MAC address formats beyond Ethernet, like Firewire, etc.

$\text{cccccceeeeeeeeee}_{16}$ (a full EUI-64 value)

where c...c = 24 bit company code,
and e......e = 40 bit extension identifier

A new DNS idea for IPv6

- A proposal will soon be made for IPv6 DNS changes to assist with easy site renumbering given the new Aggregatable Global Unicast Addressing format
- For forward address (AAAA) lookup there will be a two part identity consisting of a pointer to the site prefix, allowing easy site-level change, and the actual site local address
- The reverse path lookup would use the formalized delegation of aggregation prefixes to provide automatic creation of the reverse (PTR) tree

How does AGGR help?

- Minimizes (or at least limits) top level routing complexity
- Allows top level ISPs and Exchanges to allocate addresses
- Provides for the new concept of Exchanges that can broker ISP service for you and not require a change in your numbering at all

Renumbering Sites

- With hierarchical addressing in IPv6 (as in IPv4), it will be necessary to have sites renumbered when they change ISPs
- A well defined “public” topology prefix separate from the “site” topology and host interface identifier helps this by allowing mechanisms to change the prefix with minimal impact, in real time
- IPv6’s stateless Autoconfiguration working with IPv6 Neighbor Discovery and Router Reconfiguration are the mechanisms to implement this renumbering

Renumbering example

- administratively lower lifetime of old routing prefix
- using router renumbering protocol, set new routing prefix(es) in site's routers
- after lifetime expires, routers will distribute (via IPv6 ND [neighbor discovery]) the new prefix to hosts
- old prefix will still work until administratively removed from routers (i.e., won't route that prefix)
- at this point, long lived tcp connections break... but this is the only non-seamless part about it.

Seamless Transition

- IPv6 is unlikely to become the Internet network layer protocol of choice unless there is literally no choice to be made...and it can operate alongside IPv4 indefinitely
- By this we mean that it must be very easy for the private network (your corporate net) and public network (your ISP) operators to equip, enable and operate IPv6 while operating IPv4
- and...that the user doesn’t notice that it is there at all. Yes, a sys admin must be conscious of it in a minimal sense, but not the user

What makes it seamless?

- At the network operator level it is just another "routing stack" that can easily be turned on (many of us certainly do that now with IPX, AppleTalk, DECnet,...)
- IPv6 inter-domain routing can be operated just like IPv4 IDR as it uses BGP just like IPv4 does
- The presence of IPv6 addresses in other sites will automatically trigger the use of IPv6 in your host system (if it has IPv6) with no difference in user level appearance
- Automatic configuration will make your host system configure itself for IPv6 without you knowing about it...at all!

Will IPv4 Go Away?

- **If it does, it will be very slow and long term**
- **One characteristic of it will be that you won’t notice it has gone away**

... just as you won’t notice IPv6 has arrived!

Important areas for 6bone activity

- Completely experience the new address structure and the new DNS (not specified yet)
- Testing site renumbering
- Testing IPv6 Security
- Testing IPv6 Multicast
- Testing user seamlessness

Will the 6bone go away someday?

- We sure hope so...
- The 6bone is a tool towards an end

  moving the Internet towards IPv6 without breaking IPv4 and the Internet you know

- Please join us if you want to help in this,
  if nothing else you will learn a lot!
- Visit us at www.6bone.net

Some Examples of Configuration

- Cisco router (native IPv6 and configured tunnels, and site peering)
- DEC Alpha (native IPv6 and configured tunnels)
- SunOS (native IPv6 and configured tunnels)
- 6bone registry (adding a site)

Cisco Router - Ethernet configured with native IPv6

interface Ethernet0/0
ip address 198.128.2.27 255.255.255.0
ip broadcast-address 198.128.2.255
ipv6 address 5F01:2500:C680:200::C51:65D8 80

In this example, the IPv6 address of the Ethernet interface is:

5F01:2500:C680:200::C51:65D8 with a mask of 80

notice leading 0s in a “:” delimited section are omitted in the cisco router configuration file notation - this is the standard way of abbreviating the long address into something more readable - also, “::” are intervening zeros

The prefix is 5F01:2500:C680:0200:0000 (80 bits)

and the host portion is 0000:0C51:65D8 (48 bits)

Cisco Router - A tunnel using a static IPv6 route

```
interface Tunnel1
description ESnet TWC to 5F01:2500:C680:0400/64
no ip address
ipv6 enable
ipv6 unnumbered Ethernet0/0
tunnel source Ethernet0/0
tunnel destination 198.128.4.74
tunnel mode ipv6ip

ipv6 route 5F01:2500:C680:400::0/64 Tunnel1
```

In this example, all IPv6 addresses destined to an address with the prefix 5F01:2500:C680:400 will be forwarded out this tunnel. The tunnel destination indicates the IPv4 endpoint of the tunnel. Several or many router hops may lie in the path between tunnel endpoints.

Cisco Router - A tunnel using BGP4+

```
ipv6 bgp neighbor 5F01:2500::293:1225:2 remote-as 1225
ipv6 bgp neighbor 5F01:2500::293:1225:2 route-map ESNET out

interface Tunnel10
description BGP4+ To CICnet: 5F04:C900::/32
no ip address
ipv6 enable
ipv6 address 5F01:2500::293:1225:1 112
tunnel source Ethernet0/0
tunnel destination 131.103.1.54
tunnel mode ipv6ip

ipv6 access-list ESNET permit 5F01:2500::0/32 priority 1
ipv6 access-list ESNET permit 5F09:C900::0/32 priority 2
ipv6 access-list ESNET permit 5F02:BD00:8006::0/64
```

In this example the IPv6 address on the tunnel endpoint's local side is 5F01:2500::293:1225:1. The IPv6 address on the tunnel endpoint's remote side is 5F01:2500::293:1225:2.

Cisco Router - A tunnel using RIPng

```
interface Ethernet0/0
ip address 198.128.2.27 255.255.255.0
ip broadcast-address 198.128.2.255
ipv6 address 5F01:2500:C680:200::C51:65D8 80
ipv6 rip 6bone enable

interface Tunnel6
description To KEK:5f09:c900::/32
no ip address
ipv6 enable
ipv6 unnumbered Ethernet0/0
ipv6 rip 6bone enable
ipv6 rip 6bone summary-address 5F01:2500::0/32
ipv6 rip 6bone output-filter ESNET-TO-KEK
tunnel source Ethernet0/0
tunnel destination 130.87.57.37
tunnel mode ipv6ip

ipv6 access-list ESNET-TO-KEK permit 5F01:2500::0/32 priority 1
ipv6 access-list ESNET-TO-KEK permit 5F02:BD00:8006::0/64
```

Notice that to enable rip another line must be added to the interface configuration (ipv6 rip 6bone enable).

DEC Alpha Configuration Examples

The IPv6 configuration file is located in:

/sbin/rc3.d/S06.50ipv6

Following are a few examples taken from a configuration file where the host is dual-stack ipv4/ipv6.

Note that in some cases a reboot is required after making changes (not sure if this is a standing feature).

This is a sh script that executes on initialization. The examples show portions of it that are specific to a particular configuration.

DEC Alpha - Ethernet configured with native IPv6

```
#
# Add manual address to LAN interface
#
ip6_aifaddr ln0 5F01:2500:C680:0400::0800:2BBC:E96C
#
# Create on-link route
#
route6 add -inet6 -cloning
5F01:2500:C680:0400::0800:2BBC:E96C/80
$IPV6_LAN_LLADDR -interface
```

DEC Alpha - a static route

```
#
# Create default route
#
# route6 add -inet6 -cloning default FE80::0000
route6 add -inet6 5F00::/8 ::198.128.2.27
```

Note that this also creates the manual tunnel. The other endpoint of the tunnel is a cisco and it's configuration is shown in one of the Cisco examples.

SunOS Configuration Examples

The IPv6 configuration file is located in:

/etc/rc2.d/S70ipv6

Following are a few examples taken from a configuration file where the host is dual-stack ipv4/ipv6.

Note that in some cases a reboot is required after making changes (not sure if this is a standing feature).

SunOS - Ethernet configured with native IPv6

```
echo "Creating link local address for interface ip1"
/usr/ipv6/sbin/ifconfig ip1:1 ipv6addr fe80::80030b2d
ipv6dst fe80:: ipv6mask 10 up

/usr/ipv6/sbin/ifconfig le0\#v6 plumb6
/usr/ipv6/sbin/ifconfig le0\#v6  up
```

SunOS - tunnel pointing default to a remote endpoint

echo "Creating a configured tunnel to 5F01:2500:C680:200::C51:65D8"

/usr/ipv6/sbin/ifconfig ip1.atun plumb6

/usr/ipv6/sbin/ifconfig ip1 ipv4tsrc 128.3.11.45 ipv4tdst 198.128.2.27 ipv6addr

5F01:2500:8003:800:0:800:2005:1CAB ipv6dst
5F01:2500:C680:200::C51:65D8 ipv6mask 128 up

SunOS - static default route

**/usr/ipv6/sbin/route add ipv6 5f00::
5F01:2500:C680:200::C51:65D8 8**

SunOS - turning in IPv6 forwarding

Takendirectly out of the /etc/rc2.d/S70ipv6 file:

```
# Configure any IPv6 interfaces. The name of an IPv6 interfaces
# is of the form <name><unit>#v6. That is they are the same as IPv4
# interfaces only have a #v6 appended. The "ipv6router" subcommand
# sets the interface to be a router and "ipv6advrt" allows router
# advertisments to be sent from this interface. "ipv6rttime"
# specifies the time advertised to other hosts that this node is
# a default router.
#
# /usr/ipv6/sbin/ifconfig <interface>#v6 plumb6
# /usr/ipv6/sbin/ifconfig <interface>#v6 [ ipv6router ] [ ipv6advrt ] up
```

6bone Registry - Add a new IPv6 Site

```
To: auto-dbm@isi.edu
Subject: Update ESnet 6bone entry
---------------
password:    <clear-text-password>
ipv6-site:   ESNET
origin:      AS293
descr:       ESnet - Energy Science Network
descr:       Lawrence Berkeley National Laboratory
descr:       1 Cyclotron Road
descr:       Berkeley, CA 94720
country:     US
prefix:      5F01:2500::/32
application: ping esnet-v6r1.es.net
application: ping tw4.es.net
tunnel:      IPv6 in IPv4 esnet-v6r1.es.net -> ispy.cisco.com CISCO BGP4+
...
```

6bone Registry - Add a new IPv6 Site - contd.

```
...
tunnel:     IPv6 in IPv4 esnet-v6r1.es.net -> yucc.kek.jp KEK RIPng
contact:    RLN1-6BONE
contact:    ES1-6BONE
remarks:    Operational
remarks:    Prefix Announced by ESnet: 5F01:2500::0/32 - ESnet
remarks:    Prefix Announced by ESnet: 5F09:C900::0/32 - KEK
remarks:    Prefix Announced by ESnet: 5F02:BD00:8006::0/64 - Rutgers
url:        http://www.es.net
notify:     routing@es.net
mnt-by:     MAINT-ESNET-V6
changed:    nitzan@es.net 19970801
source:     6BONE
```