IPv6
Transition and Coexistence

APNIC 44
Taichung - Taiwan
September 2017

Jordi Palet (jordi.palet@theipv6company.com)

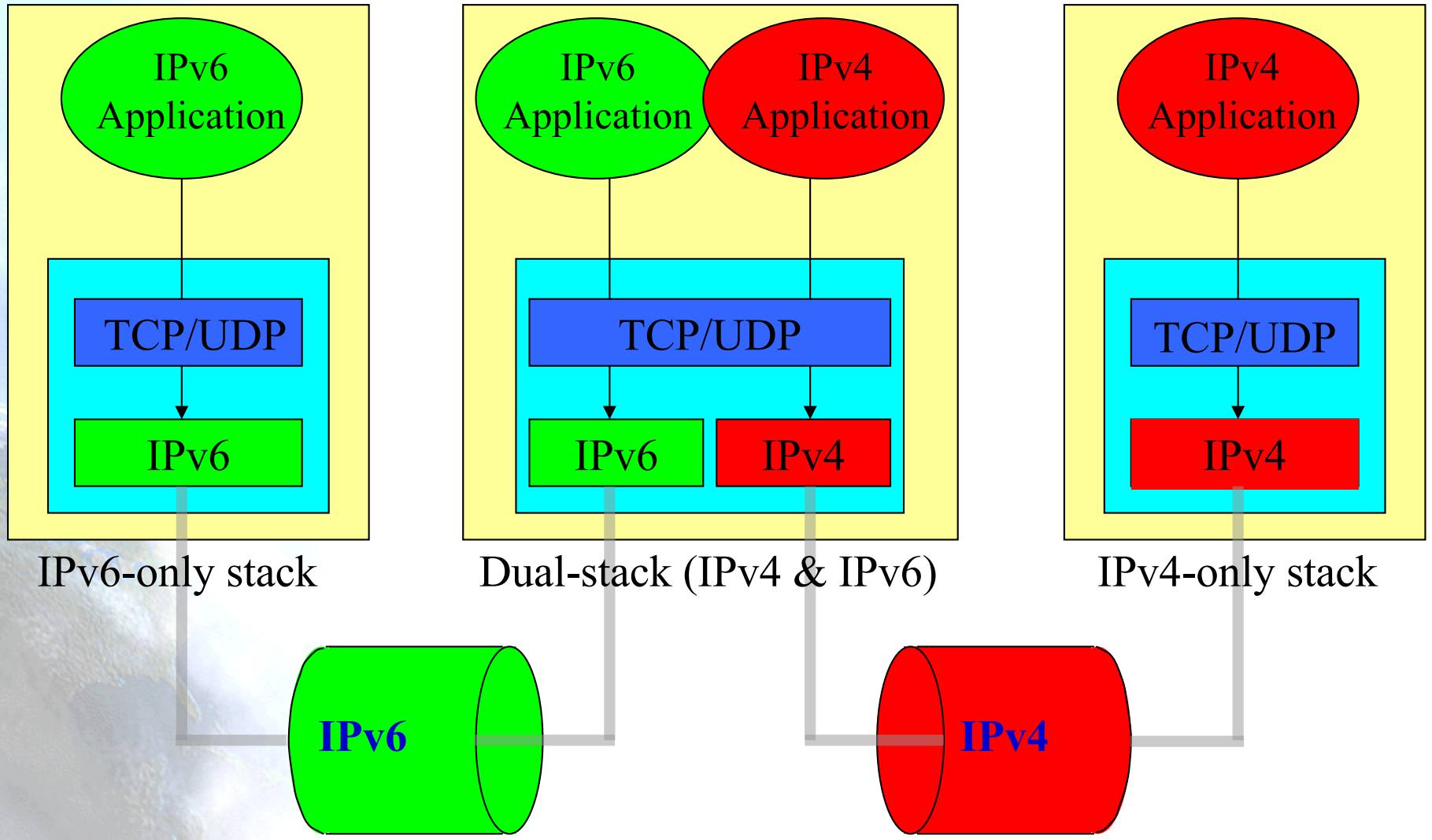
Transition / Co-Existence Techniques

- IPv6 has been designed for easing the transition and coexistence with IPv4
- Several strategies have been designed and implemented for coexisting with IPv4 hosts, grouped in three categories:
  - Dual stack: Simultaneous support for both IPv4 and IPv6 stacks
  - Tunnels: IPv6 packets encapsulated in IPv4 ones
    - This has been the commonest choice
    - Today expect IPv4 packets in IPv6 ones!
  - Translation: Communication of IPv4-only and IPv6-only. Initially discouraged and only “last resort” (imperfect). Today no other choice!
- Expect to use them in combination!

Dual-Stack Approach

- When adding IPv6 to a system, do not delete IPv4
  - This multi-protocol approach is familiar and well-understood (e.g., for AppleTalk, IPX, etc.)
  - In the majority of the cases, IPv6 is be bundled with all the OS release, not an extra-cost add-on
- Applications (or libraries) choose IP version to use
  - when initiating, based on DNS response:
    - if (dest has AAAA record) use IPv6, else use IPv4
  - when responding, based on version of initiating packet
- This allows indefinite co-existence of IPv4 and IPv6, and gradual app-by-app upgrades to IPv6 usage
- A6 record is experimental

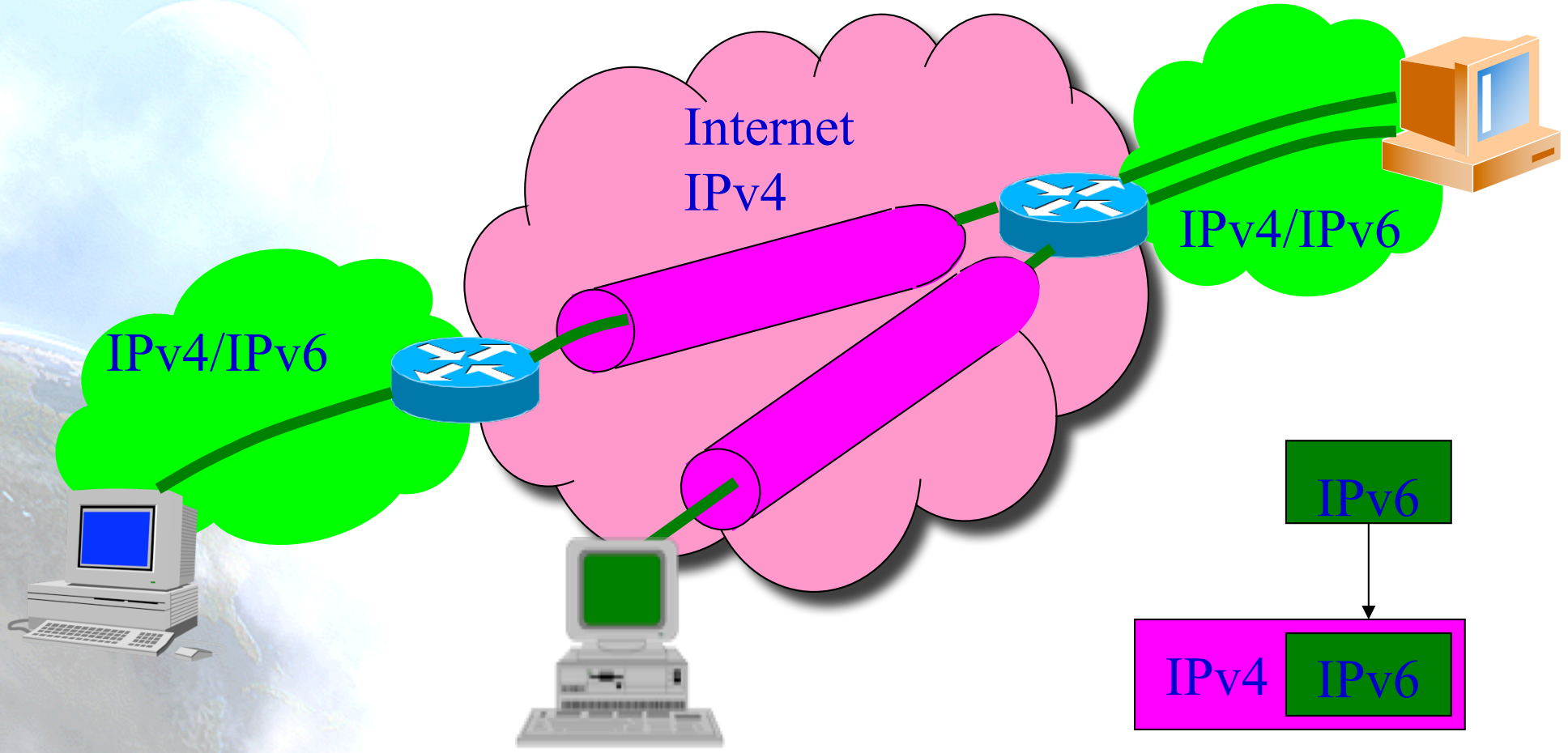
Dual-Stack Approach



Tunnels to Get Through IPv6-Ignorant Routers

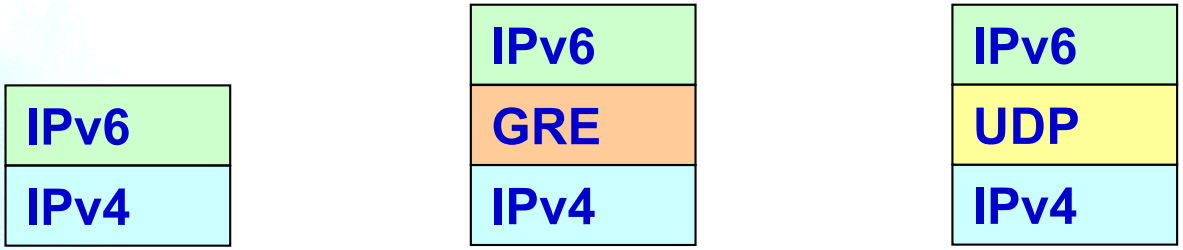
- Encapsulate IPv6 packets inside IPv4 packets (or MPLS frames) in order to provide IPv6 connectivity through IPv4-only networks
- Many methods exist for establishing tunnels:
  - manual configuration
  - "tunnel brokers" (using web-based service to create a tunnel)
  - "6over4" (intra-domain, using IPv4 multicast as virtual LAN)
  - "6to4" (inter-domain, using IPv4 addr as IPv6 site prefix)
- Can view this as:
  - IPv6 using IPv4 as a virtual link-layer, or
  - an IPv6 VPN (virtual public network), over the IPv4 Internet (becoming "less virtual" over time, we hope)

Tunnels IPv6 in IPv4 (1)



Tunnels IPv6 in IPv4 (2)

- There are different ways for encapsulating the IPv6 packets into IPv4 ones



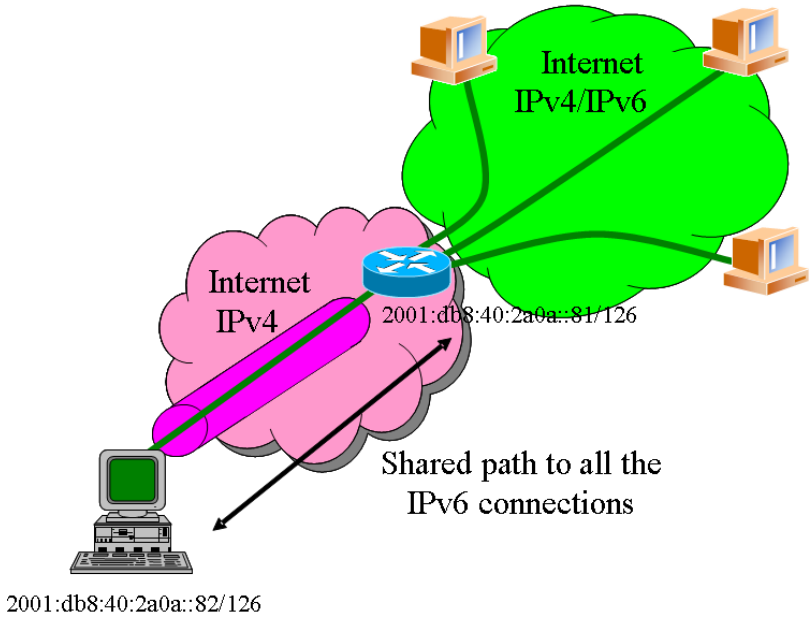
- Same for IPv4 being used in IPv6-only networks

Tunnels IPv6 in IPv4 (3)

- Some transition mechanism based on tunnels:
  - 6in4 [6in4]
  - TB [TB]
  - TSP [TSP]
  - 6to4 [6to4]
  - Teredo [TEREDO], [TEREDOC]
  - Túneles automáticos [TunAut]
  - …
  - ISATAP [ISATAP]
  - 6over4 [6over4]
  - Softwires
  - 6RD
  - NAT64
  - DS-Lite
  - 464XLAT
  - MAP E/T

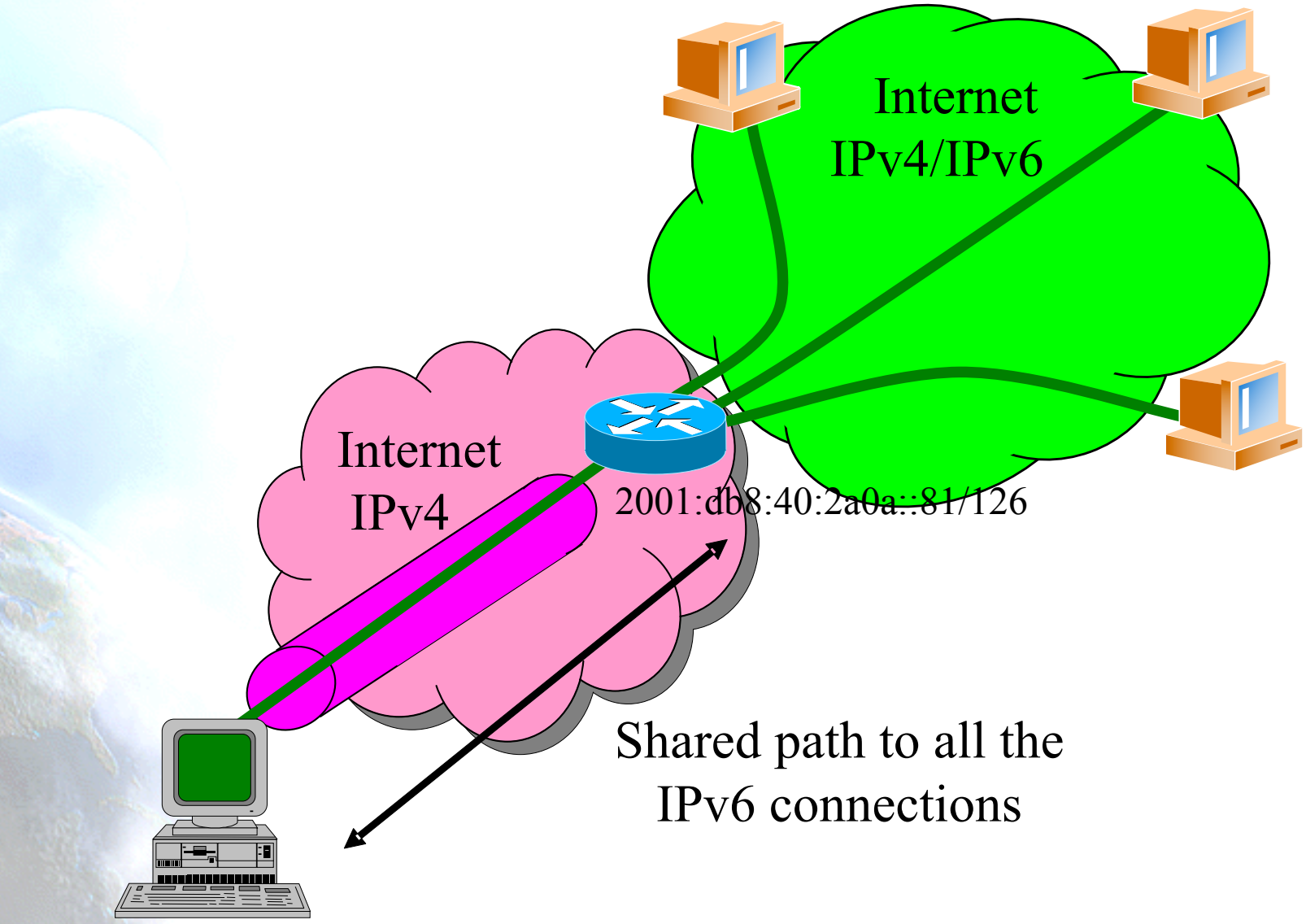
6in4 Tunnels Details

- It encapsulates directly the IPv6 packet into the IPv4 packet
- It is usually used between:
  - end host ==> router
  - router ==> router



- However, it is also possible for
  - end host ==> end host
- From the point of view of IPv6 the tunnel is considered as a point-to-point link
  - Only an IPv6 network-hop although several IPv4-hops exist in the path
- The IPv6 addresses of both tunnel-ends belong to the same prefix
- All the IPv6 connections of the end-host flow always through the router located at the tunnel-end-point
- The 6in4 tunnels can be built from end-hosts located behind a NAT box
  - It is essential that the NAT implementation supports “proto-41 forwarding” [PROTO41] to let the IPv6-encasulated packets traverse the NAT box

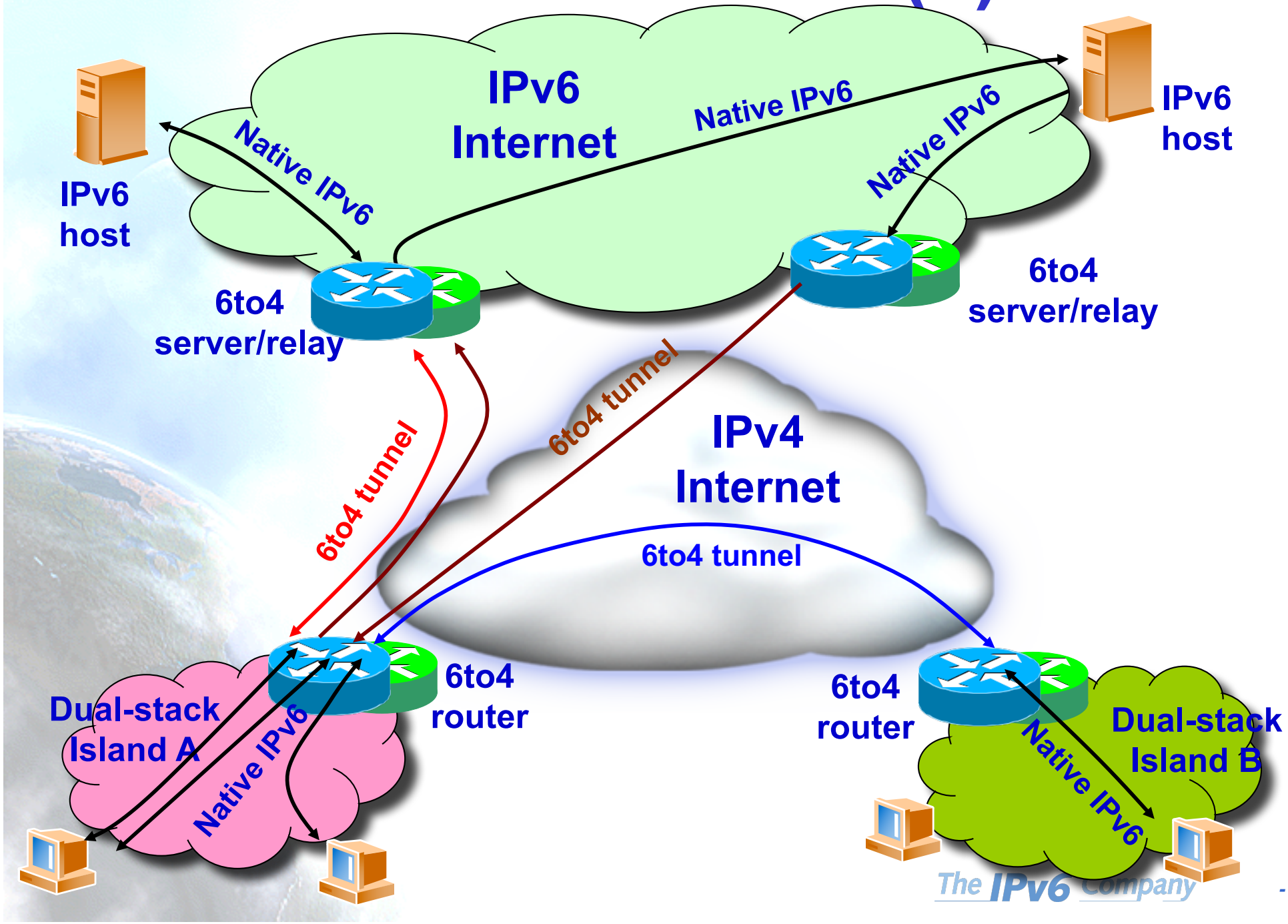
Tunnel Broker



Tunnel Broker [RFC3053]

- The 6in4 tunnels require the manual configuration of the devices involved in the tunnel creation
- To easy the address assignment and the IPv6 tunnel creation, the Tunnel Broker (TB) concept has been developed
  - It is a intermediate host which the end user is connected, usually by using a web browser
- The user asks to the TB the creation of an IPv6 tunnel. The TB assigns to the user an IPv6 address and gives to the user instructions for building the tunnel in the user's side
- The TB also configures the router, which is the TEP for the end user
- In http://www.ipv6tf.org/using/connectivity/test.php exists a list of available TBs
- TSP [TSP] is a special case of TB because it is based on an application installed in the user's host which contacts to the TSP server to built the IPv6 tunnel. However, the concept is similar to the one previously enounced

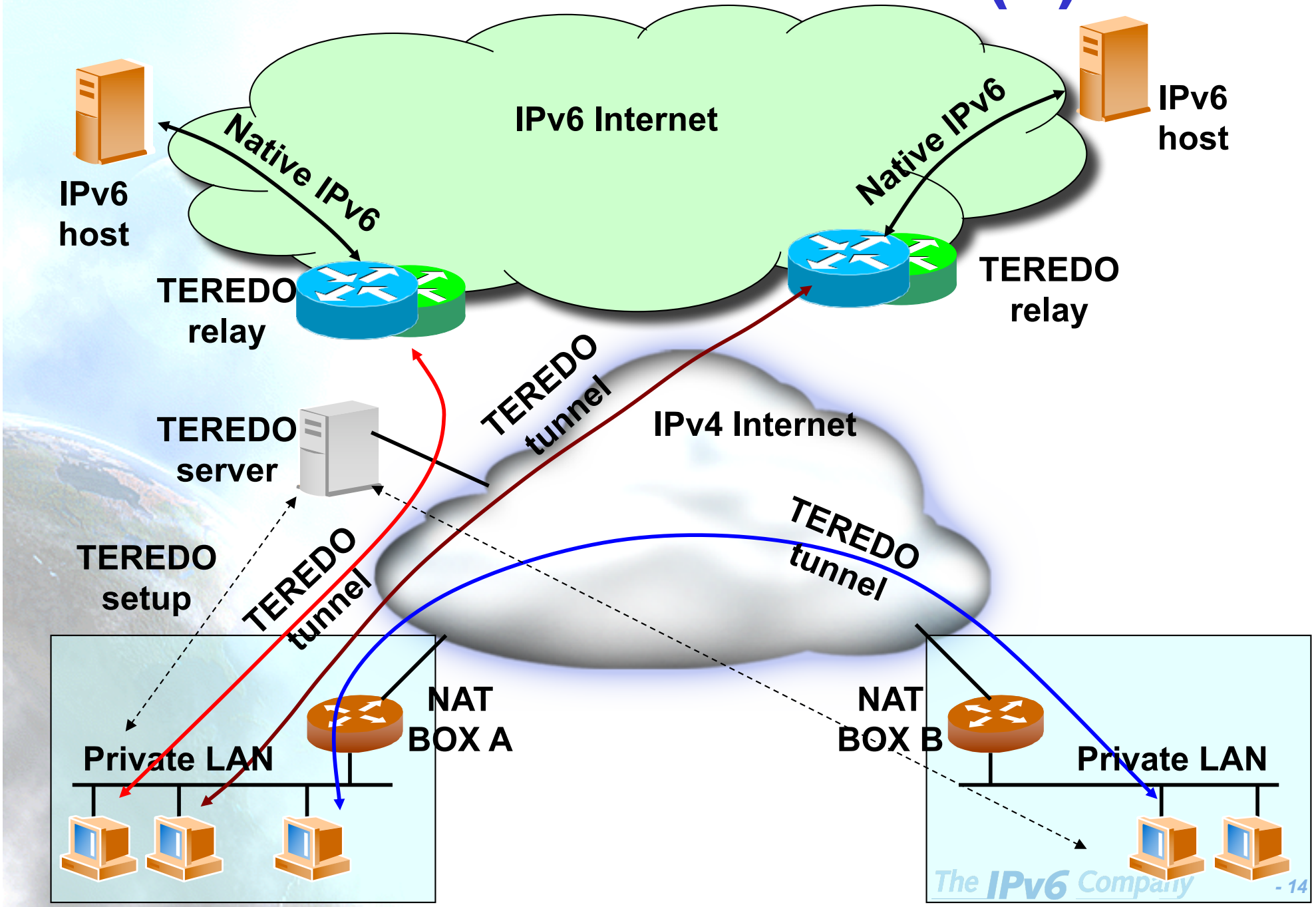
6to4 Tunnels (1)



6to4 Tunnels (2)

- Defined on [RFC3056]
- IPv6 packets are encapsulated into IPv4 ones, in a similar way than the 6in4 tunnels
- Differences:
  - The user's IPv6 address does not depend on the router used to get IPv6 connected but on the public IPv4 used by the user
    - Prefix 2002::/16
  - All the user's outgoing IPv6 packets are always sent to the same "6to4 relay". However the user's incoming IPv6 packets could come from different "6to4 relays"
- IPv4 anycast prefix:
  - 192.88.99.1 [RFC3068] (deprecated)

Teredo: RFC4380 (1)



Teredo: RFC4380 (2)

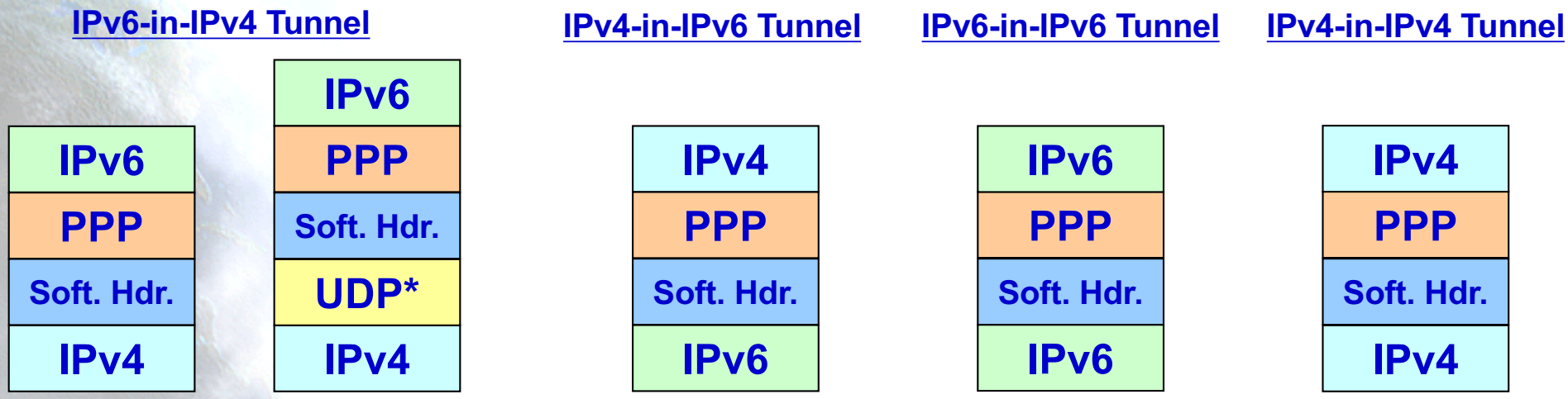
- Teredo [TEREDO] [TEREDOC] is thought for providing IPv6 to hosts that are located behind a NAT box that is not "proto-41 forwarding"
  - It encapsulates the IPv6 packets into UDP/IPv4 packets
- It only works in the following NAT types:
  - Full Cone
  - Restricted Cone
- It does not work in the following NAT type:
  - Symmetric (Solved in Windows Vista)
- Teredo uses different agents to work:
  - Teredo Server
  - Teredo Relay
  - Teredo Client
- The user configures in its host a Teredo Server which provides an IPv6 address from the 2001:0000::/32 prefix and such an address is based on the user's public IPv4 address and used UDP port
  - If the Teredo Server is also a Teredo Relay, the user has also IPv6 connectivity with any IPv6 hosts
  - Otherwise, the user only has IPv6 connectivity with other Teredo users
- Microsoft currently provides public Teredo Servers for free, but not Teredo Relays

Softwires

- Protocol being discussed within IETF's Softwire WG. Characteristics:
  - "Universal" transition mechanism based on tunnels
    - IPv6-in-IPv4, IPv6-in-IPv6, IPv4-in-IPv6, IPv4-in-IPv4
    - NAT traversal on access networks
    - Provides IPv6 prefix delegation (/48, /64, etc.)
    - User authentication for tunnel creation using AAA infrastructure
    - Possibility of secure tunnels
    - Low overhead of IPv6 packets over the tunnels
    - Supports portable devices with scarce hardware resources
  - Will enable provision of IPv6 connectivity to devices like ADSL routers, mobile phones, PDAs, etc. when no native IPv6 connectivity exists
  - Could provide IPv4 connectivity to devices with IPv6 only connectivity
- Softwires is not a new protocol but the definition of how to use existing protocols in order to provide IPv6 connectivity on IPv4 only networks and vice versa
- It is based on:
  - L2TPv2 (RFC2661)
  - L2TPv3 (RFC3991)

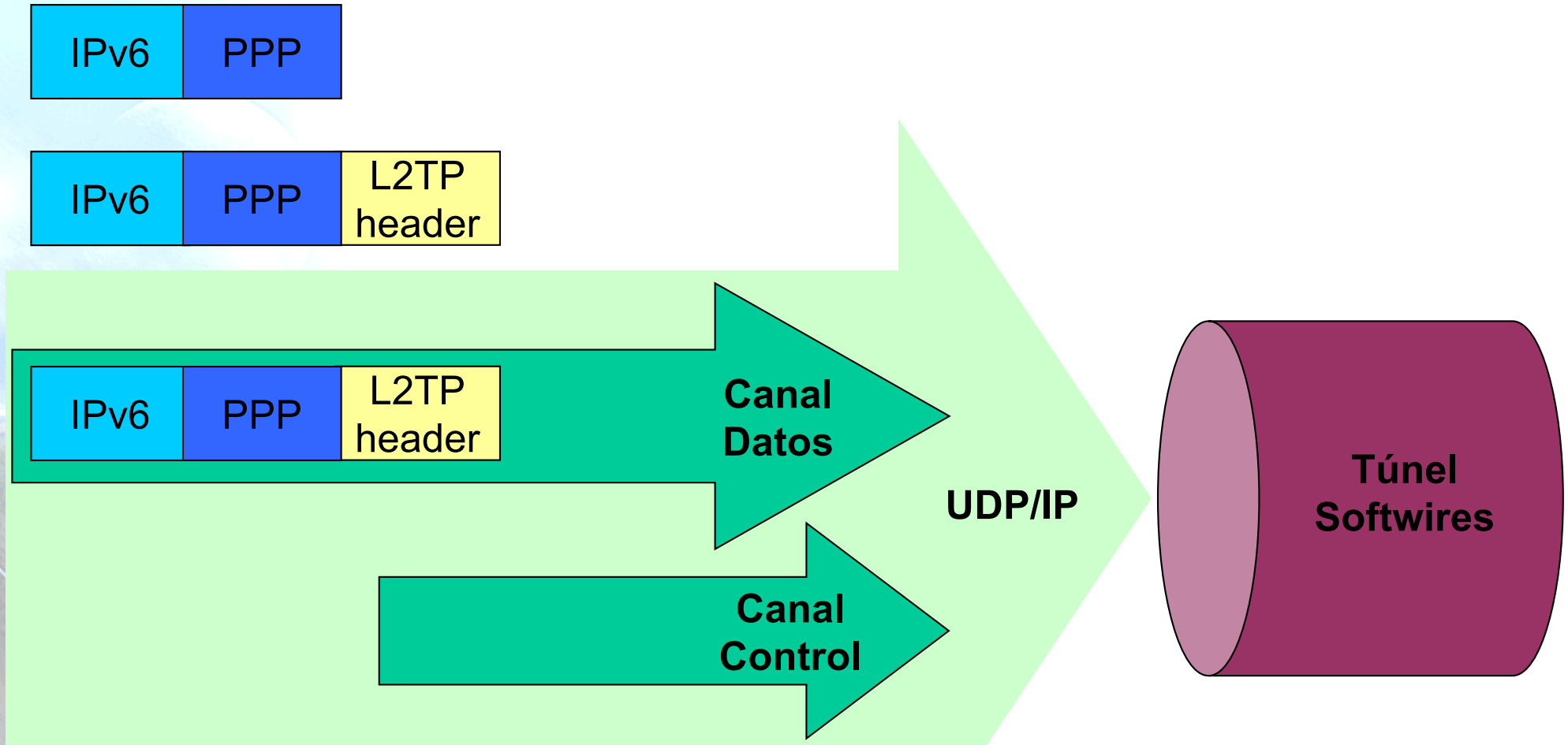
Softwires Encapsulating based on L2TPv2

- Described on draft-ietf-softwire-hs-framework-l2tpv2
- There are two entities:
  - Softwires Initiator (SI): agent who solicits the tunnel
  - Softwires Concentrator (SC): agent who creates the tunnel (tunnel end-point)
- PPP is used to transport IPvx (x=4 or 6) in IPvx (x=4 or 6) packets
  - Optionally PPP packets can be encapsulated on UDP for NAT traversal



* Optional

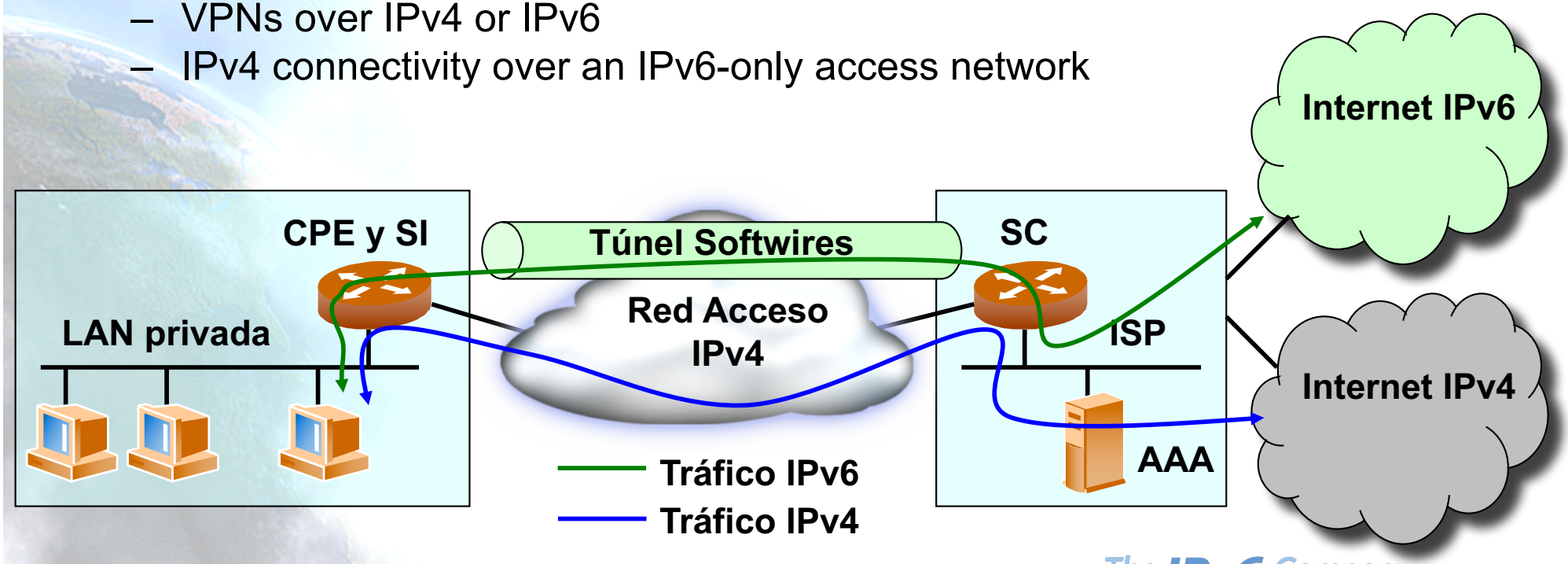
Softwires Based on L2TPv2



- There are a Control and a Data Plane
- PPP is used as an encapsulating protocol

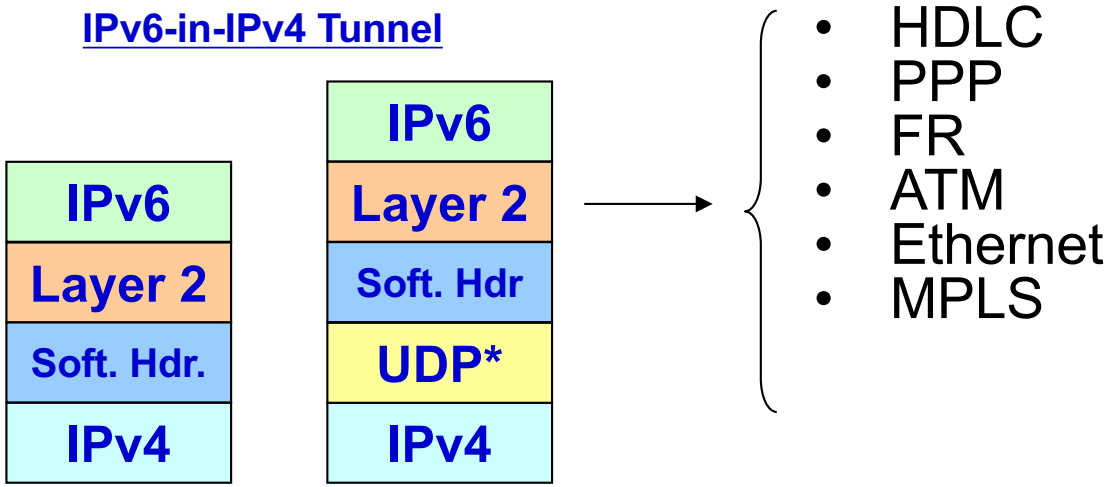
Example of Use

- An expected use of Softwires is for providing IPv6 connectivity to domestic users through an IPv6-only access network
  - The SC is on ISP's network (DSLAM, Aggregation Router, or other device)
  - The SI is on user's network (the CPE or other device)
  - The SC provides IPv6 connectivity to the SI and the SI act as IPv6 router for user networks
  - Prefix delegation (DHCP-PD) is used between the SC and the SI to provide an IPv6 prefix (typically a /48)
- Other uses are possible:
  - VPNs over IPv4 or IPv6
  - IPv4 connectivity over an IPv6-only access network



Softwires Encapsulating Based on L2TPv3

- Same philosophy as with L2TPv2 but with L2TPv3 particularities:
  - Transport over IP/UDP of other layer two protocols different than PPP:
    - HDLC, FR, ATM, Ethernet or MPLS
  - Enhanced header format for better performance in the SC
    - T1/E1, T3/E3, OC48
  - Minimum overhead on encapsulated packets (only 4 to 12 extra bytes)
  - Adds EAP as authentication mechanism to CHAP and PAP used in L2TPv2

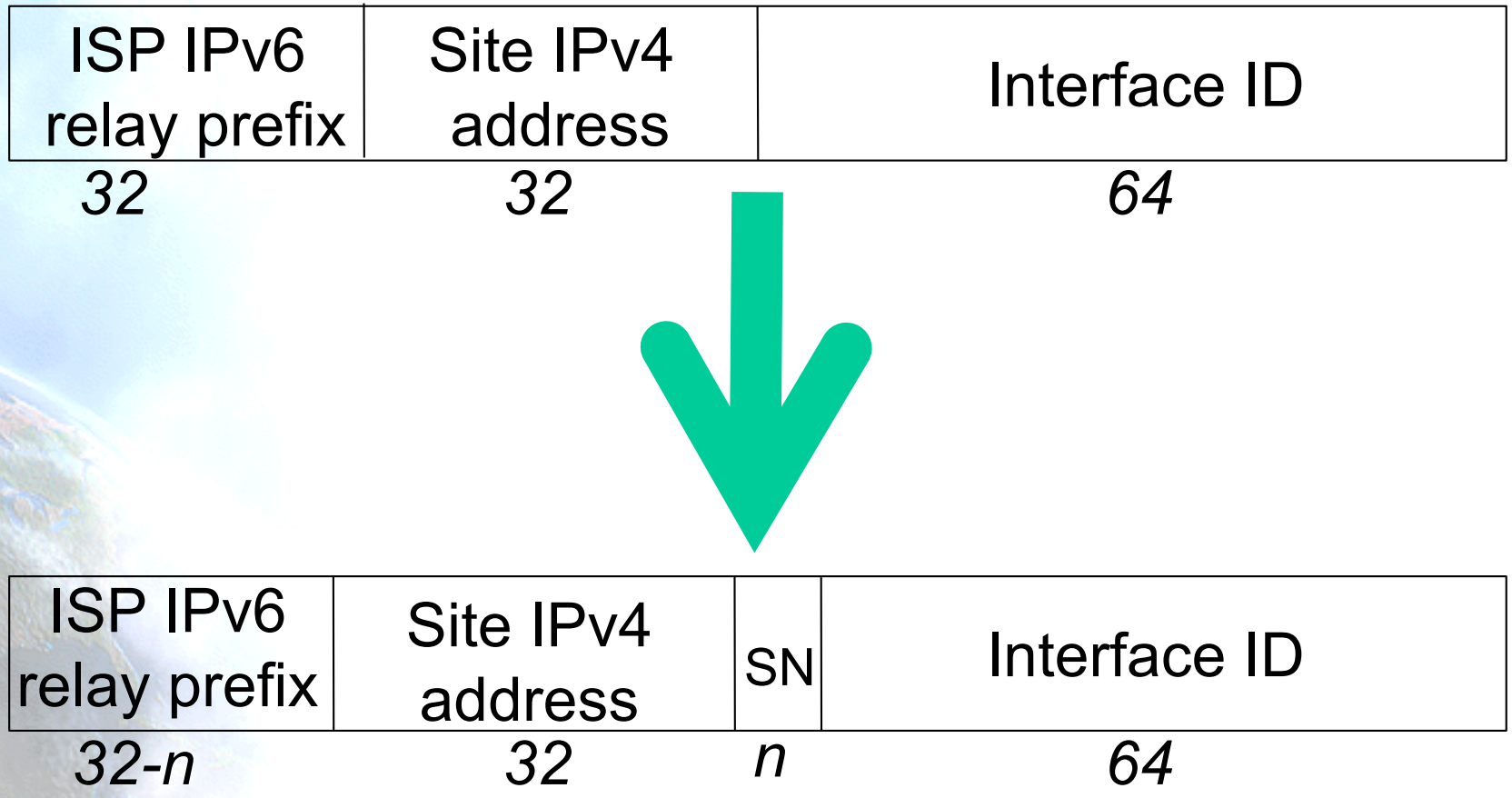


* Optional

6RD: Refining 6to4 …

- 6RD: IPv6 Rapid Deployment in IPv4 infrastructures
  - 6RD depende de IPv4
- RFC5969
- Implemented originally by FREE (French ISP)
- Changes from 6to4:
  - Addressing format
  - Relays (6rd gateway) only inside the ISP

6RD: Addressing Format



6RD: Pros and Cons

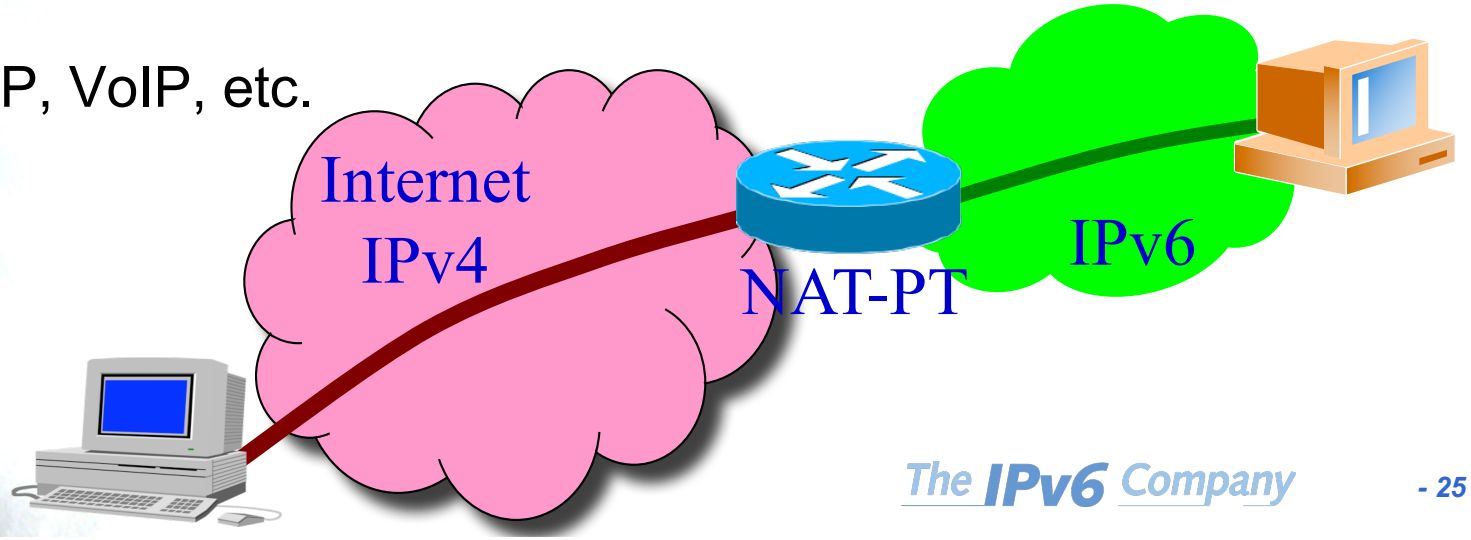
- Pros
  - Seems easy to implement and deploy if network gears are « under control » (CPEs, …)
  - Solve all (?) the 6to4 issues
    - security, asymmetric routing, …
    - Relay (or gateway) is in the ISP network then under its control
  - Transparent for the customer
    - Automatic configuration of the CPE
  - Works with public as well as private IPv4 addresses
    - allocated to the customer
- Cons
  - Not well supported by RIRs
    - Less subnets per customer
  - Change the code running on all the CPEs
    - Only few of them support it
  - Add a new box: 6RD relay/gateway

Translation IPv4/IPv6 (1)

- May prefer to use IPv6-IPv4 protocol translation for:
  - new kinds of Internet devices (e.g., cell phones, cars, appliances)
  - benefits of shedding IPv4 stack (e.g., serverless autoconfig)
- This is a simple extension to NAT techniques, to translate header format as well as addresses
  - IPv6 nodes behind a translator get full IPv6 functionality when talking to other IPv6 nodes located anywhere
  - they get the normal (i.e., degraded) NAT functionality when talking to IPv4 devices
  - methods used to improve NAT functionality (e.g, RSIP) can be used equally to improve IPv6-IPv4 functionality

Translation IPv4/IPv6 (2)

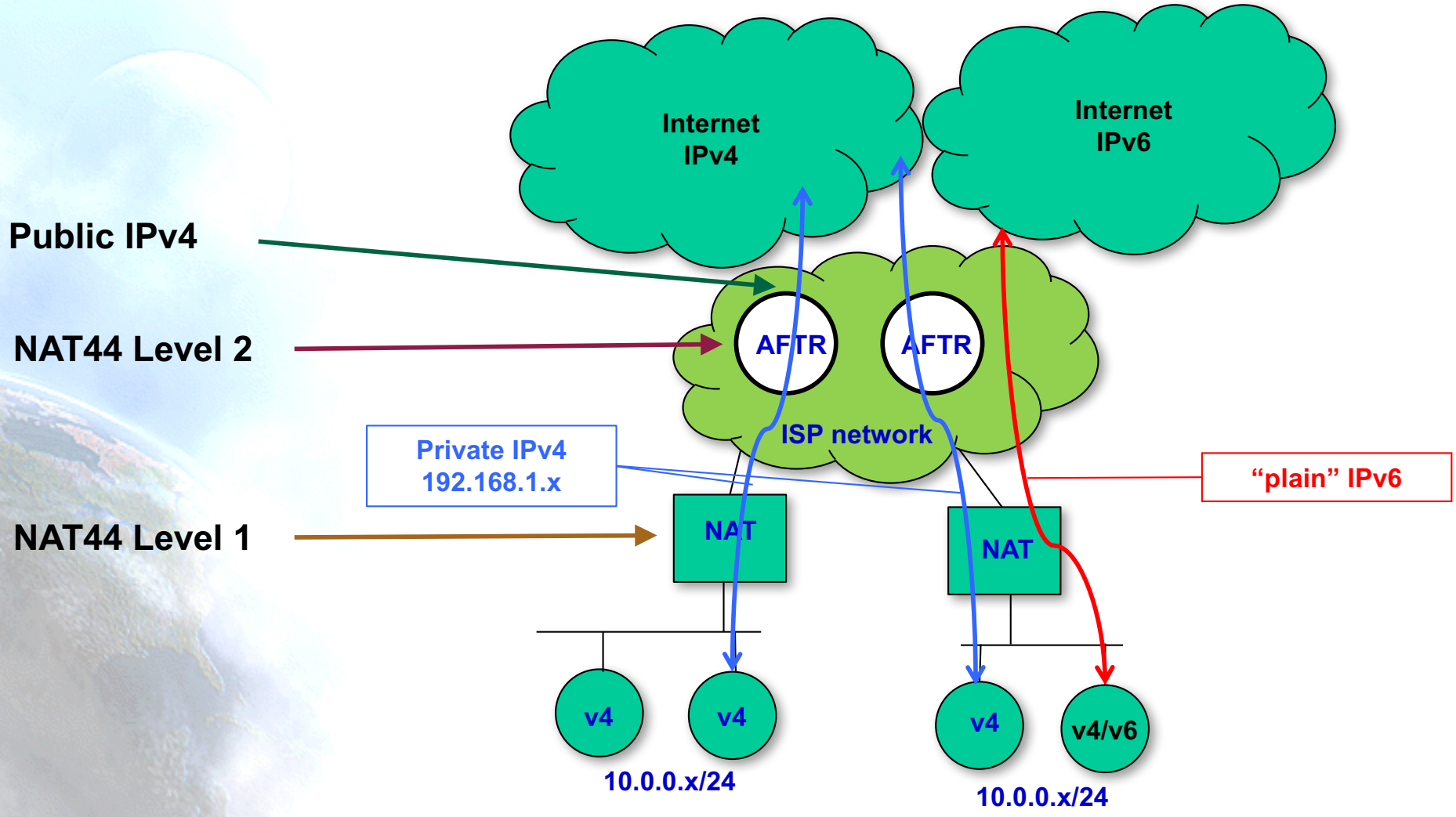
- There are several solutions, but all of them try to translate IPv4 packets into IPv6 and vice-versa
  - [SIT], [BIS], [TRT], [SOCKSv64]
- The commonest is NAT-PT [NATPT], [NATPTIMPL]
  - An intermediate node (router) modifies the IPv4 headers to convert them into IPv6 headers
  - The treatment of the packets is complex
- It is the worst solution because the translation is not perfect and it requires ALGs support, in the same way that IPv4-NATs
  - DNS, FTP, VoIP, etc.



NAT444 (1)

- Known also as CGN, CGNAT or LSN
  - Standard name AFTR (Address Family Transition Router)
- Only allows **artificially** extending IPv4 lifetime
- Doesn't allows deploying IPv6
  - Don't requires replacing the CPE
- Sharing SAME IPv4 addresses among several customers, by combining:
  - NAT + NAT
- Requires several NAT levels
- Applies NAT and PAT (Port Address Translation)
- Requires ALGs (Application Layer Gateways)

NAT444 (2)


Internet
IPv4
Internet
IPv6
Public IPv4
NAT44 Level 2
AFTR
AFTR
ISP network
Private IPv4
192.168.1.x
"plain" IPv6
NAT44 Level 1
NAT
NAT
v4
v4
v4
v4/v6
10.0.0.x/24
10.0.0.x/24

CGN breaks …

- UPnP-IGD (Universal Plug & Play - Internet Gateway Device protocol)
- NAT-PMP (NAT Port Mapping Protocol)
- Other NAT Traversal mechs
- Security
- AJAX (Asyncronous Javascript And XML)
- FTP (big files)
- BitTorrent/Limewire (seeding – uploading)
- On-line gaming
- Video streaming (Netflix, Hulu, …)
- IP cameras
- Tunnels, VPN, IPsec, ...
- VoIP
- Port forwarding
- ...

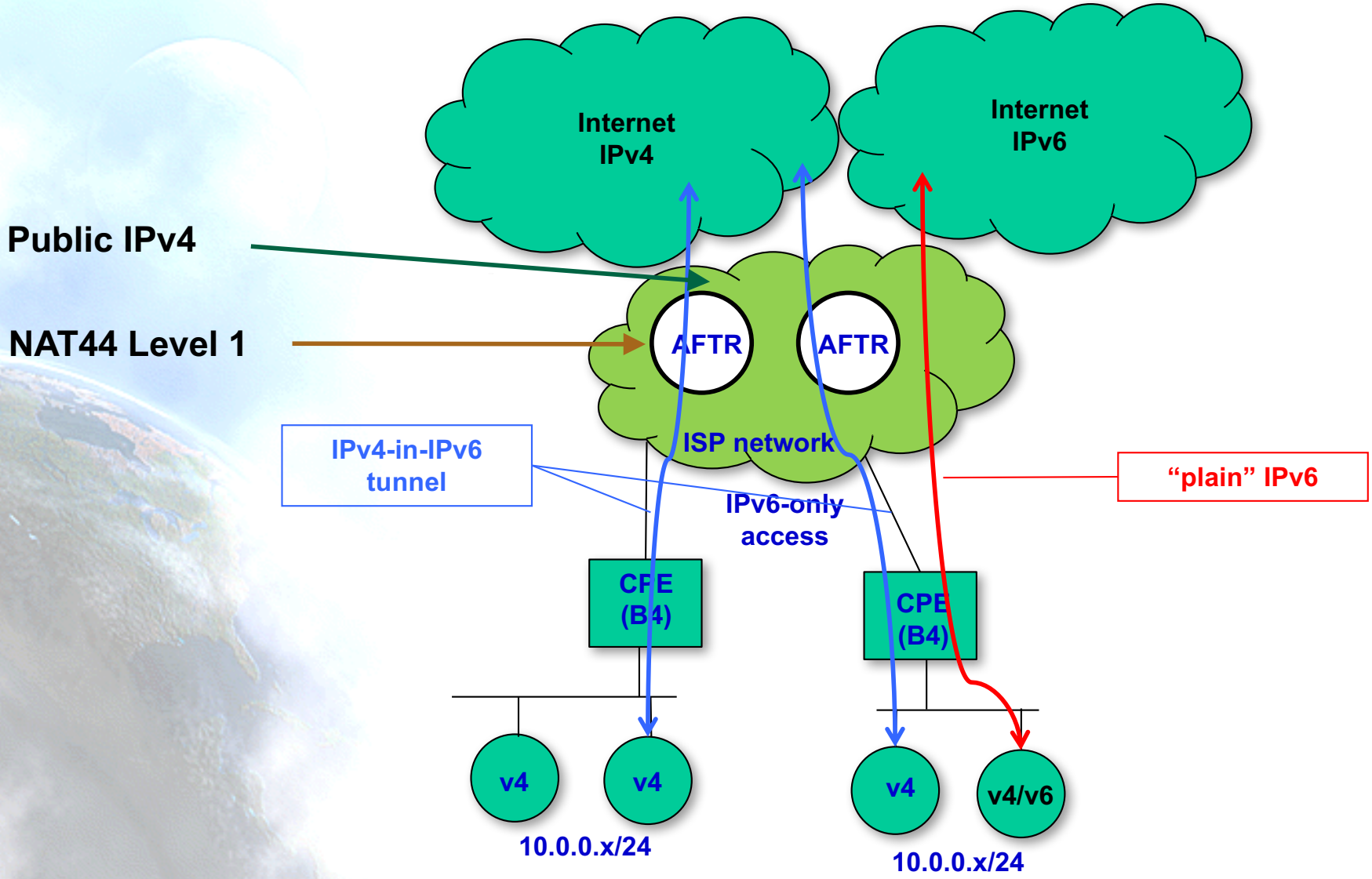
We don’t have IPv4 …

- **IPv4 exhaustion avoids**
  - **Assigning IPv4 to end-users**
  - **Assigning IPv4 even in public networks**
  - **Keep scalable interoperability with IPv4-only networks**
- **Consequence: In many cases, we need to deploy IPv6-only networks**
  - OpEx
  - No IPv4 resources (CapEx if you buy them)
  - Performance
  - Efficiency
  - RFCs
  - Other issues …

Dual Stack Lite (DS-Lite)

- To cope with the IPv4 exhaustion problem.
- Sharing (same) IPv4 addresses among customers by combining:
  - Tunneling
  - NAT
- No need for multiple levels of NAT.
- Two elements:
  - DS-Lite Basic Bridging BroadBand (B4)
  - DS-Lite Address Family Transition Router (AFTR)
    - Also called CGN (Carrier Grade NAT) or LSN (Large Scale NAT)

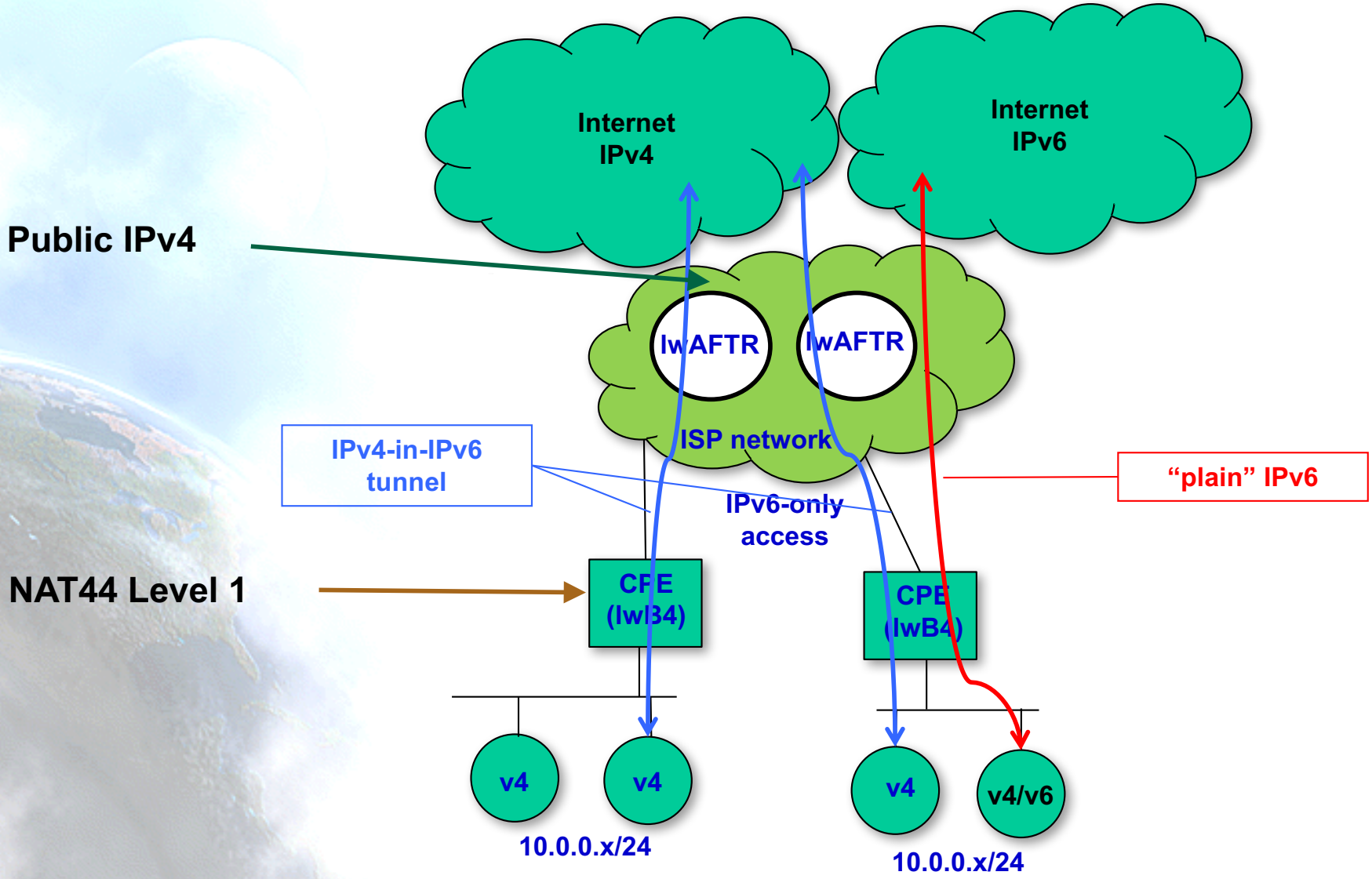
DS-Lite


Internet
IPv4
Internet
IPv6
Public IPv4
NAT44 Level 1
AFTR
AFTR
ISP network
IPv4-in-IPv6
tunnel
IPv6-only
access
"plain" IPv6
CPE
(B4)
CPE
(B4)
v4
v4
v4
v4/v6
10.0.0.x/24
10.0.0.x/24

Lightweight 4over6 (lw4o6)

- Similar to DS-Lite -> Changes NAT location
  - Better scalability
  - Reduces logging
- Sharing SAME IPv4 addresses among several customers, combining:
  - Tunneling
  - NAT
- No need for multiple levels of NAT
- Two elements:
  - Lw Basic Bridging BroadBand (lwB4) - CPE
  - Lw Address Family Transition Router (lwAFTR)

lw4o6


Internet
IPv4
Internet
IPv6
Public IPv4
lwAFTR
lwAFTR
ISP network
IPv4-in-IPv6
tunnel
IPv6-only
access
"plain" IPv6
NAT44 Level 1
CPE
(lwB4)
CPE
(lwB4)
v4
v4
v4
v4/v6
10.0.0.x/24
10.0.0.x/24

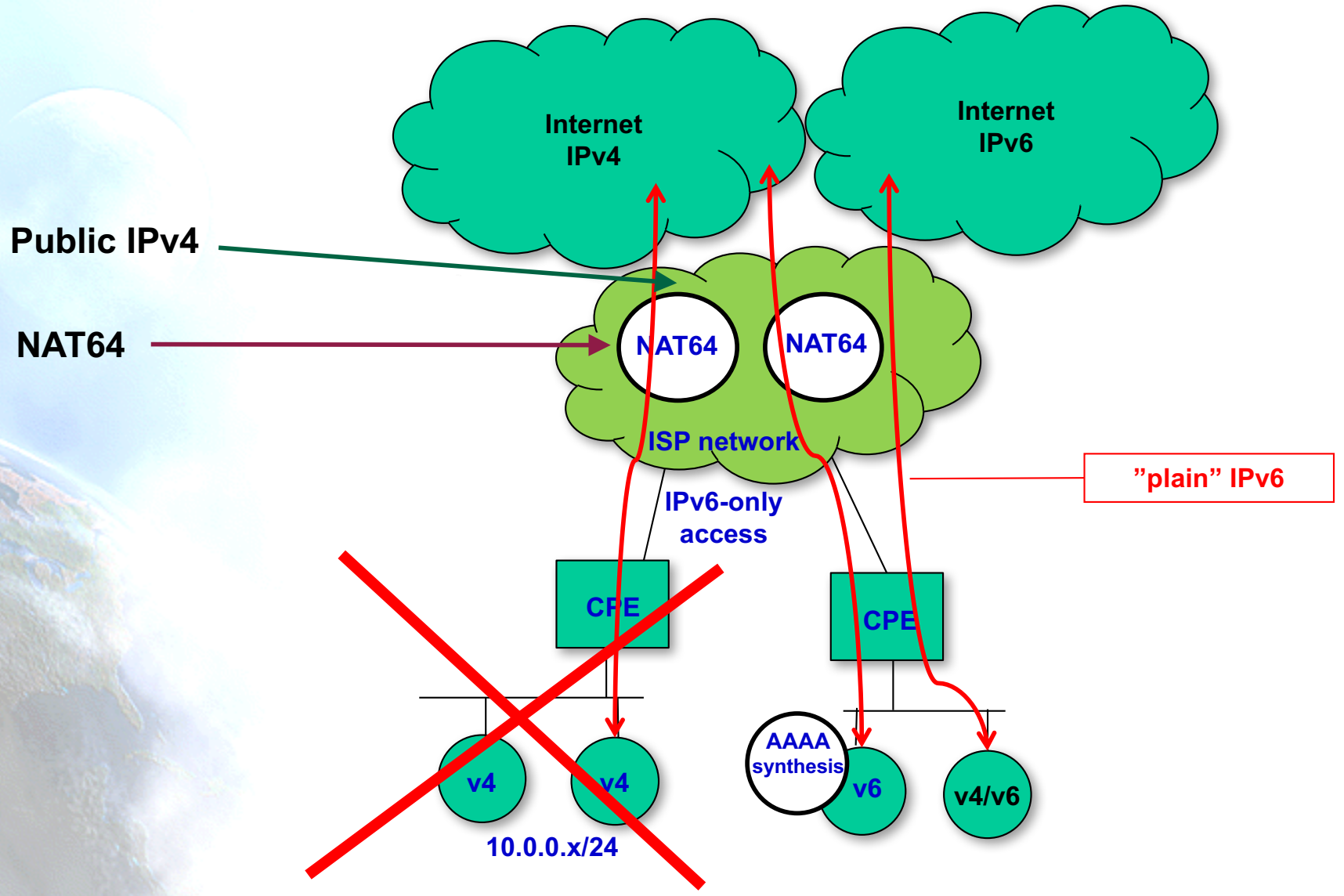
NAT64 (1)

- When ISPs only provide IPv6 connectivity, or devices are IPv6-only (cellular phones)
- But still some IPv4-only boxes are on the Internet
- Similar idea as NAT-PT, but working correctly
- Optional element, but decoupled, DNS64
- Good solution if IPv4 is not required at the client
  - Client is IPv6-only
- Some apps don't work (Skype …)
  - Peer-to-peer using IPv4 "references"
  - Literal addresses
  - Socket APIs

NAT64 (2)

- Stateful NAT64 is a mechanism for translating IPv6 packets to IPv4 packets and vice-versa
  - The translation is done by translating the packet headers according to the IP/ICMP Translation Algorithm.
  - The IPv4 addresses of IPv4 hosts are algorithmically translated to and from IPv6 addresses by using a specific algorithm.
  - The current specification only defines how stateful NAT64 translates unicast packets carrying TCP, UDP and ICMP traffic.
  - DNS64 is a mechanism for synthesizing AAAA resource records (RR) from A RR. The IPv6 address contained in the synthetic AAAA RR is algorithmically generated from the IPv4 address and the IPv6 prefix assigned to a NAT64 device
- NAT64 allows multiple IPv6-only nodes to share an IPv4 address to access the IPv4 Internet

NAT64 (3)


Internet
IPv4
Internet
IPv6
Public IPv4
NAT64
NAT64
NAT64
ISP network
"plain" IPv6
IPv6-only
access
CPE
CPE
AAAA
synthesis
v4
v4
v6
v4/v6
10.0.0.x/24

NAT64 breaks …

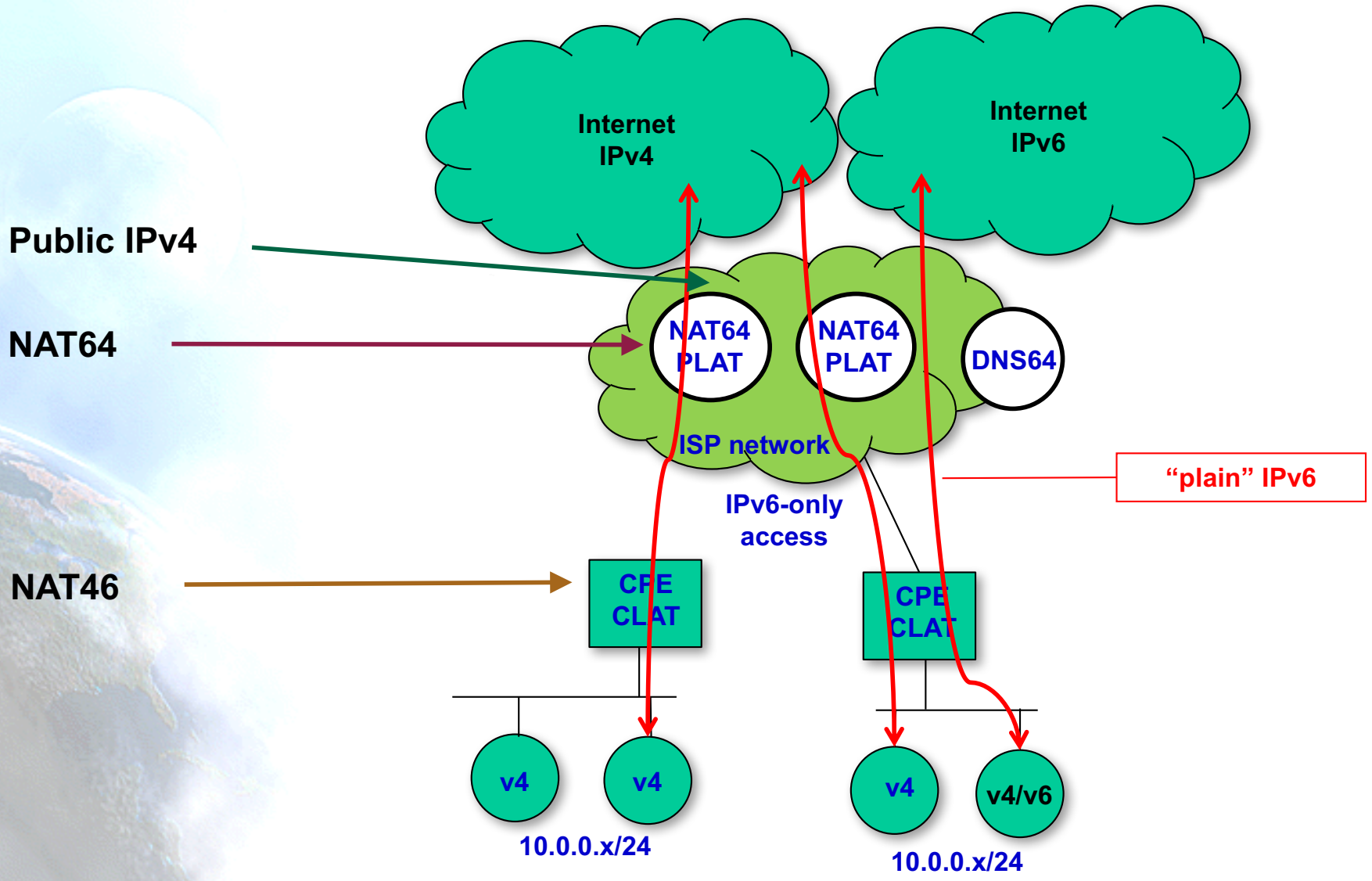
App Name	Functionality	Version	464XLAT Fixed
connection tracker	Broken	NA	NA
DoubleTwist	Broken	1.6.3	YES
Go SMS Pro	Broken	NA	YES
Google Talk	Broken	4.1.2	YES
Google+	Broken	3.3.1	YES
IP Track	Broken	NA	NA
Last.fm	Broken	NA	YES
Netflix	Broken	NA	YES
ooVoo	Broken	NA	YES
Pirates of the Caribean	Broken	NA	YES
Scrabble Free	Broken	1.12.57	YES
Skype	Broken	3.2.0.6673	YES
Spotify	Broken	NA	YES
Tango	Broken	NA	YES
Texas Poker	Broken	NA	YES
TiKL	Broken	2.7	YES
Tiny Towers	Broken	NA	YES
Trillian	Broken	NA	YES
TurboxTax Taxcaster	Broken	NA	
Voxer Walkie Talkie	Broken	NA	YES
Watch ESPN	Broken	1.3.1	
Zynga Poker	Broken	NA	YES
Xabber XMPP	Broken	NA	

***T-Mobile**

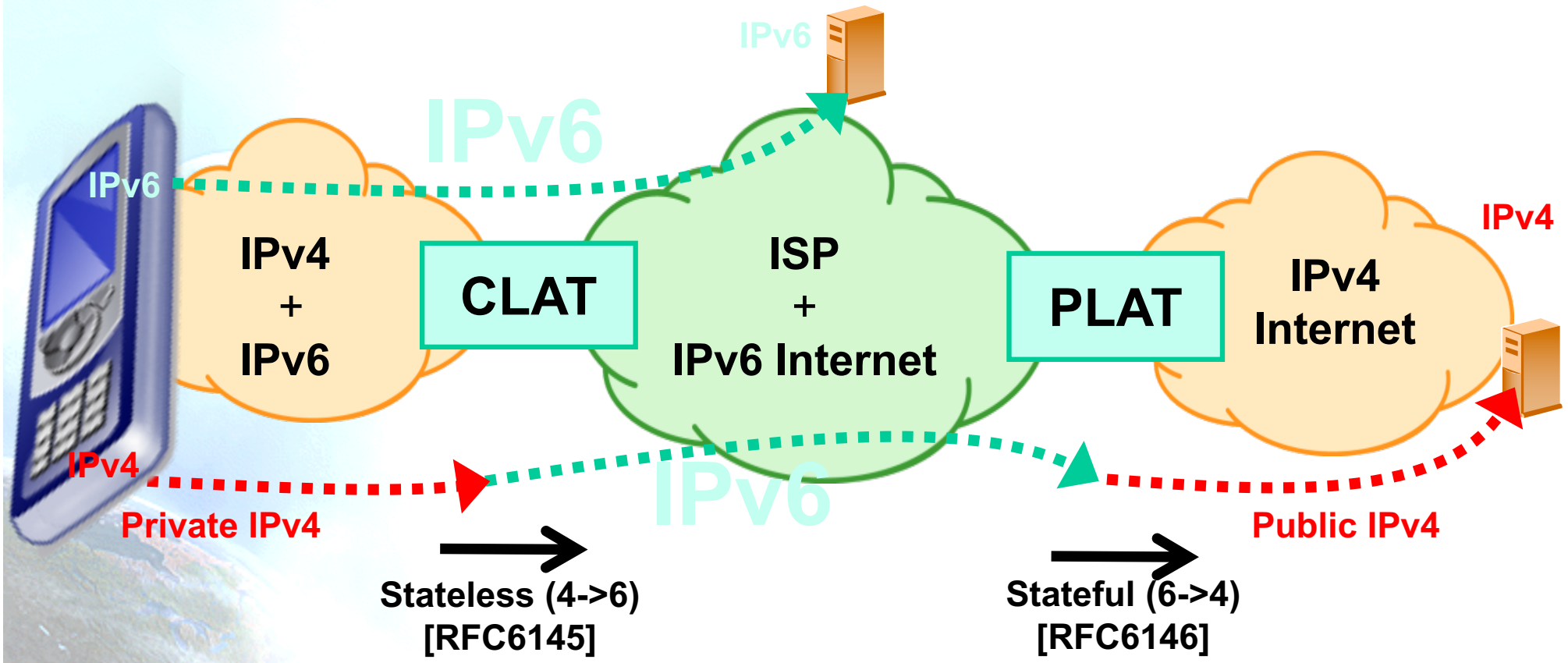
464XLAT

- 464XLAT (RFC6877): RFC6145 + RFC6146
- Very efficient use of scarce IPv4 resources
  - N*65.535 flows per each IPv4 address
  - Network growth not tied to IPv4 availability
- IPv4 basic service to customers over an-IPv6 only infrastructure
  - **WORKS** with applications that use socket APIs and literal IPv4 addresses (Skype, etc.)
- Allows traffic engineering
  - Without deep packet inspection
- Easy to deploy and available
  - Commercial solutions and open source

464XLAT


Internet
IPv4
Internet
IPv6
Public IPv4
NAT64
NAT64
PLAT
NAT64
PLAT
DNS64
ISP network
"plain" IPv6
IPv6-only
access
NAT46
CPE
CLAT
CPE
CLAT
v4
v4
v4
v4/v6
10.0.0.x/24
10.0.0.x/24

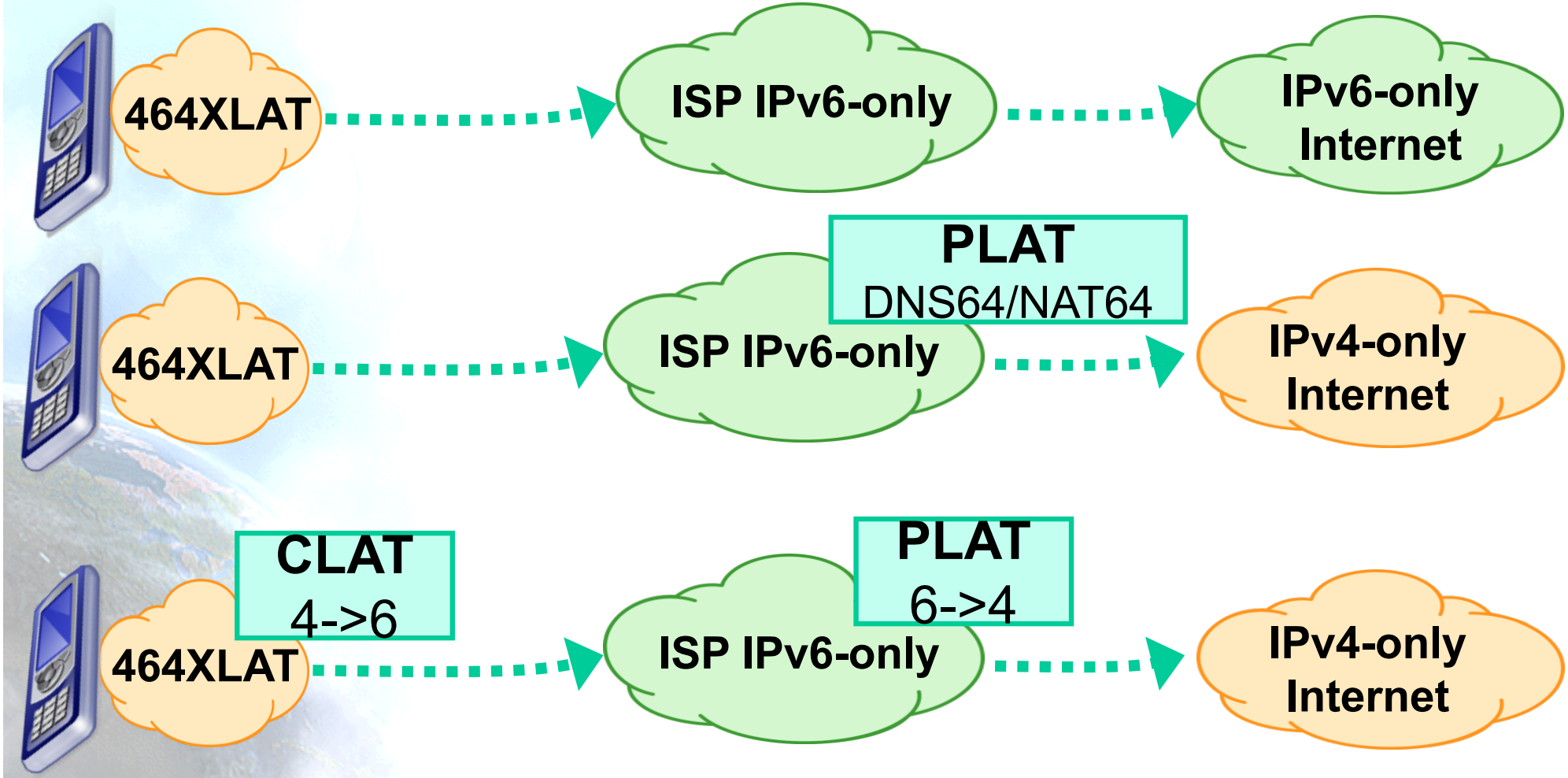
How it works 464XLAT?



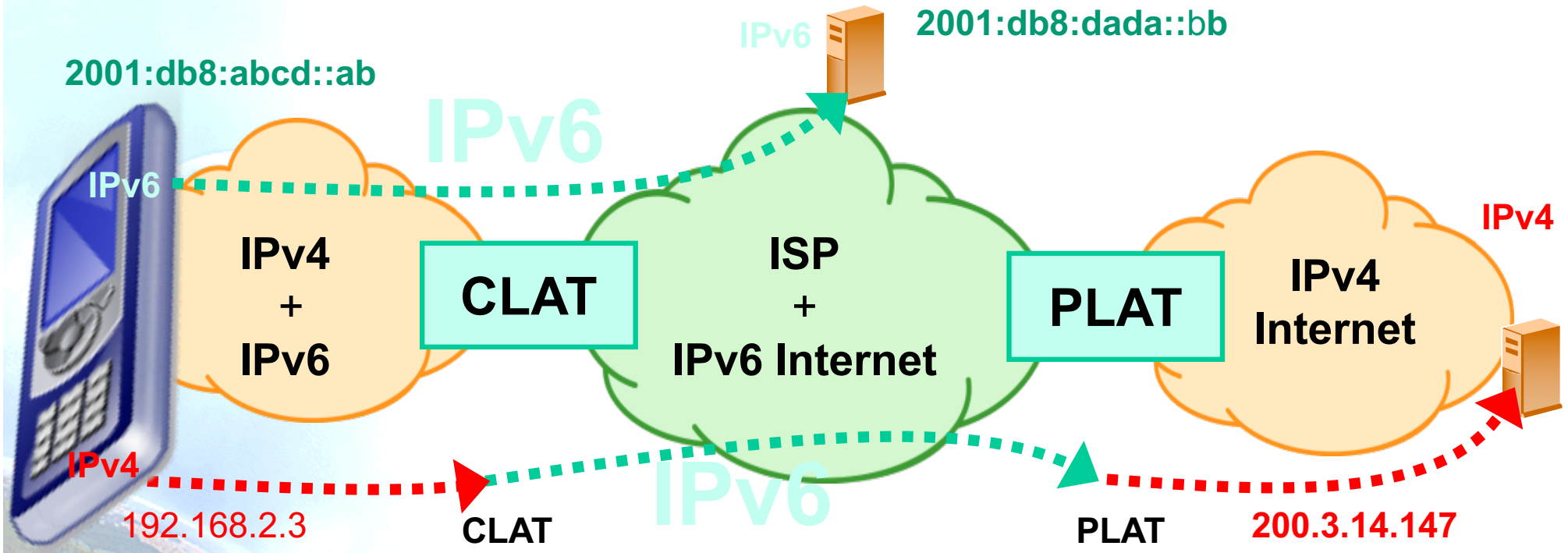
CLAT: Customer side translator (XLAT)

PLAT: Provider side translator (XLAT)

Possible “app” cases



464XLAT Addressing



CLAT
XLATE SRC prefix
[2001:db8:abcd::/96]
XLATE DST prefix
[2001:db8:1234::/96]

PLAT
IPv4 pool
(192.1.0.1 – 192.1.0.250)
XLATE DST prefix
[2001:db8:1234::/96]

IPv4 SRC
192.168.2.3
IPv4 DST
200.3.14.147

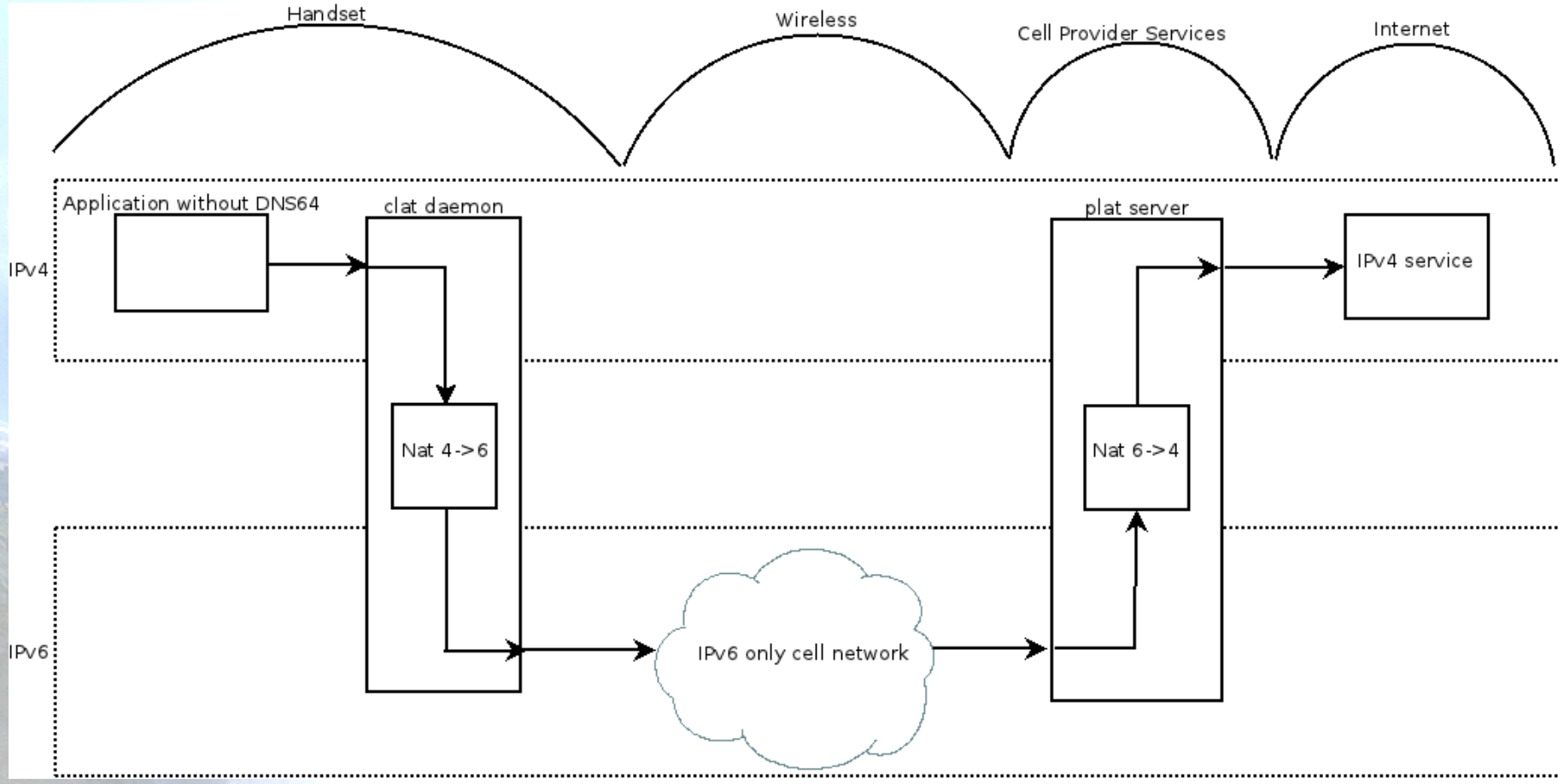
→ **Stateless** XLATE **[RFC6145]**

IPv6 SRC
2001:db8:abcd::192.168.2.3
IPv6 DST
2001:db8:1234::200.3.14.147

→ **Stateful XLATE [RFC6146]**

IPv4 SRC
192.1.0.1
IPv4 DST
200.3.14.147

Simplicity



* Dan Drown

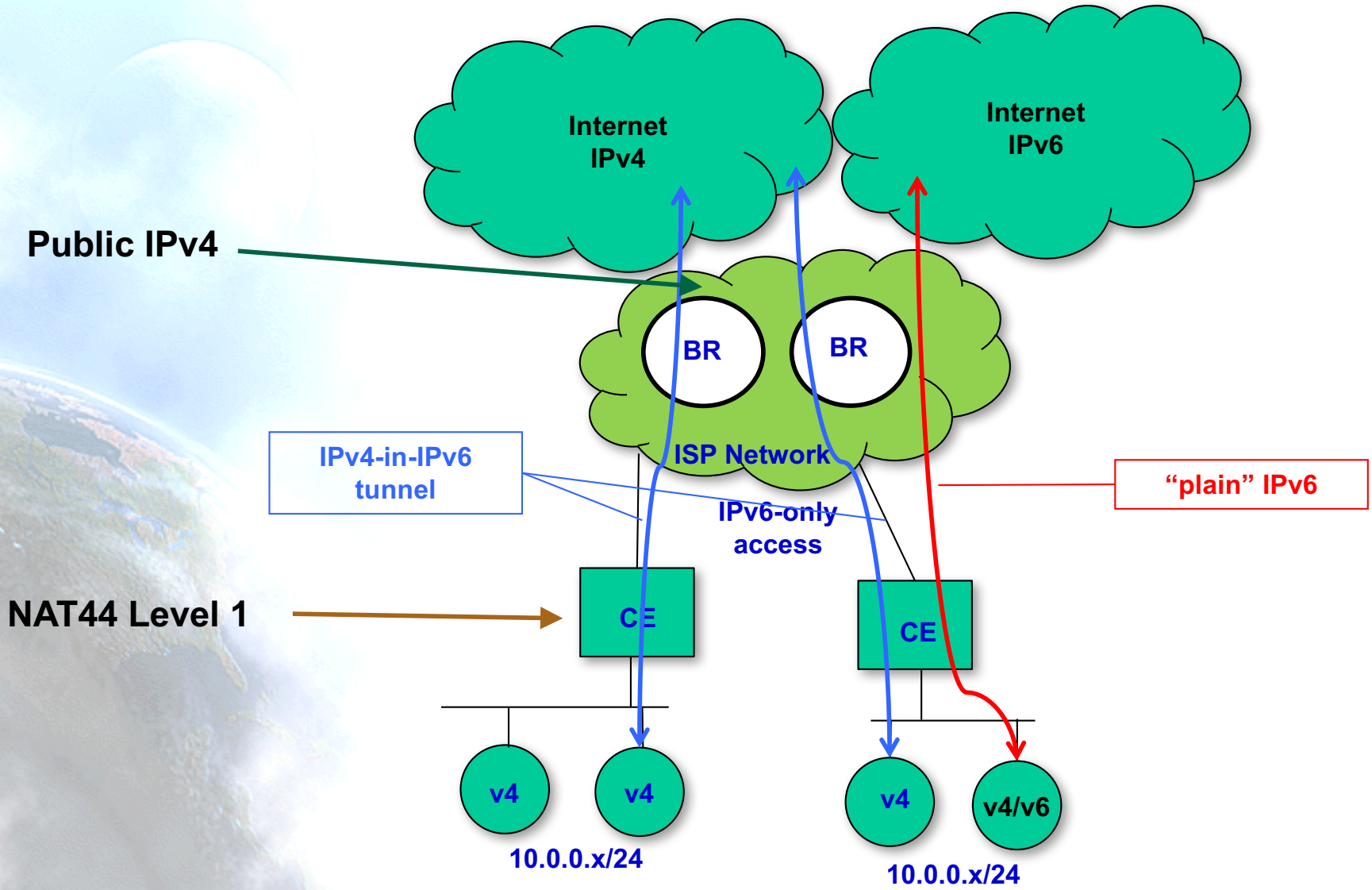
Availability and Deployment

- NAT64:
  - A10
  - Cisco
  - F5
  - Juniper
  - NEC
  - Huawei
  - Jool, Tayga, Ecdsys, Linux, OpenBSD, …
- CLAT
  - Android (since 4.3)
  - Nokia
  - Windows
  - NEC
  - Linux
  - Jool
  - OpenWRT
  - Apple (sort-of, is Bump-in-the-Host [RFC6535] implemented in Happy Eyeballs v2) - IPv6-only since iOS 10.2
- Commercial deployments:
  - T-Mobile US: +68 Millions of users
  - Orange
  - Telstra
  - SK Telecom
  - …
  - Big trials in several ISPs

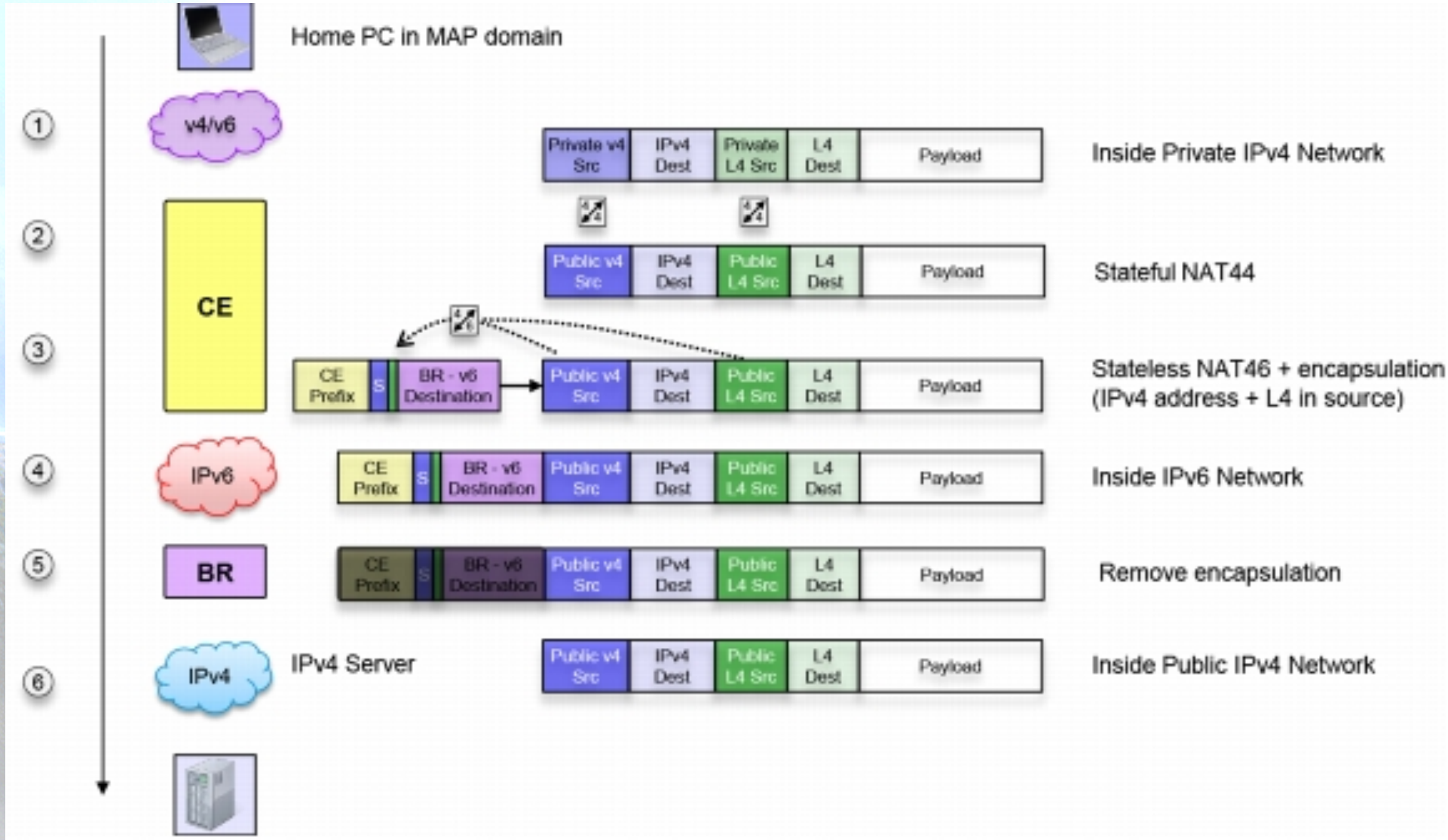
MAP Encapsulation (MAP-E)

- Mapping of Address and Port with Encapsulation
- Is a “stateless” DS-Lite
  - Provision of an IPv4 prefix, address or “shared” address
  - Algorithmic mapping between IPv4 and an IPv6 address
  - Extends CIDR to 48 bits (32 IP + 16 port)
- Allows encapsulating IPv4 in IPv6 for both mesh and hubs&spoke topologies, including mapping-independent IPv4 and IPv6
- Two elements:
  - MAP Customer Edge (CE)
  - MAP Border Relay (BR)

MAP-E


Internet
IPv4
Internet
IPv6
Public IPv4
BR
BR
ISP Network
IPv4-in-IPv6
tunnel
IPv6-only
access
"plain" IPv6
NAT44 Level 1
CE
CE
v4
v4
v4
v4/v6
10.0.0.x/24
10.0.0.x/24

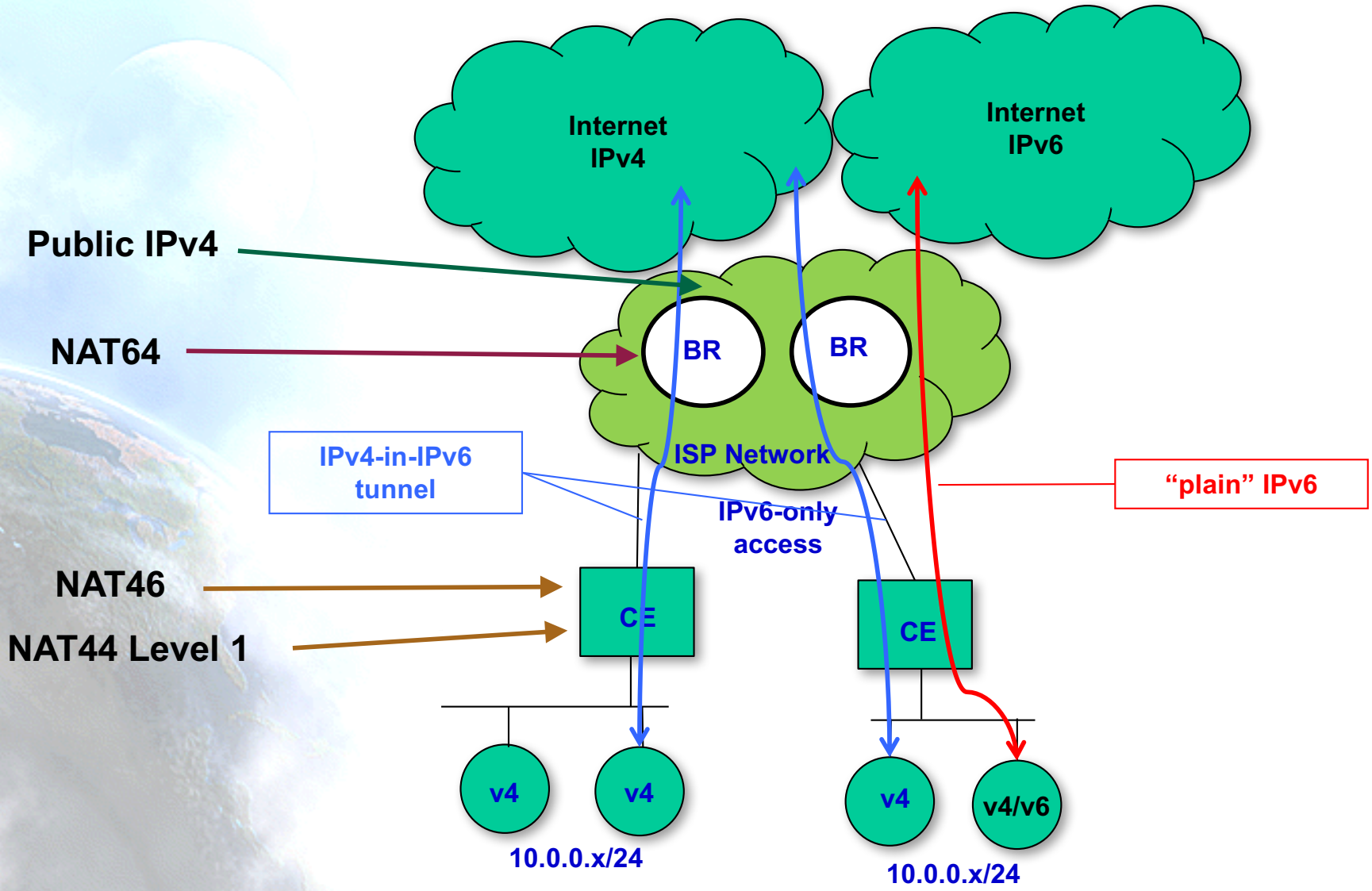
MAP-E Packet Path



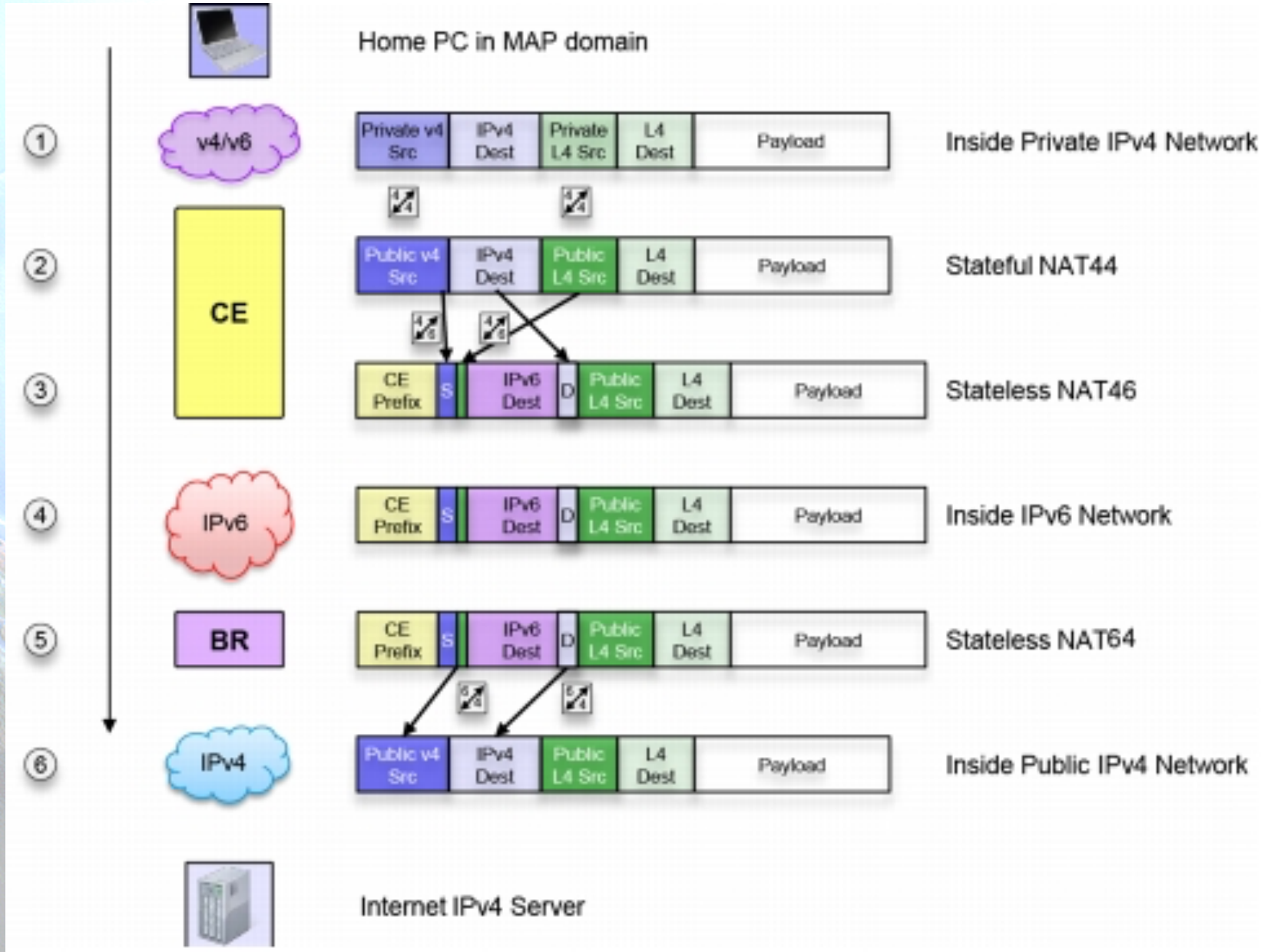
MAP Translation (MAP-T)

- Mapping of Address and Port using Translation
- Similar to MAP-E
- Similar to 464XLAT in the sense of the double translation NAT46 (CLAT) and NAT64 (PLAT)

MAP-T

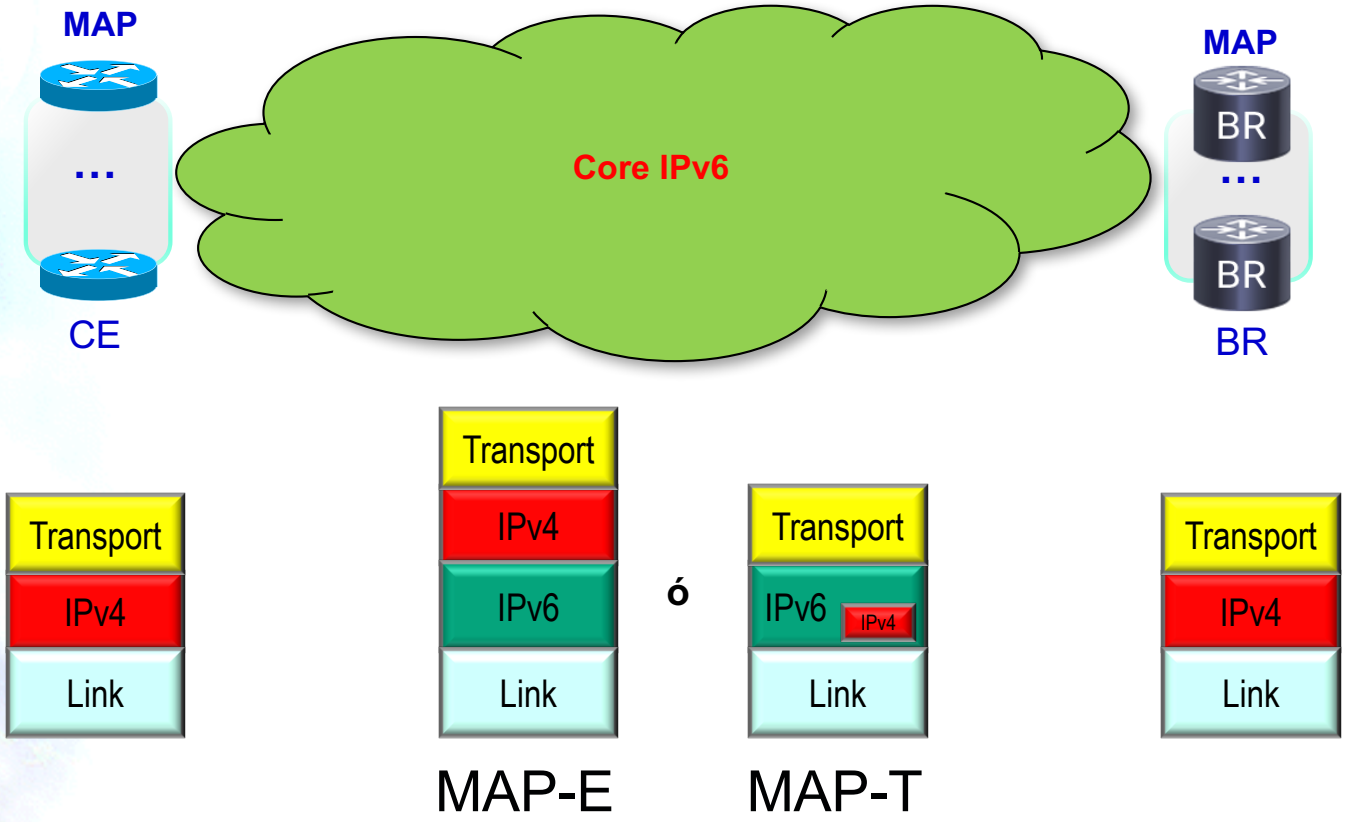

Internet
IPv4
Internet
IPv6
Public IPv4
NAT64
BR
BR
ISP Network
IPv4-in-IPv6
tunnel
IPv6-only
access
"plain" IPv6
NAT46
NAT44 Level 1
CE
CE
v4
v4
v4
v4/v6
10.0.0.x/24
10.0.0.x/24

MAP-T Packet Path

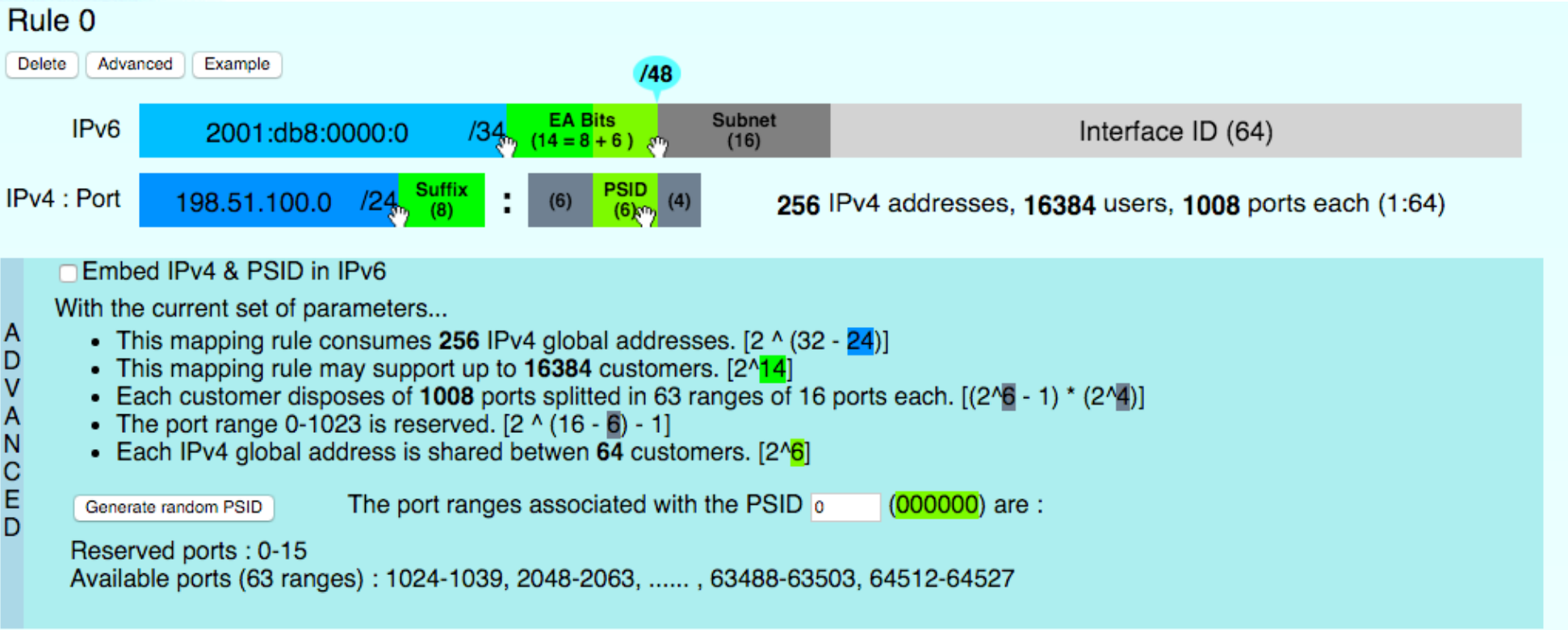


MAP-E vs MAP-T

- MAP-E uses extra 20 bytes for the encapsulation (IPv4-in-IPv6 tunnel).

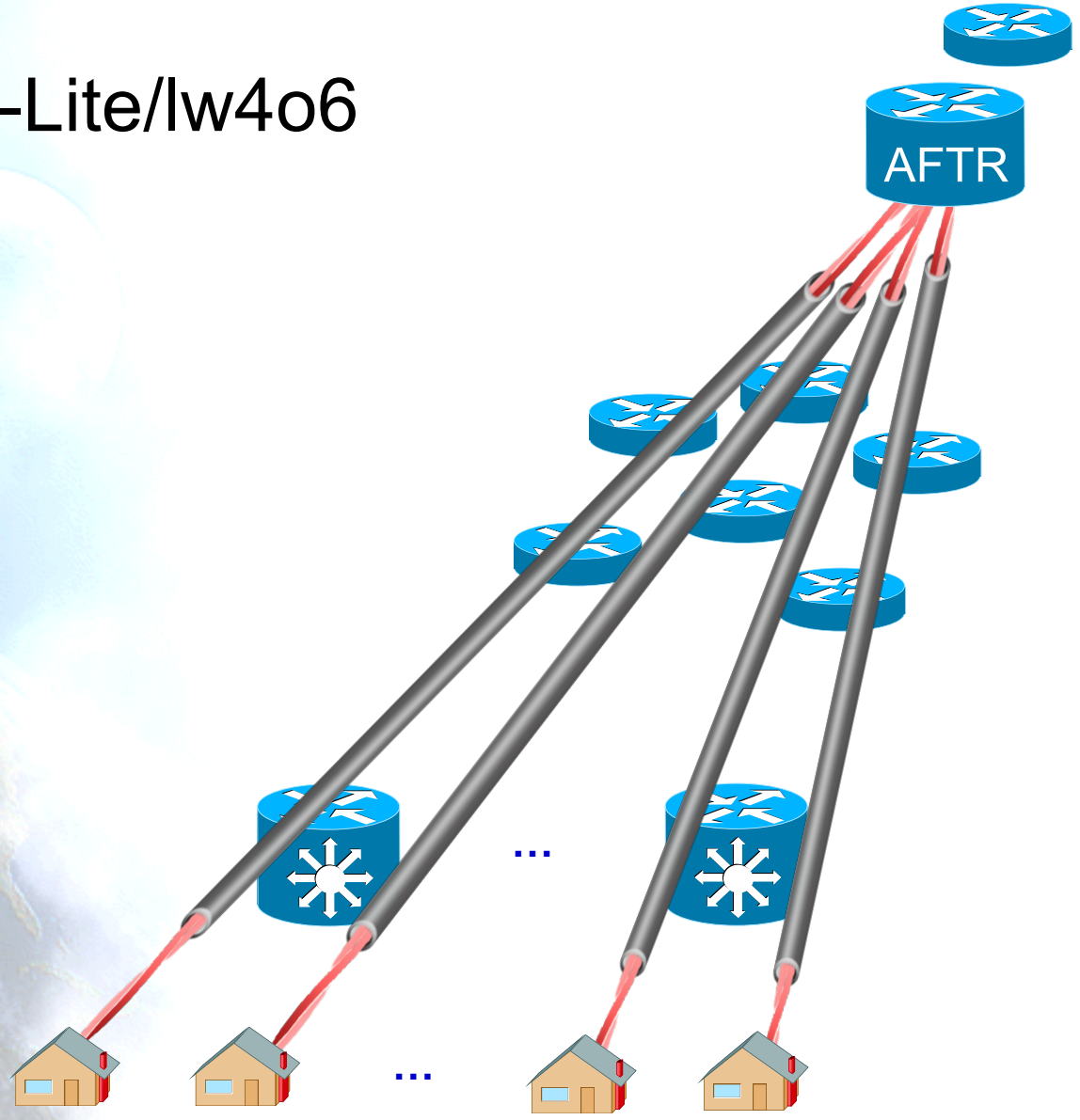


MAP Addressing



Tunnels per subscribers

- DS-Lite/lw4o6



BGP prefixes: Tens

Tunnels: Millions

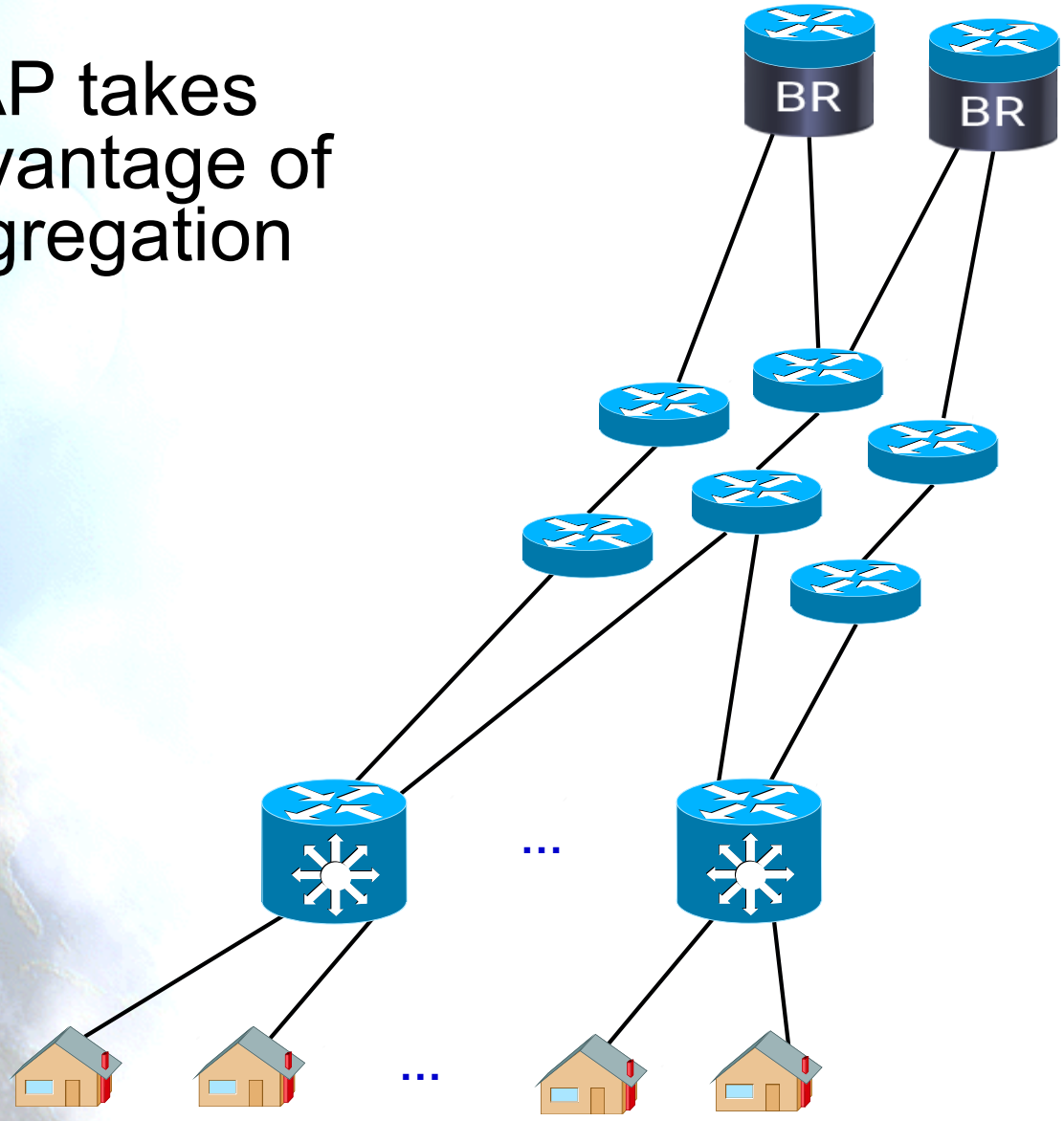
IGP prefixes: Hundreds

BNG routes: Thousands

Subscribers: Millions

IPv6 routing

- MAP takes advantage of aggregation



BGP prefixes: Tens

MAP rules: Tens
NO CGN

IGP prefixes: Hundreds

BNG routes: Thousands

Subscribers: Millions

DS-Lite vs MAP performance

Cisco ASR9K

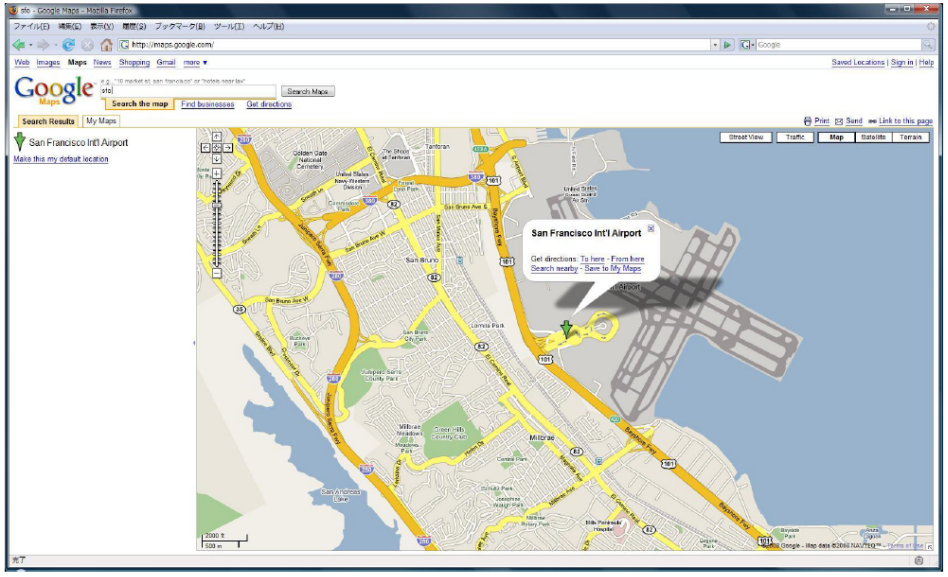
- DS-Lite routes traffic in the ISM Blade
  - 14 Gbps per slot
- MAP NO needs that
  - 240 Gbps per slot

Comparing …

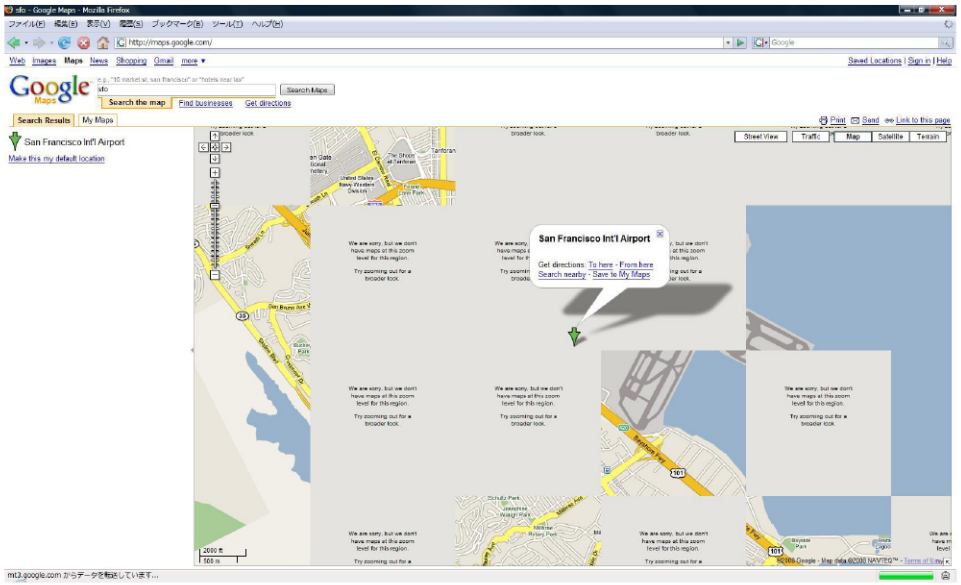
	6RD	Softwires v2	NAT444	DS-Lite	Lw4o6	NAT64	464XLAT	MAP-E	MAP-T
Tunel/Translation (X)	T 6in4	T 6in4	X	T 4in6	T 4in6	X	X	T 4in6	X
Dual-stack LAN	YES	YES	optional	YES	YES	YES	YES	YES	YES
IPv4 Multicast	YES	YES	YES	NO	NO	NO	NO	NO	NO
Access Network	IPv4	IPv4	IPv4 /dual	IPv6	IPv6	IPv6	IPv6	IPv6	IPv6
Overhead	20 bytes	40 bytes	-	40 bytes	40 bytes	20 bytes	20 bytes	40 bytes	20 bytes
Impact in IPv6 addressing plan	YES	NO	NO	NO	NO	NO	NO	YES	YES
CPE Update	YES	YES	optional	YES	YES	YES	YES	YES	YES
NAT44/NAPT	CPE	CPE	CPE + CGN	CGN	CPE	CPE	CPE	CPE	CPE
46/64 Translation	-	-	-	-	-	ISP	ISP +/or CPE	-	CPE + ISP
Translation at ISP with or w/o state	-	-	with	-	-	with	with	w/o	w/o
Scalability	High	Medium	Medium	Medium	High	High	High	High	High
Performance	High	Low	Low	Low	High	Medium	High	High	High
ALGs	NO	NO	YES	YES	NO	YES	YES	YES	YES
Any Protocol or only-TCP/UDP/ICMP	YES	YES	YES	YES	YES	NO	NO	NO	NO
Sharing IPv4 Ports	NO	NO	YES	YES	YES	NO	NO	YES	YES
IPv6 Aggregation	NO	NO	optional	YES	YES	YES	YES	YES	YES
IPv4 Mesh	YES	YES	YES	NO	NO	NO	NO	YES	YES
IPv6 Mesh	YES	NO	optional	YES	YES	YES	YES	YES	YES
Impacts on logging	NO	NO	YES	YES	NO	YES	YES	NO	NO
HA simplicity	High	Low	Low	Low	High	Medium	High	High	High
DPI simplicity	Low	Low	High	Low	Low	High	High	Low	High
Support in cellular	NO	NO	YES	NO	NO	YES	YES	NO	NO
Support in CPEs	YES	YES	YES	YES	YES	YES	YES	YES	YES
	15.5	12.5	10.5	9.5	15	12.5	14	13	13.5

How many ports per user?

Max 30 Connections



Max 15 Connections



- **Possibly a minimum of 300 per user behind each CPE**
  - **More as AJAX/similar technologies usage increase**
  - **Times average number of users behind each NAT**
  - **And going up**
- **Be aware of IP/port sharing implications …**

Update of RFC7084

- Basic Requirements for IPv6 Customer Edge Routers
  - Originally include support only for 6RD and DS-LITE
  - Being updated to include support for 464XLAT, MAP T/E, lw4o6, …
- https://tools.ietf.org/html/draft-ietf-v6ops-rfc7084-bis

Thanks !

Contact:

- Jordi Palet: jordi.palet@theipv6company.com