



Este proyecto ha sido
cofinanciado por PROFIT



Porte de aplicaciones y servicios a IPv6

Eva M. Castro

eva@gsync.escet.urjc.es

Grupo de Sistemas y Comunicaciones (GSyC)

Universidad Rey Juan Carlos (URJC)

- **Arquitectura de transición**
 - Red
 - Nodos finales
 - Aplicaciones
- Evolución de aplicaciones
- Escenarios de transición de aplicaciones
- Dependencias en el código fuente
- Conclusiones

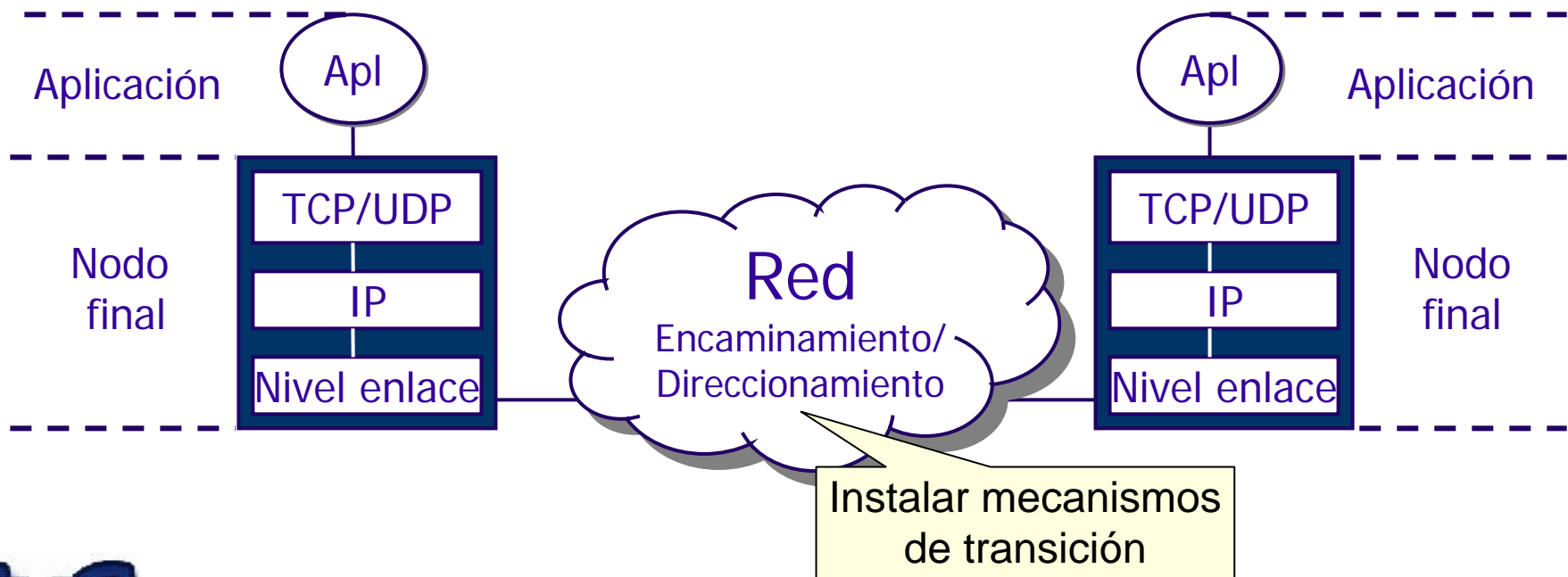
Arquitectura de transición

- Red
- Nodos finales
- Aplicaciones



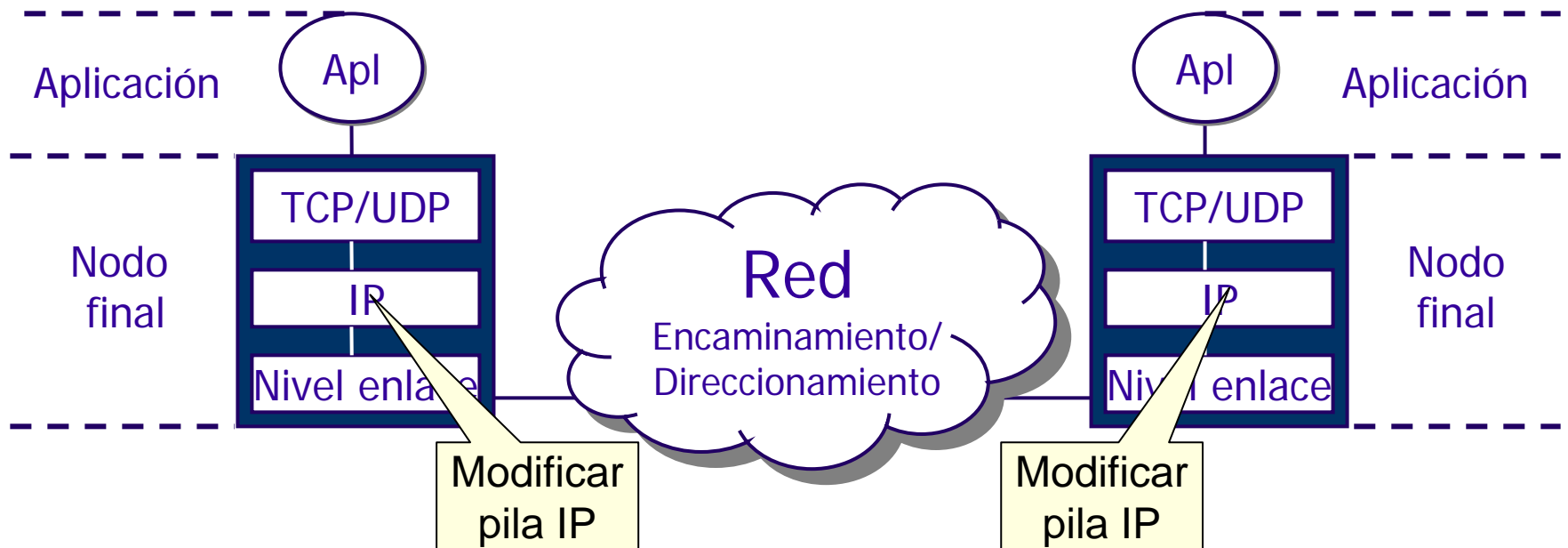
Arquitectura de transición

- Red
 - Red sólo IPv4
 - Red sólo IPv6
 - Red dual
 - Red heterogénea
- Nodos finales
- Aplicaciones



Arquitectura de transición

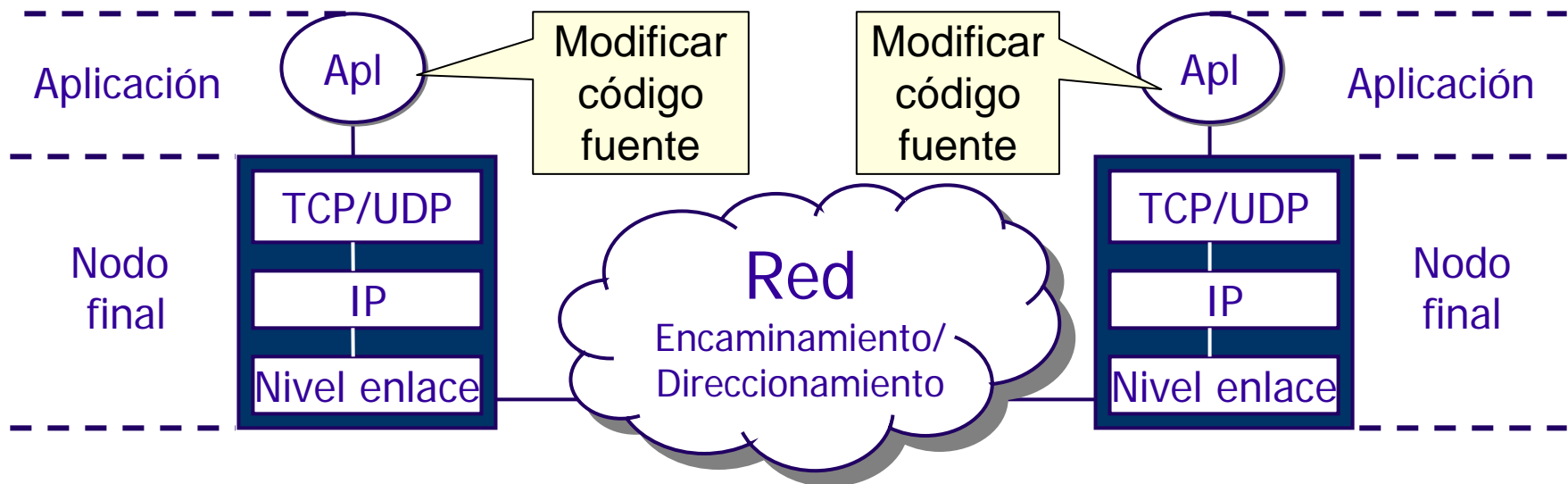
- Red
- Nodos finales
 - Nodo sólo IPv4
 - Nodo sólo IPv6
 - Nodo con doble pila
- Aplicaciones



Arquitectura de transición

- Red
- Nodos finales
- Aplicaciones

- Aplicación sólo IPv4
- Aplicación sólo IPv6
- Aplicación dual

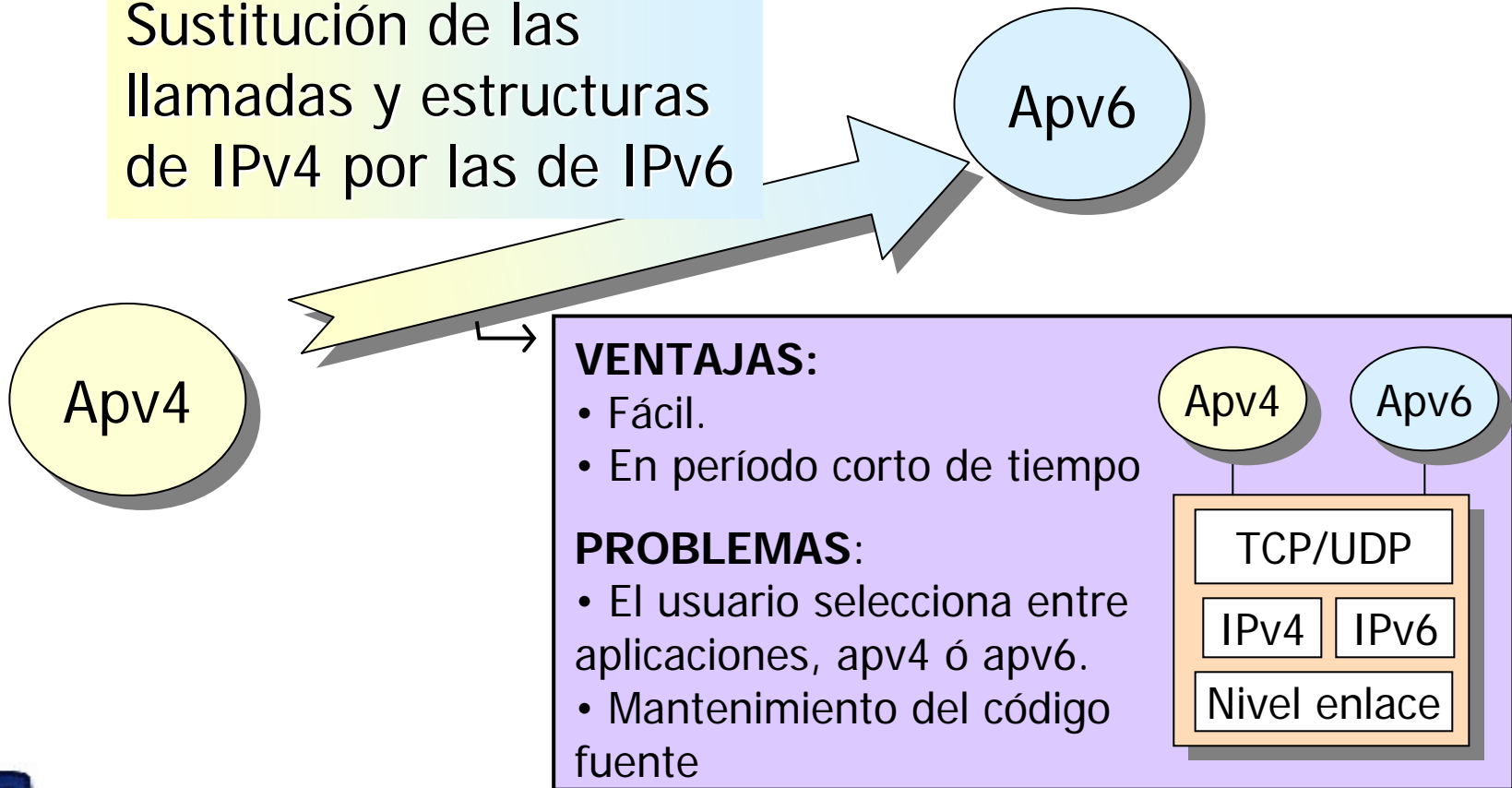


Índice

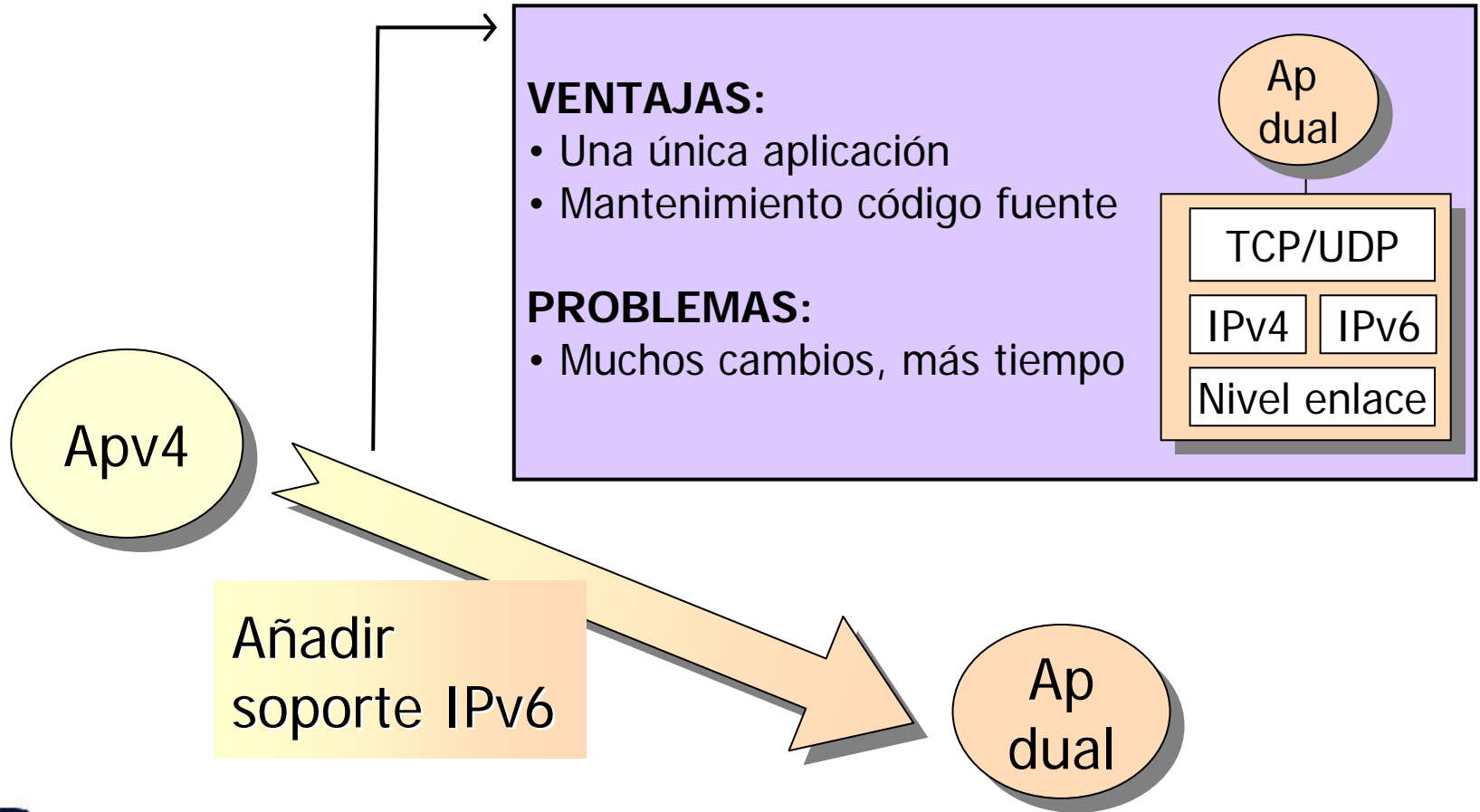
- Arquitectura de transición
- **Evolución de aplicaciones**
 1. De aplicaciones IPv4 a aplicaciones IPv6
 2. De aplicaciones IPv4 a aplicaciones duales
 3. Transición gradual
- Escenarios de transición de aplicaciones
- Dependencias en el código fuente
- Conclusiones

1. De aplicaciones v4 a aplicaciones IPv6

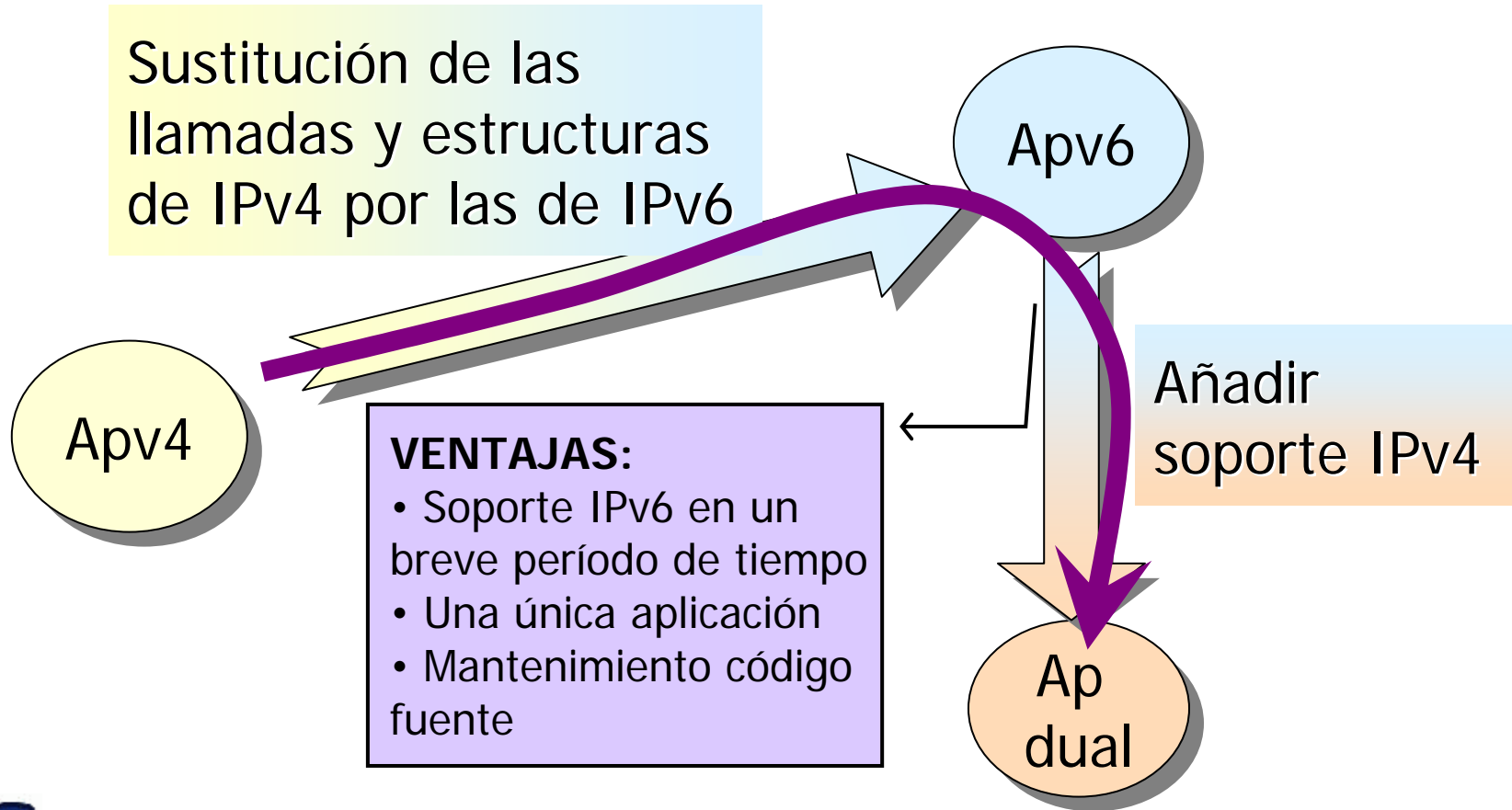
Sustitución de las llamadas y estructuras de IPv4 por las de IPv6



2. De aplicaciones IPv4 a aplicaciones duales



3. Transición gradual



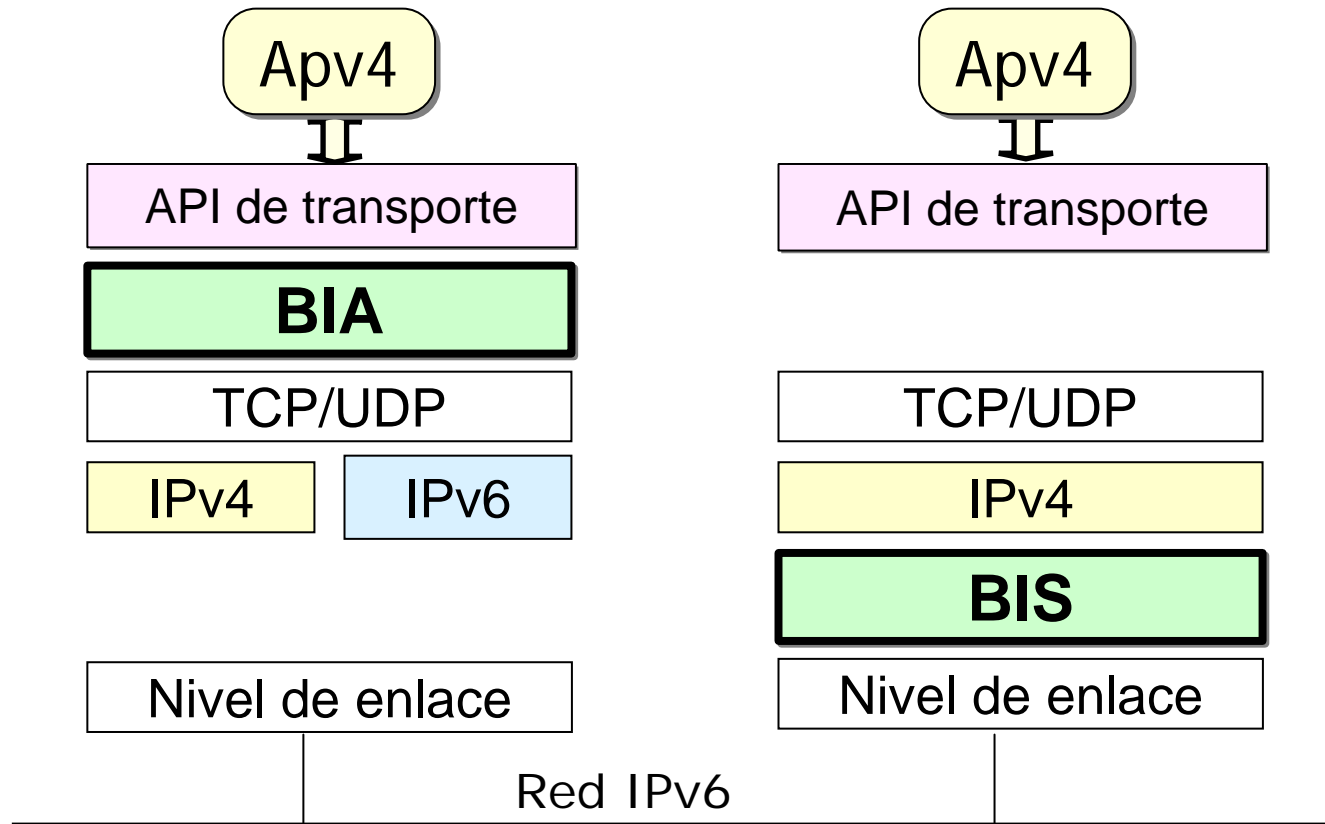
Índice

- Arquitectura de transición
- Evolución de aplicaciones
- **Escenarios de transición de aplicaciones**
 1. Aplicaciones IPv4 en nodos duales
 2. Aplicaciones IPv6 en nodos duales
 3. Aplicaciones duales en nodos duales
 4. Aplicaciones duales en nodos sólo IPv4
- Dependencias en el código fuente
- Conclusiones

1. Aplicaciones IPv4 en nodos duales

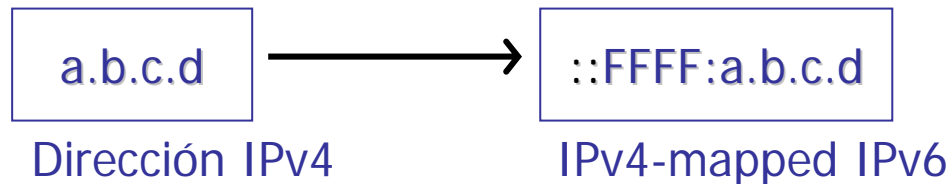
- Dependencias en el código fuente con IPv4.
- Intercambian paquetes IPv4.
- Para su funcionamiento en redes IPv6:
 - Portar el código a IPv6
 - Si no es posible, utilizar mecanismos de transición. Las aplicaciones utilizan IPv4 pero se intercambian paquetes IPv6:
 - BIA (Bump In the API)
 - BIS (Bump In the Stack)

BIS/BIA

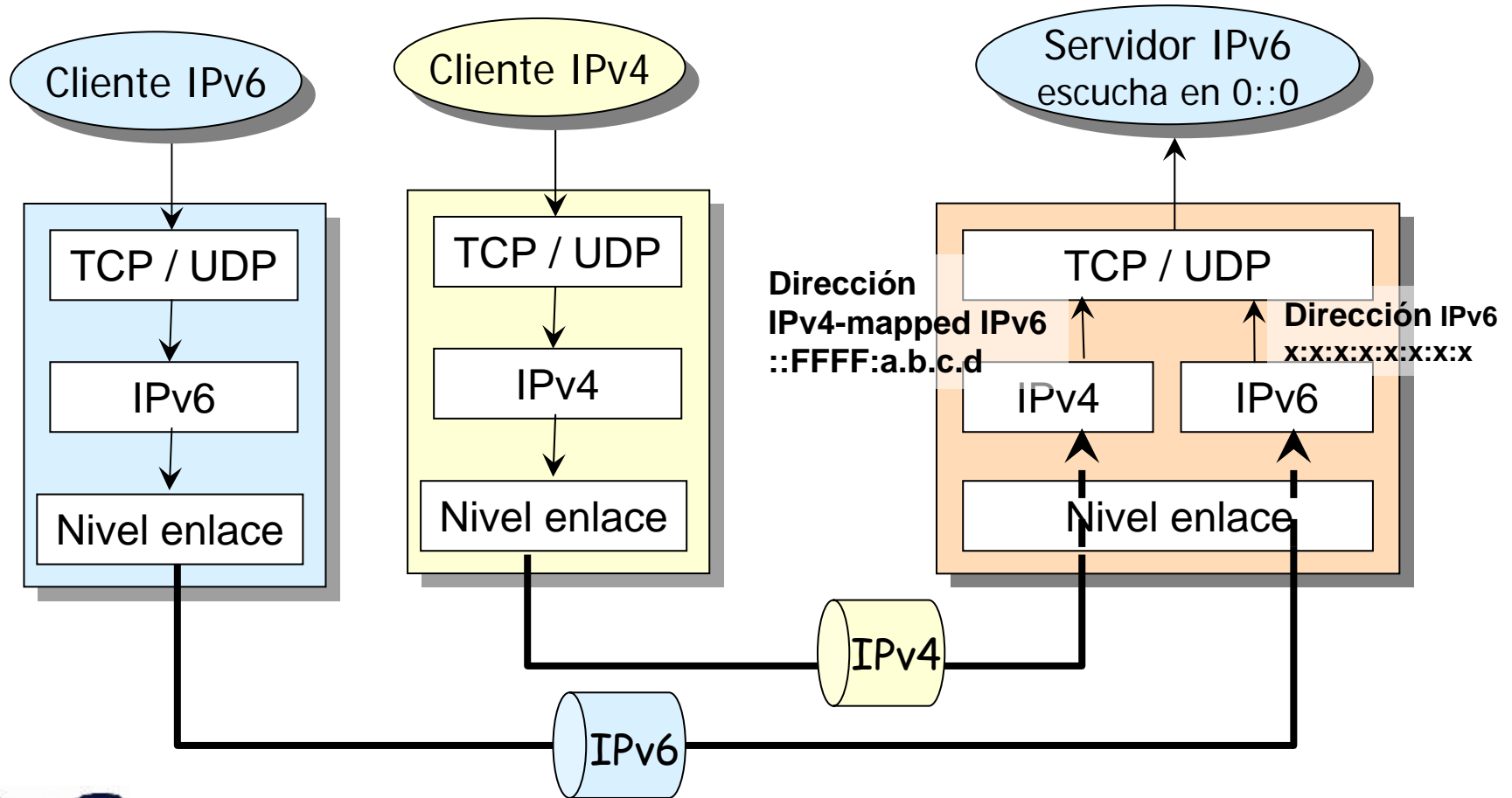


2. Aplicaciones IPv6 en nodos duales

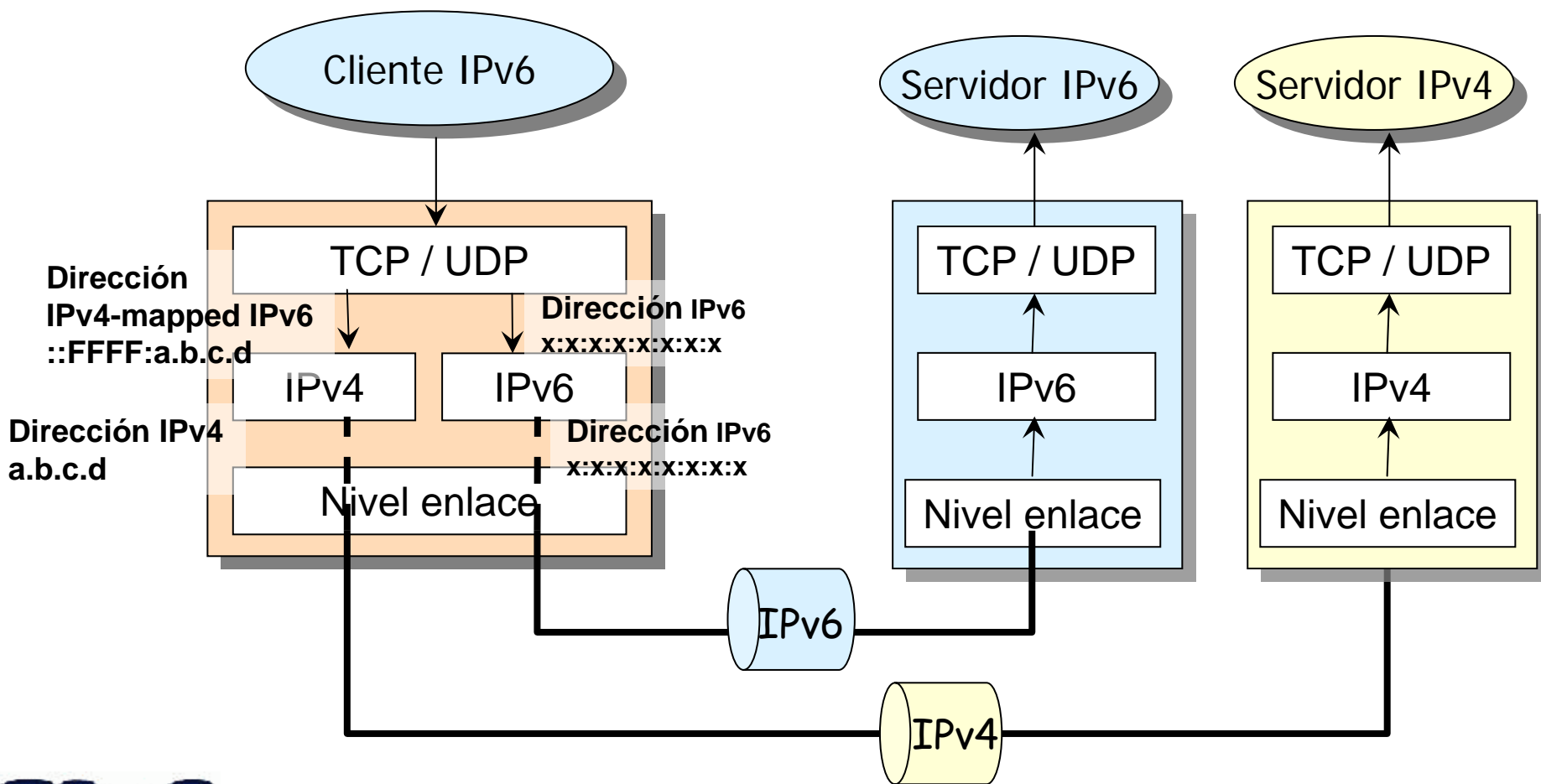
- Se han sustituido las estructuras y funciones de IPv4 por las de IPv6. Dependencias en el código fuente con IPv6
- Intercambian de paquetes IPv6.
- Para su funcionamiento en redes IPv4:
 - Portar a aplicaciones duales
 - Mantener 2 aplicaciones: *ping4* y *ping6* o
 - Utilizar direcciones IPv6 a partir de las direcciones IPv4, *IPv4-mapped IPv6 addresses*, no soportadas en todas las implementaciones.



Aplicación servidor IPv6 en nodo dual



Aplicación cliente IPv6 en nodo dual



3. Aplicaciones duales en nodos duales

- Aplicaciones válidas para redes IPv4 e IPv6:
 - Implementación de **aplicaciones cliente**:
 - Resolver nombre de máquina del servidor a las posibles direcciones IP. Intentar conectar primero usando IPv6 y si falla probar con IPv4.
 - Implementaciones de **aplicaciones servidor**:
 1. Mantener conexiones diferentes de forma explícita para IPv4 e IPv6, o
 2. Desarrollar una aplicación servidor IPv6 y confiar en las direcciones IPv4-mapped IPv6 para los clientes IPv4.

4. Aplicaciones duales en nodos sólo IPv4

- Las aplicaciones duales deberían funcionar en los nodos sólo IPv4 para evitar tener varias versiones de la misma aplicación.

REQUISITO

- Desarrollar el código fuente para que nodos que no tengan soporte del protocolo IPv6 puedan ejecutar dichas aplicaciones.

Índice

- Arquitectura de transición
- Evolución de aplicaciones
- Escenarios de transición de aplicaciones
- **Dependencias en el código fuente**
 1. Formato de presentación de las direcciones IP
 2. Resolución de nombres
 3. API de la capa de transporte
 4. Otras dependencias específicas
- Conclusiones

1. Formato de presentación de direcciones IP

- El formato de presentación es una cadena que contiene la dirección IP. Diferentes en IPv4 e IPv6:
 - IPv4: “a.b.c.d”
 - IPv6: “x:x:x:x:x:x:x:x”
- El formato de presentación en IPv6 necesita más memoria.
- Los analizadores sintácticos de direcciones deben ser revisados para adecuarse al nuevo formato.
- Ambigüedad con el carácter “:” en URLs (RFC 2732):
 - [http://\[DirecciónIPv6\]:puerto](http://[DirecciónIPv6]:puerto)
- **RECOMENDACIÓN:**
 - Utilizar FQDN (Fully Qualified Domain Name)

2. Resolución de nombres

- Tipos de resolución:
 - **Directa:**
 - a partir del nombre de la máquina obtener su dirección IP.
 - **Inversa:**
 - a partir de la dirección IP obtener el nombre de máquina.
- Utilizar FQDN.
- Las consultas/respuestas de DNS se envían usando IPv4/IPv6 independientemente del tipo de registros que se soliciten.
- **RECOMENDACIÓN:**
 - Usar las funciones de resolución independientes de protocolo.

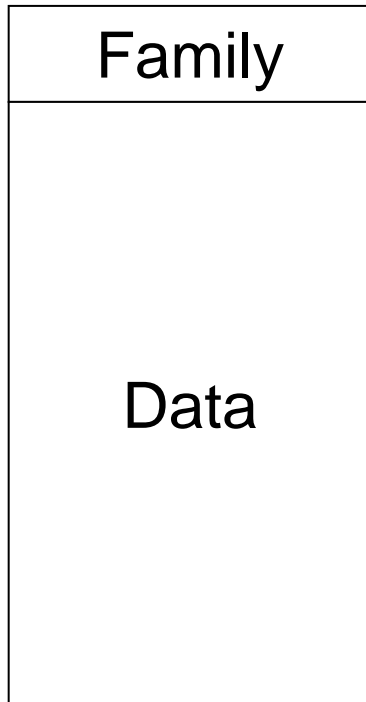
3. API de la capa de transporte

En el caso concreto del API de *sockets* existen las siguientes dependencias:

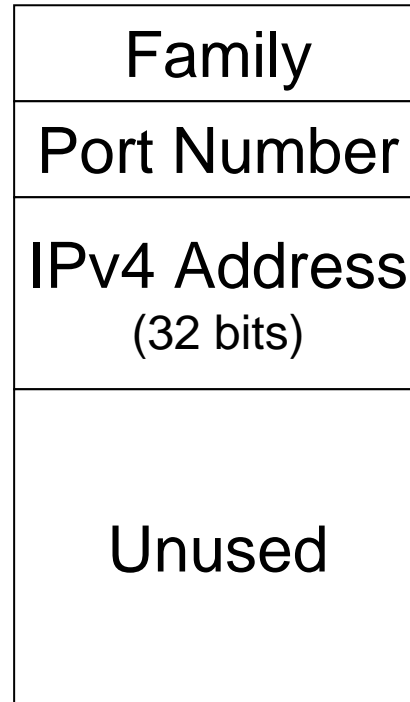
- **Estructuras de datos para las direcciones IP:**
 - *sockaddr_in*, *sockaddr_in6*, *sockaddr_storage*
 - **Funciones del API de comunicaciones:**
 - *socket()*, *bind()*, *connect()*, *read()/write()* ...
 - **Funciones de conversión de direcciones:**
 - entre el formato de presentación y las estructuras de datos de direcciones.
 - **Opciones de configuración de red.**
- **RECOMENDACIÓN:**
 - Desarrollar aplicaciones independientes de la versión IP

Estructuras de datos

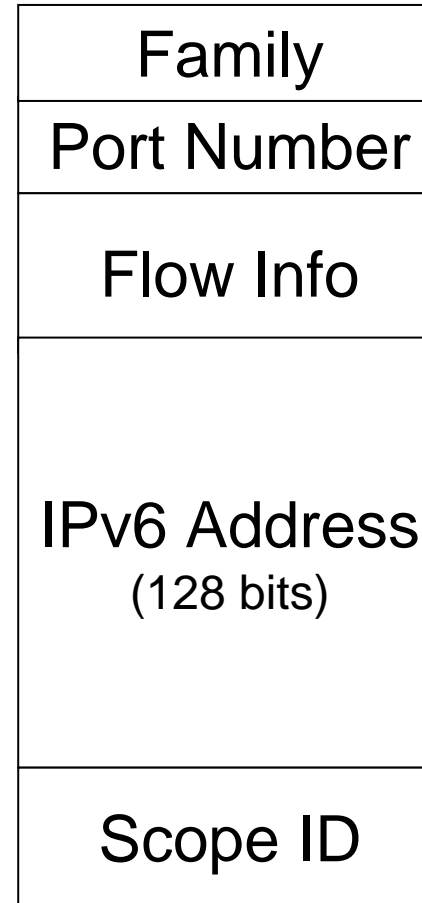
sockaddr



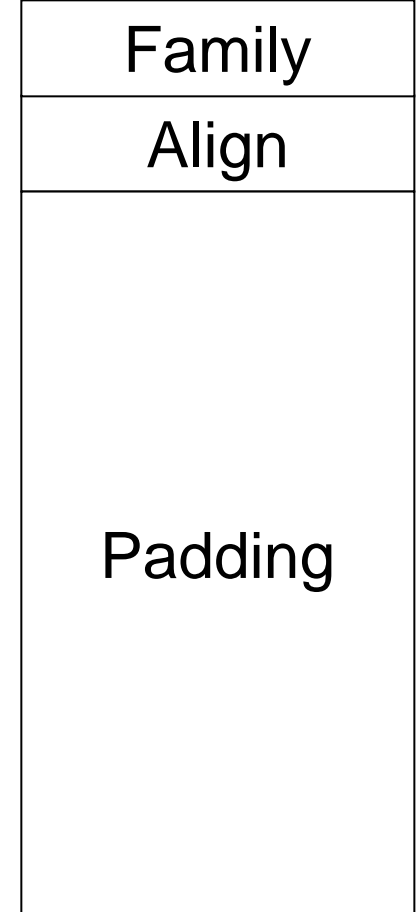
sockaddr_in



sockaddr_in6



sockaddr_storage



4. Otras dependencias específicas

- **Selección de la dirección IP:**
 - Los nodos automáticamente resuelven el problema de la selección de la dirección de origen, siguiendo una serie de reglas predefinidas (RFC 3484).
- **Fragmentación a nivel de aplicación:**
 - Cálculo del tamaño del fragmento para que no haya degradación de prestaciones por fragmentación a nivel IP .
- **Almacenamiento de direcciones IP:**
 - No almacenar direcciones IP, pueden cambiar. Si es necesario almacenar nombres y solicitar la resolución en el momento que se necesiten.
- **RECOMENDACIÓN:**
 - Revisar el código exhaustivamente

Índice

- Arquitectura de transición
- Evolución de aplicaciones
- Escenarios de transición de aplicaciones
- Dependencias en el código fuente
- **Conclusiones**

Conclusiones

- Desarrollar aplicaciones duales:
 - Válidas para cualquier tipo de nodo y para comunicarse con cualquier aplicación utilizando IPv4 o IPv6.
 - Intentar la comunicación con cada una de las direcciones IP obtenidas a través de las funciones de resolución.
- Usar FQDN:
 - Eliminar las direcciones IP cableadas del código.
- No almacenar direcciones IP.
- Las direcciones “IPv4-mapped IPv6 addresses” no siempre funcionan:
 - No siempre están implementadas.
 - En algunos nodos están deshabilitadas por seguridad.