The IPv64 IPv4/IPv6 Transition Mechanism

Arturo Azcorra

University Carlos III of Madrid

March 2002

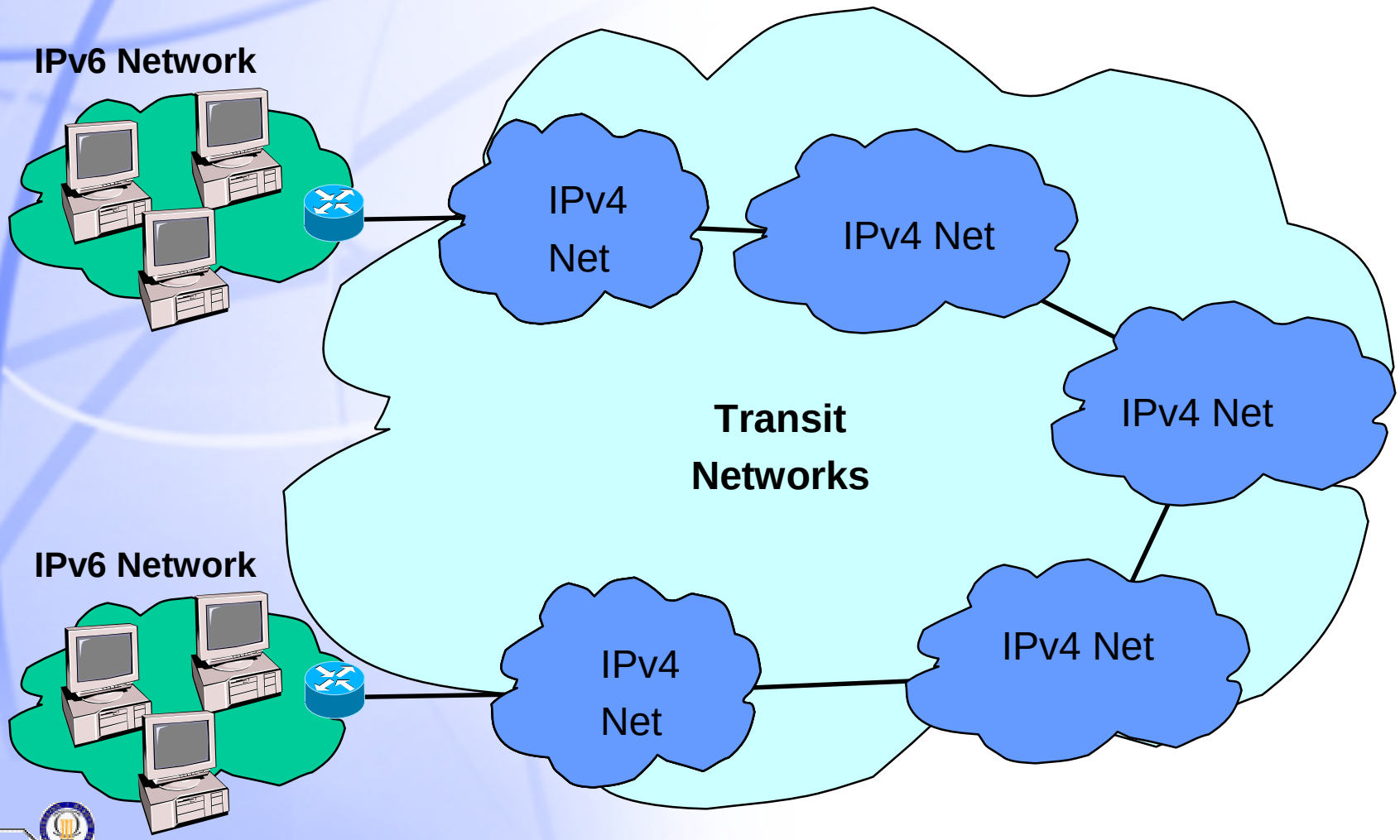
 http://www.it.uc3m.es

Contents

- Problem Discussion
- Description of IPv64
- Advantages of IPv64
- Conclusions

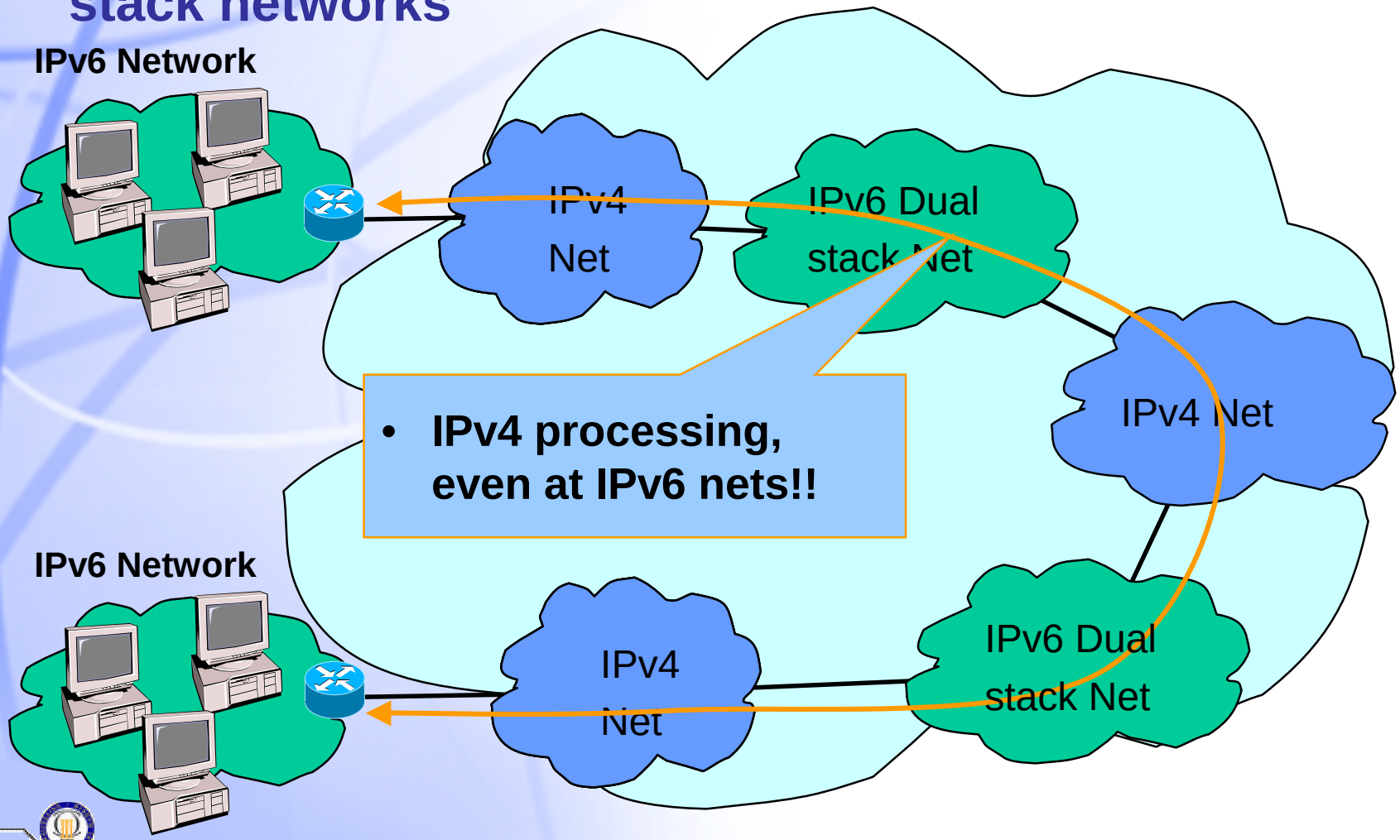
TODAY

- 6to4 is very suitable to interconnect IPv6 islands



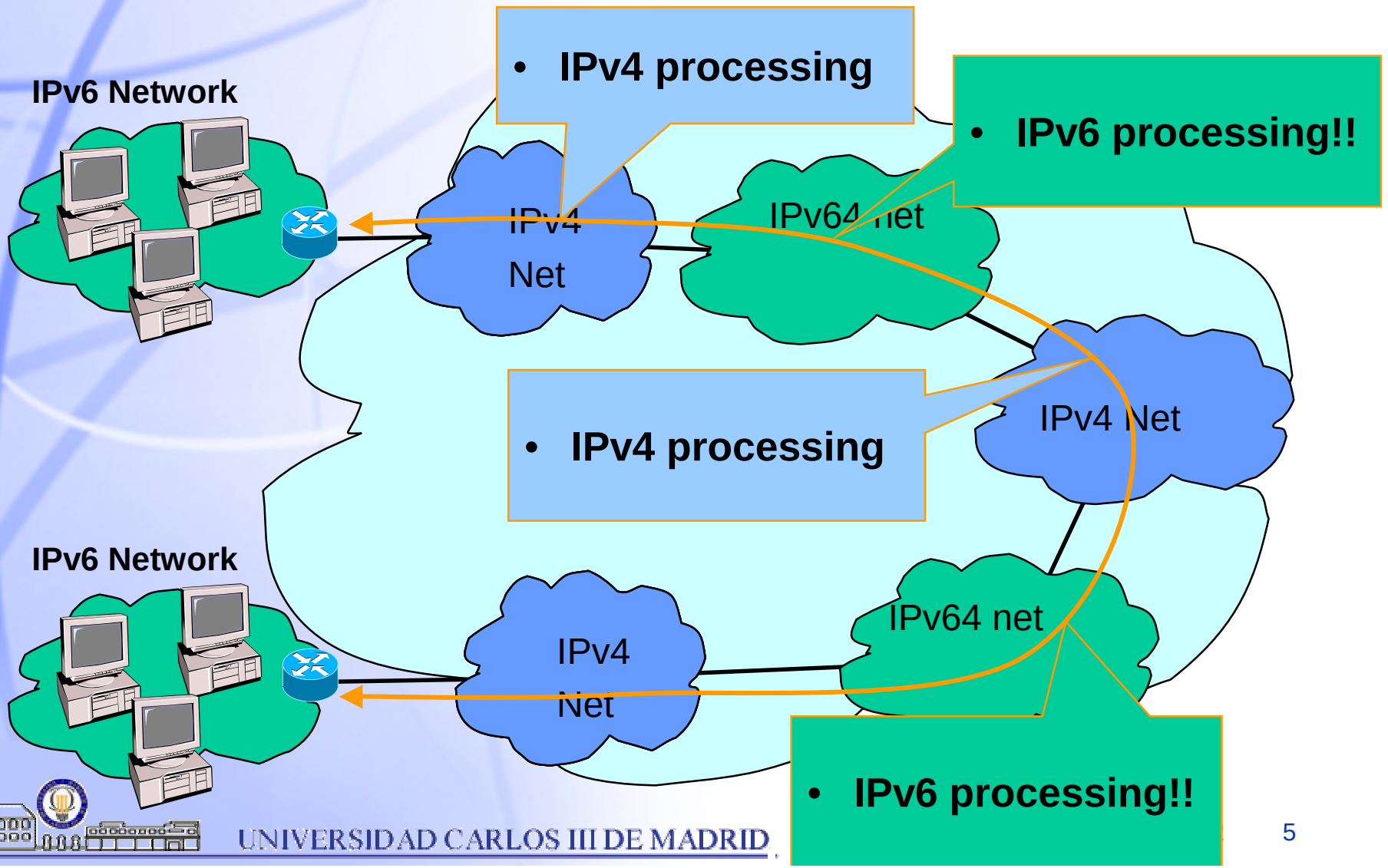
TOMORROW!!

- Always IPv4 processing in transit, even in dual stack networks



Advantage of IPv64

- Process as IPv6 in transit, at every IPv64 network



Description of IPv64

- Problem Discussion
- Description of IPv64
- Advantages of IPv64
- Conclusions

Design Principles of IPv64

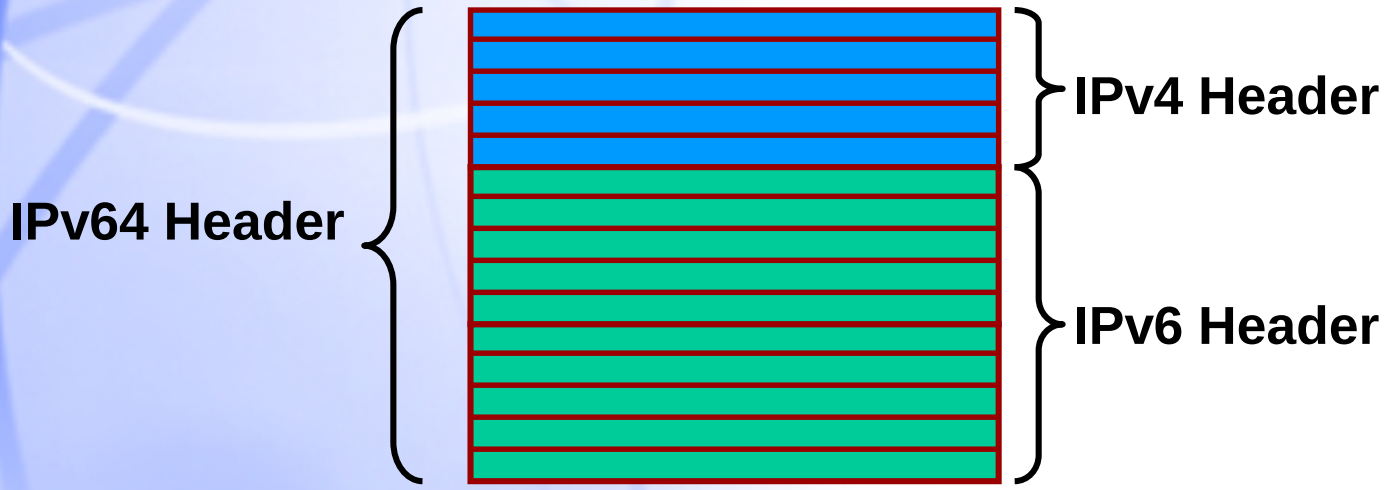
- Make advantage of the complexity of dual-stack systems
- Maintain compatibility with other transition mechanisms
- Stimulate the migration of transit networks to IPv6 by obtaining immediate benefits from migration

IPv64 basics

- End-to-end usage of double header: [IPv4 + IPv6]
- Concept of “Sufficiently close” IPv4 address
- Processing as IPv6 at every IPv64 router
- Processing as IPv4 at every IPv4-only router
- Simplified transit through IPv6-only networks

Format of IPv64 Packets

- Double header: IPv4 header followed by IPv6 header
- It is NOT plain tunneling, as the IPv6 header remains vissible to IPv64 routers

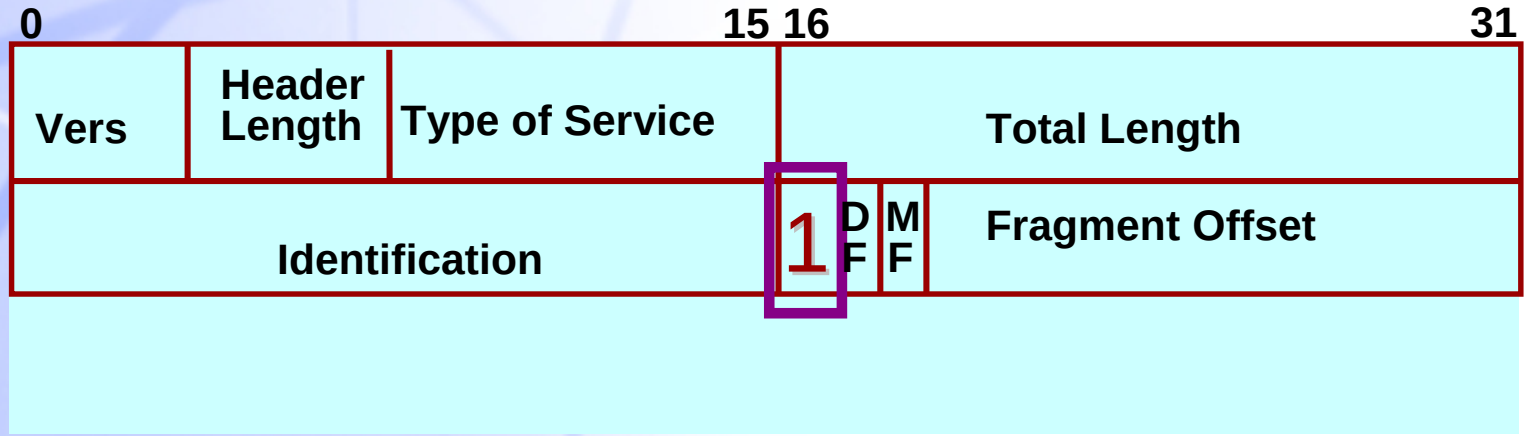


Creating IPv64 packets

- Create IPv4 sub-header of the IPv64 packet
- Obtaining the sufficiently close IPv4 source and destination addresses
- An IPv4 address beyond which no IPv4 router will be transversed until to the IPv6 destination
- The IPv4 sufficiently close source address is known by context to the source system
- The IPv4 sufficiently close destination address is obtained from the IPv6 destination address

Identifying IPv64 packets

- Bit 16 of the second word of the IPv4 header
  - RFC791 => Bit unused: should be set to 0 by hosts and forwarded unchanged by routers
  - Set to 1 => IPv64 Packet!



Processing IPv64 packets

- IPv4-only routers process IPv64 packets as conventional IPv4 packets
- IPv64-enabled routers
  - Detect that it is IPv64 packet
  - Conventional Processing of IPv6 header:
    - Forwarding, Hop Limit and
    - Options, diffserv, flow label, extension headers, ...
  - Adapt outgoing IPv4 header

Cost of IPv64

- Implementation complexity: minor over that of a dual-stack system
- Processing overhead: negligible at transit routers
- Processing overhead at hosts or local routers:
  - Obtaining the “Sufficiently close” IPv4 address
  - Only required once at each side of the end-to-end transit
  - Same requirements of tunnelling approaches
- Transmission overhead: IPv4 added header (same as tunnelling approaches)
- Minor restrictions (max. length, do not fragment, …)

Advantages of IPv64

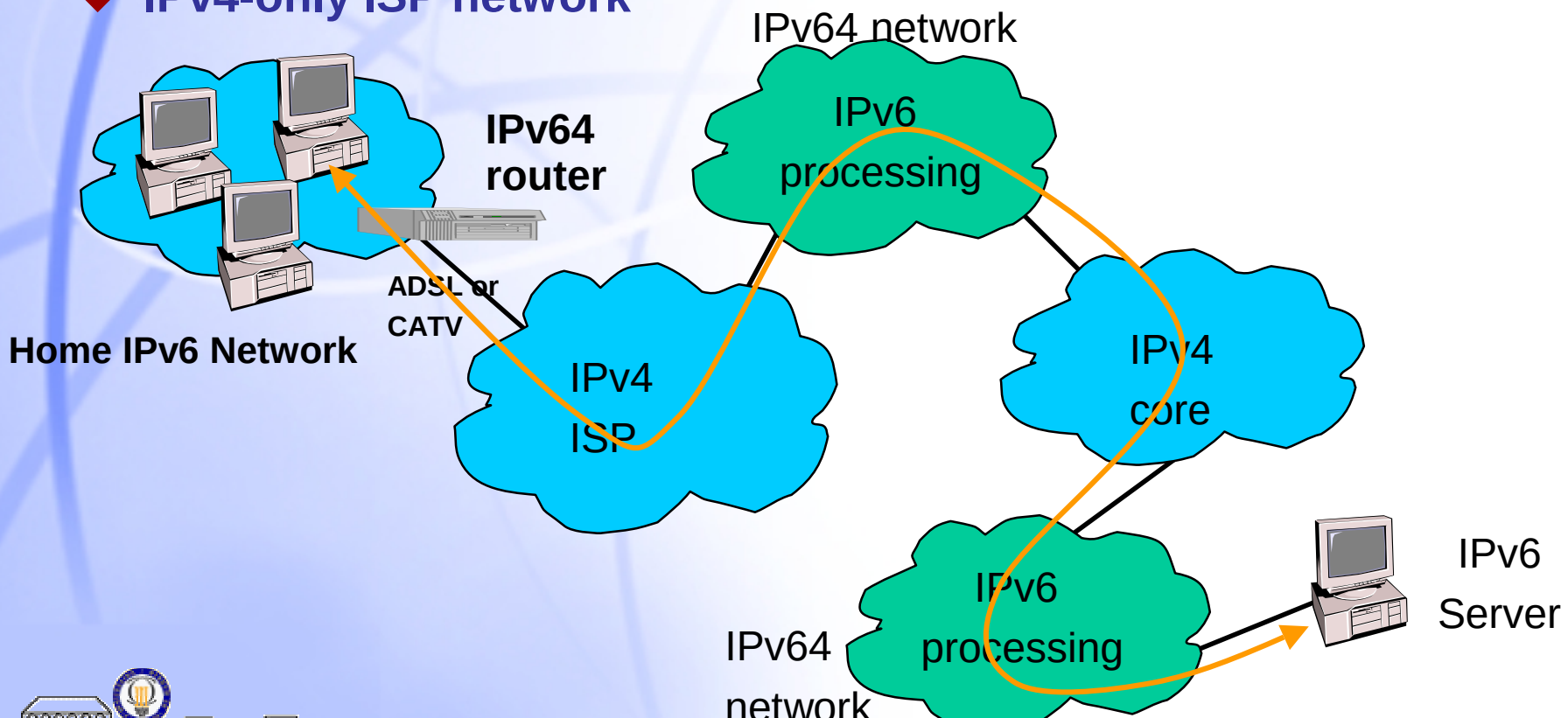
- Problem Discussion
- Description of IPv64
- **Advantages of IPv64**
- Conclusions

Advantages of IPv64

- Packet processing as IPv6 at every transit IPv64 network
- Packet processing as IPv4 at every transit IPv4 network
- Compatible with other transition mechanisms

Case Study: ADSL permanent addresses

- ADSL users provided with several permanent IPv6 addresses at their home network
- IPv4-only ISP network



Conclusions

- Problem Discussion
- Description of IPv64
- Advantages of IPv64
- **Conclusions**

Conclusions (I)

- Does IPv64 work?

- YES: available prototype of host and router
- YES: experiments have been performed

- Software is available for download at:
  matrix.uc3m.es/~ipv64

Conclusions (II)

- **Is IPv64 worth deploying?**
- **IPv64 allows processing packets as IPv6 at every IPv64 router**
- **Because of this, it stimulates the migration of transit networks to IPv64, as the benefits of IPv6 packet processing are obtained for all native IPv6 and all IPv64 transition traffic**
- **IPv64 allows transparent transit through IPv4 networks because IPv64 packets are normally processed as IPv4**
- **IPv64 is complementary and compatible to other transition approaches**
- **Implementation complexity, processing overhead and transmission overhead are low (over that of a dual-stack system and tunneling approaches)**

Bibliography

- "Internet Protocol Version 64 (IPv64) Specification". A. Azcorra, A. Garcia and M. Bagnulo. draft-azcorra-ipv64-04.txt. March 2002.
- Related bibliography
  - RFC 2893. "Transition Mechanisms for IPv6 Hosts and Routers". R. Gilligan, E. Nodmark. Agosto 2000.
  - RFC 3056. "Connection of IPv6 Domains via IPv4 Clouds". B. Carpenter, K Moore. Febrero 2001.
  - RFC 2765. "Stateless IP/ICMP Translation Algorithm (SIIT)". E. Nodmark. Febrero 2000.
  - RFC 2766. "Network Address Translation - Protocol Translation (NAT-PT)". G. Tsirtsis, P. Srisuresh. Febrero 2000.
  - "An overview of the introduction of IPv6 in the Internet". W. Biemolt et al. Work in progress (<draft-ietf-ngtrans-introduction-to-ipv6-transition-07.txt>). Julio 2001.

Processing IPv64 packets

- IPv4-only routers process IPv64 packets as conventional IPv4 packets
- IPv64-enabled routers
  - Detect that it is IPv64 packet
  - Conventional Processing of IPv6 header:
    - Forwarding, Hop Limit and
    - Options, diffserv, flow label, extension headers, ...
  - Adapt outgoing IPv4 header. Might require to adjust:
    - IPv4 total length (only if IPv6 packet size changed)
    - IPv4 DSCP (only if it is an edge router)
    - IPv4 address (only if NAT used)
    - Checksum (only if any other change occurred)

Sufficiently Close IPv4 Addresses

- Definition: an IPv4 address beyond which no IPv4 router will be transversed until destination
- Placed in the IPv4 header of the IPv64 packet
- The IPv4 source address must be sufficiently close to the IPv6 source address
- The IPv4 destination address must be sufficiently close to the destination IPv6 address
- The IPv4 sufficiently close source address is known by context to the source system
- The IPv4 sufficiently close destination address must be obtained from the IPv6 destination address

Resolving the destination Sufficiently Close IPv4 Address

- Strategies for obtaining the destination sufficiently close IPv4 address from the IPv6 destination address
- Only required once, when building the IPv64 packet at host or local router
- Similar requirement to that of other approaches
- Several complementary mechanisms are used:
  - Configured table (useful for some particular cases)
  - Backward learning from source IPv4 address
    Most useful for information servers
  - Embedding IPv4 address on IPv6 address (as done in 6to4 and other approaches)
  - DNS for a domain: Domain =>
    IPv6address/prefix -> IPv4 address (~ to MX registers)
  - Cache table using IPv6 prefixes:
    IPv6address/prefix -> IPv4 address