

upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

Cloud

Cloud Computing

- *On-demand self-service*
 - El usuario puede crear nuevas instancias de servidores, almacenamiento o red por su cuenta
- *Universal network access*
 - Acceso mediante tecnologías estándar desde cualquier plataforma
- *Resource pooling*
 - Recursos compartidos entre diferentes *tenants*
- *Rapid elasticity*
 - *Provisioning* rápido o automático para un rápido *scale-out* y *scale-in*; parecen recursos ilimitados para el usuario
- *Pay per use*



Deployment models

- *Public cloud*
- *Community cloud*
- *Private cloud*
- *Hybrid cloud*



Public cloud

- Disponible para el público general o una gran industria
- Propiedad de una organización que vende estos servicios
- Ofrecidos típicamente a través de la Internet



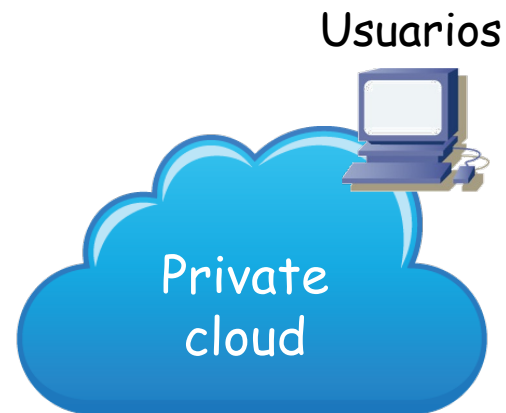
Community cloud

- Compartida por varias organizaciones
- Tienen características similares (misión, requerimientos de seguridad, políticas, cumplimiento necesario de regulación, etc)
- Gestionada por las organizaciones o por un tercero
- En sus propios edificios o en otros (*on-premises vs off-premises*)



Private cloud

- Empleada por una única organización
- Puede ser gestionada por la misma organización o por otra (servicio externalizado)
- Puede encontrarse en sus edificios o en otros



Hybrid cloud

- Utilización de infraestructura de al menos dos de los tipos anteriores para las mismas aplicaciones
- Por ejemplo:
 - Una empresa tiene su *private cloud* con recursos limitados
 - Cuando alcanza los límites, las peticiones en exceso se redirigen a una *public cloud*

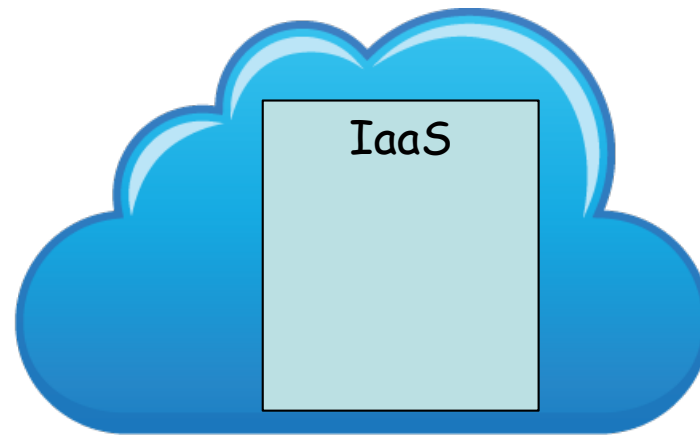


Service Models

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

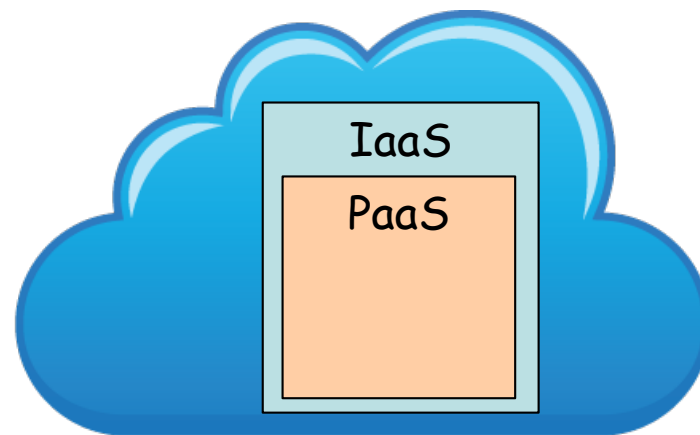
IaaS

- El cliente puede instanciar recursos de servidor, almacenamiento y/o red
- Tiene acceso a los servidores (virtuales) para poder emplear el sistema operativo que quiera
- Tiene control sobre esos sistemas operativos para instalar el software que necesite
- Ejemplos: Amazon EC2, CenturyLink Cloud, Microsoft Azure, Terremark vCloud Express, Arsys Cloud, Fujitsu IaaS Trusted Public S5, Google Compute Engine



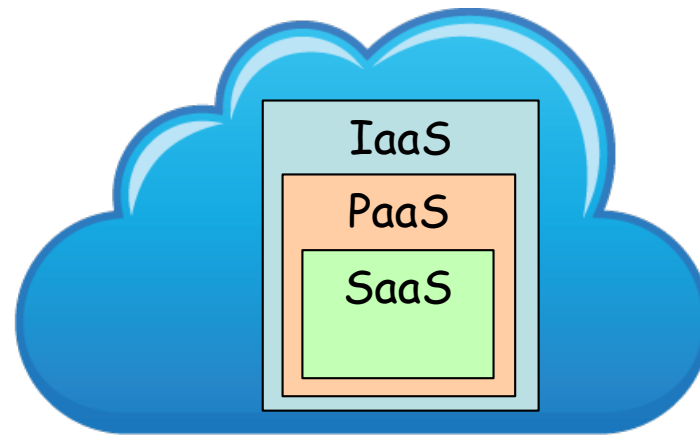
PaaS

- El cliente puede desplegar sus aplicaciones sobre la infraestructura
- Deben estar creadas empleando los lenguajes de programación y utilidades soportadas por ella
- No tiene control sobre la infraestructura
- Tiene control sobre las aplicaciones y tal vez su entorno de *hosting*
- Ejemplos: Google App Engine, AWS Elastic Beanstalk, OpenShift, Salesforce, Heroku



SaaS

- El cliente emplea las aplicaciones ofrecidas que están “en la nube” en lugar de instalarlas en sus equipos *on premises*
- Son accesibles desde diversos tipos de dispositivos
- No tiene control sobre la infraestructura, ni sistemas operativos, ni almacenamiento ni instalación de aplicaciones
- Ejemplos: Google Apps, Office 365, SharePoint Online, Cisco WebEx



upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

Cloud

upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

SDN

Internet

- Simple
- Control distribuido
- Ha permitido su gran crecimiento
- Muy bueno para los fabricantes de routers
- Sin embargo, ese crecimiento lleva a ser *commodity* (mercancía)



Equipos hardware

- La funcionalidad de red la dan equipos dedicados: switches, routers, ADCs, etc
- Dependientes de la implementación de funcionalidades en ASICs
- Eso hace su evolución muy lenta
- Y los hace propietarios respecto al desarrollador del ASIC
- Hoy en día estos ASICs y la conmutación en general son “*commodity*”
- El problema está en implementar nuevas funcionalidades
- La implementación de nuevos servicios es muy lenta en ASICs
- Es un entorno mucho menos flexible que el entorno software
- Por ejemplo si tenemos que esperar a que el fabricante del equipo corrija un bug o implemente una funcionalidad



¿SDN?

- *Software Defined Networking*
- ¿Qué es?
- El “hype” tiene poco tiempo (<10 años)
- Aunque se basa en ideas que tienen bastante más edad
- Hay confusión en los últimos años en lo que significa SDN
- ¿SDN = Un API estándar para configurar switches?
- ¿SDN = Separación del plano de datos y de control?
- ¿SDN = Plano de control centralizado?
- ¿SDN = OpenFlow?
- El movimiento actual sí empezó con OpenFlow pero hoy en día consideramos que no son lo mismo



upna

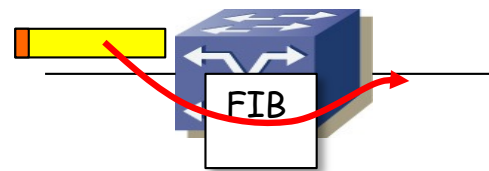
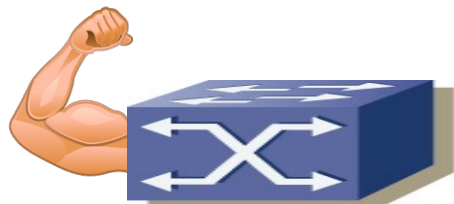
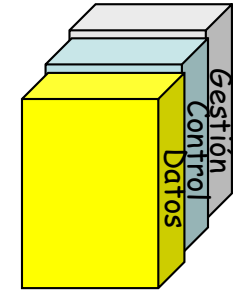
Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

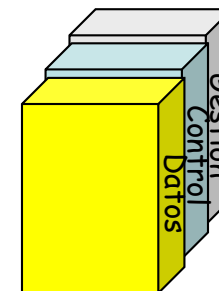
Planos

Control vs Data Plane

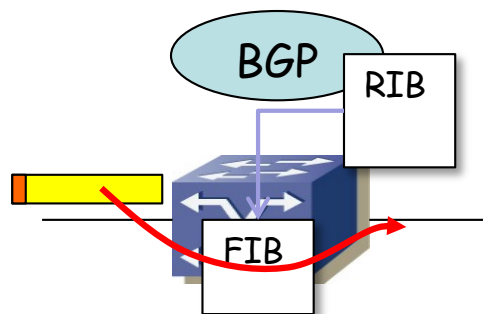
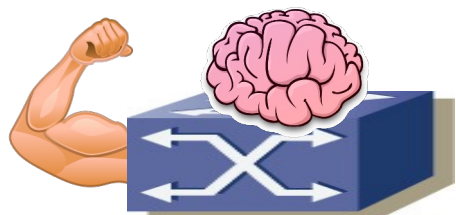
- *Data plane*
 - Conmutación, reenvío layer 2, reenvío IP (el “músculo”)
 - Aquí tenemos las abstracciones de las capas
- (...)



Control vs Data Plane



- *Data plane*
 - Conmutación, reenvío layer 2, reenvío IP (el “músculo”)
 - Aquí tenemos las abstracciones de las capas
- *Control plane*
 - Señalización y control, routing protocols (aprendizaje, el “cerebro”)
 - Los datos empleados para conocer la topología
 - Aquí no tenemos una forma abstracta de resolver el problema
- Management plane



Control plane

- ¿Cómo hacemos evolucionar hoy en día el plano de control?
- Nos inventamos un nuevo protocolo desde cero (y esperamos a que se implemente, compremos hardware que lo soporte, etc)
- O reconfiguramos algún mecanismo existente (por ejemplo para hacer ingeniería de tráfico)
- O hacemos configuración manual (ACLs, middleboxes, routers domésticos, etc)



SDN

- Pretende ofrecer abstracciones
- No el “cómo”
- No es una solución a un problema, no es un nuevo protocolo o una nueva tecnología
- Es una nueva arquitectura
- Es un facilitador de nueva innovación
- *“Software Defined Networking”*
- Networking quiere evolucionar hacia el software
- El software definirá las redes, cómo se comportan, etc
- La red pasa a ser infraestructura abstracta como para las redes lo son los enlaces
- Que podremos administrar y controlar automática y dinámicamente
- Muchas organizaciones se han involucrado: IETF, ITU, ONF, ETSI, etc.

upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

Planos

upna

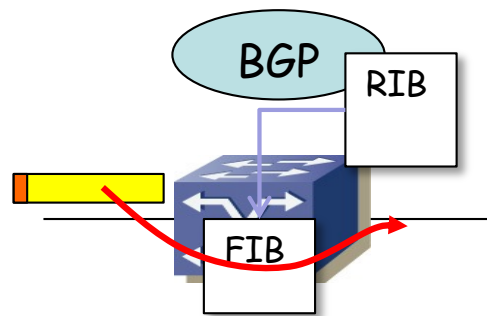
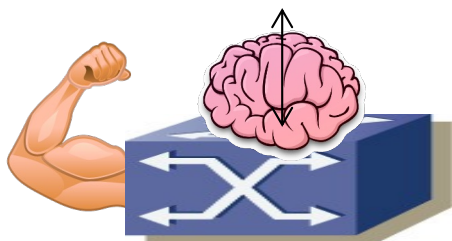
Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

Elementos en SDN

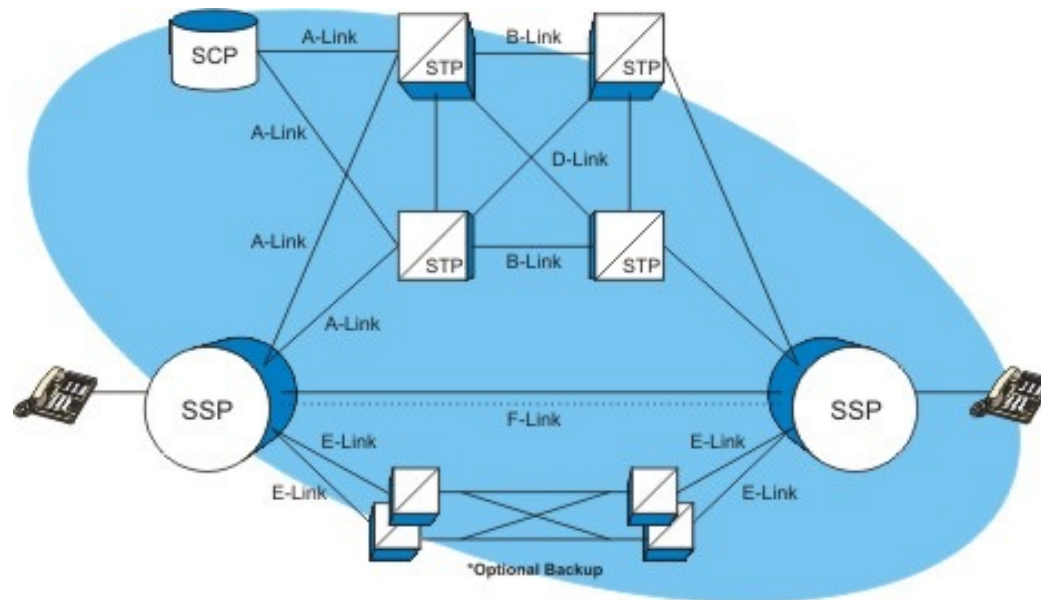
Control vs Data Plane

- En Internet el desarrollo se hizo basado en un control distribuido
- Es decir, ambos están en el mismo equipo, implementados por el fabricante
- *Software Defined Networking* (SDN) se basa en la separación de ambos y comunicación mediante un interfaz abierto (...)
- La propuesta del Open Networking Forum (ONF) es OpenFlow
- Eso no quiere decir que SDN sea igual a OF



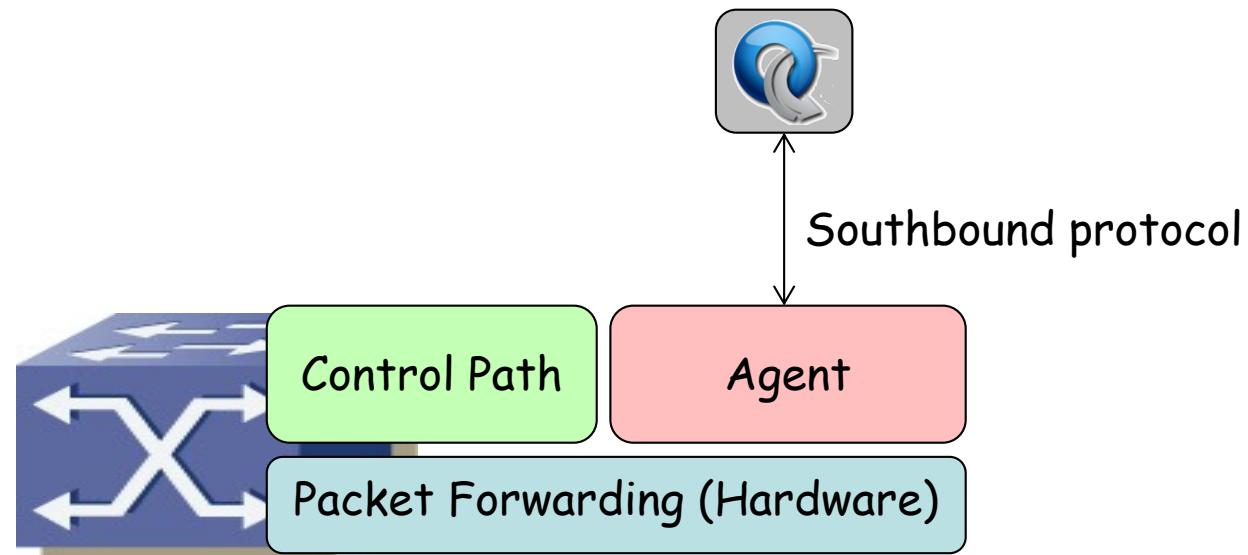
Separación control-datos

- Simplifica la evolución del control pues no está “atado” al hardware
- Permite el desarrollo de software de más alto nivel, así como su depuración, testing, simulación, etc
- La red telefónica ya tenía separado el control a elementos de señalización y control de red
- Especialmente útil en data centers y en IXPs
- Permite la optimización de los flujos
- También para una arquitectura con middleboxes
- También en el entorno WAN controlado por la misma empresa



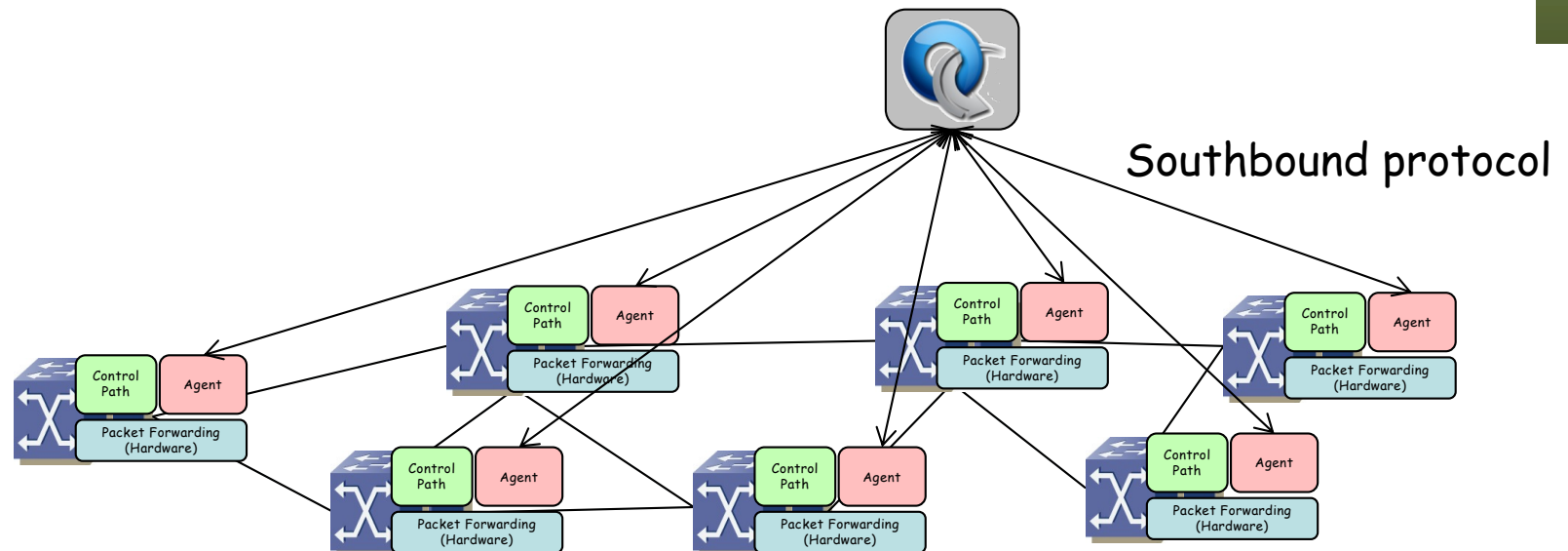
Controlador

- Esta arquitectura permite tener centralizado el plano de control
- Hoy en día el concepto de SDN no obliga a tenerlo centralizado
- Se comunica con el dispositivo mediante un *Southbound protocol*
- Para un gran número de dispositivos (...)



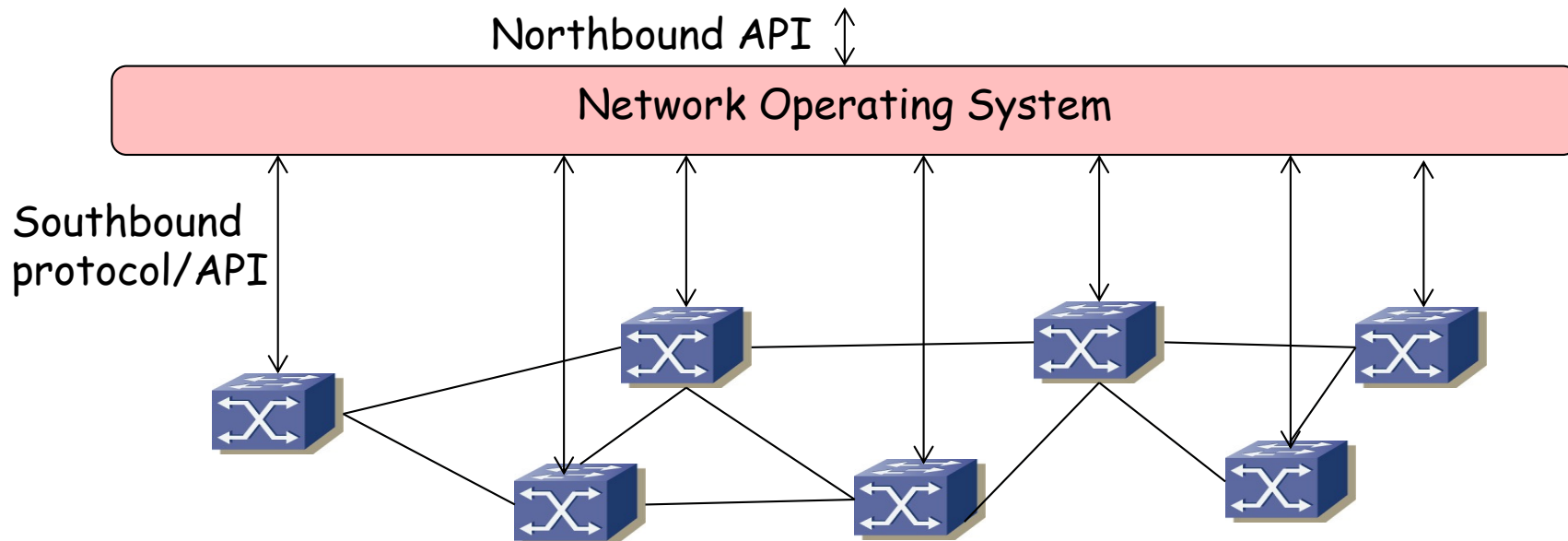
Controlador

- Esta arquitectura permite tener centralizado el plano de control
- Hoy en día el concepto de SDN no obliga a tenerlo centralizado
- Se comunica con el dispositivo mediante un *Southbound protocol*
- Para un gran número de dispositivos



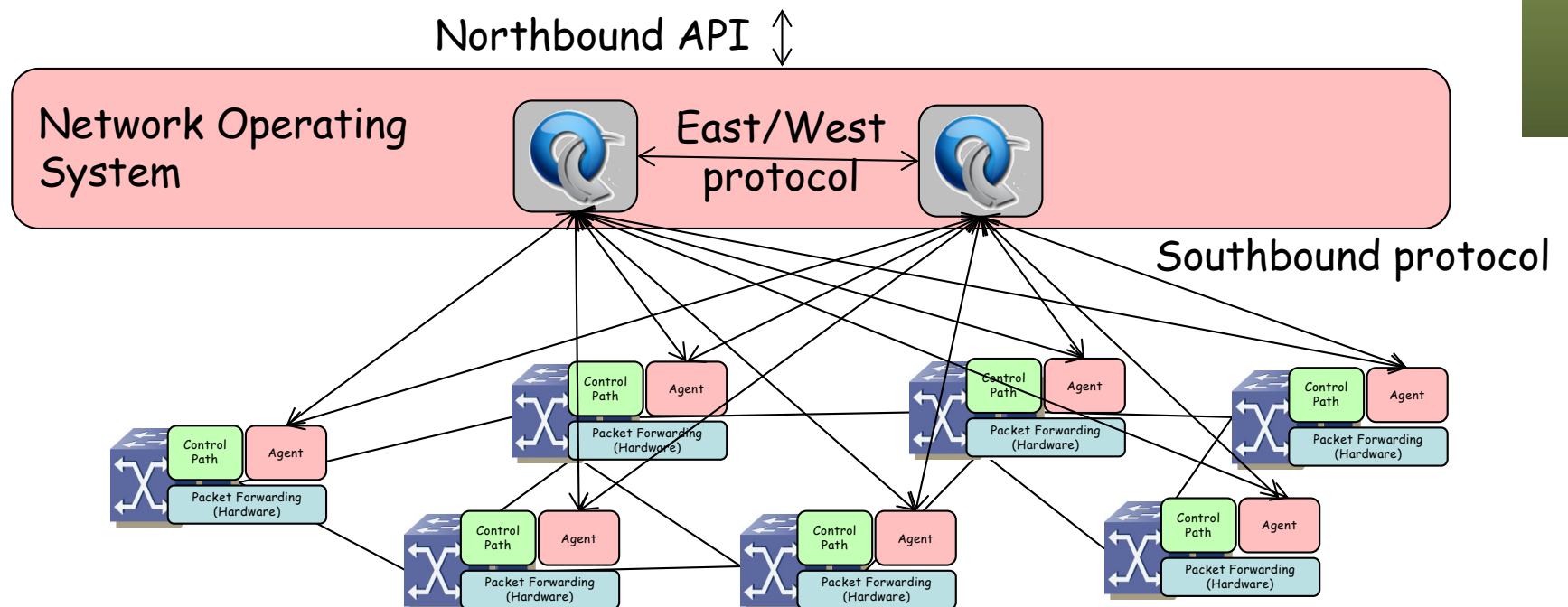
Network Operating System

- Tenemos una visión global de la red
- Mediante lo que se está viniendo a llamar un NOS
- El NOS es software en servidores que habla con los conmutadores
- El NOS da una visión virtualizada de la red, un grafo y un API (*"northbound"*)
- Sobre ella podemos escribir los programas de control
- Nos aísla del hardware, igual que un OS del hardware del PC



High Availability

- Para mayor disponibilidad no tendremos un solo controlador sino varios
- La comunicación entre los controladores se lleva a cabo mediante lo que se llama un *East/West Protocol*



upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

Elementos en SDN

upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

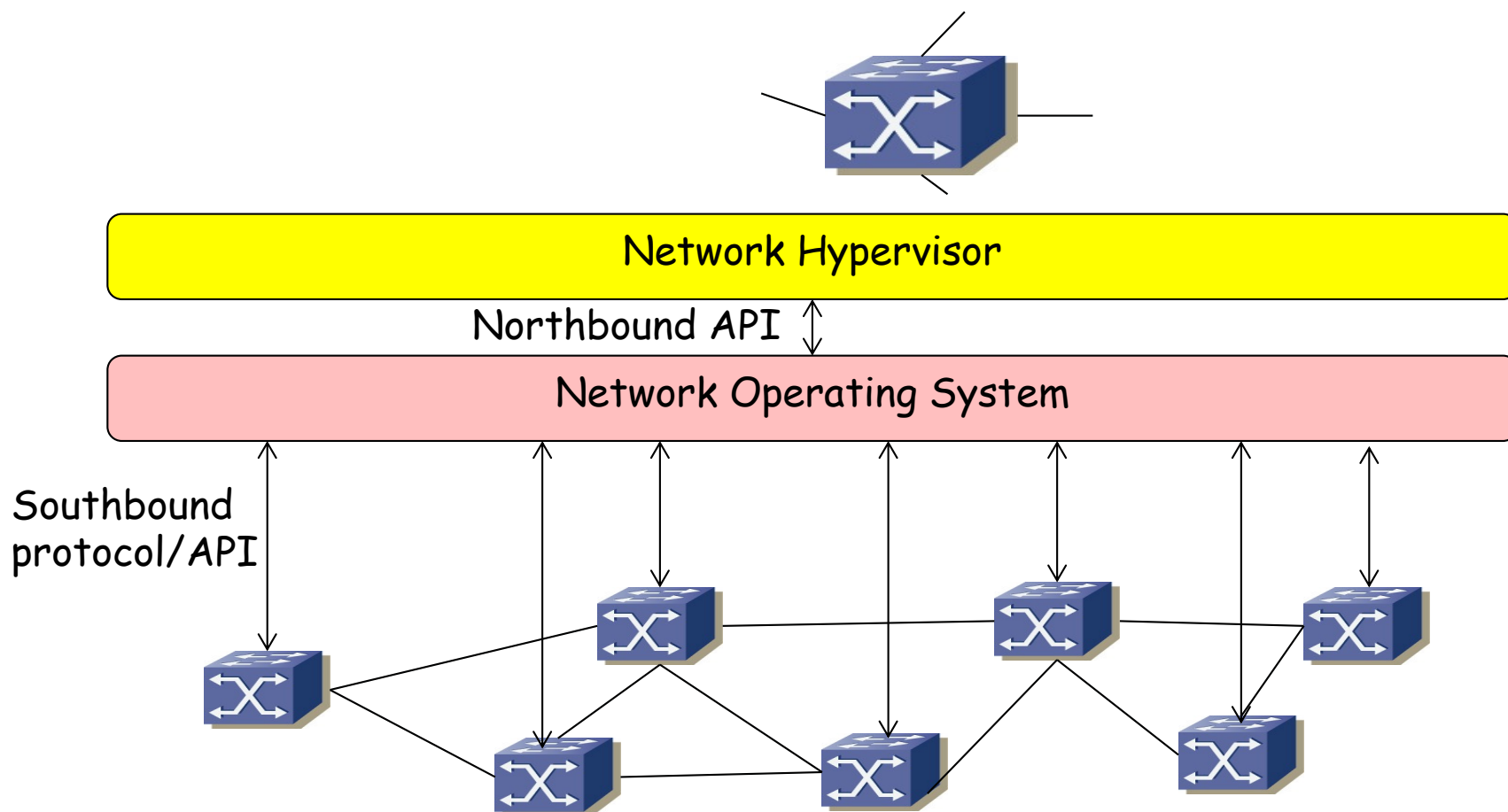


Network Virtualization



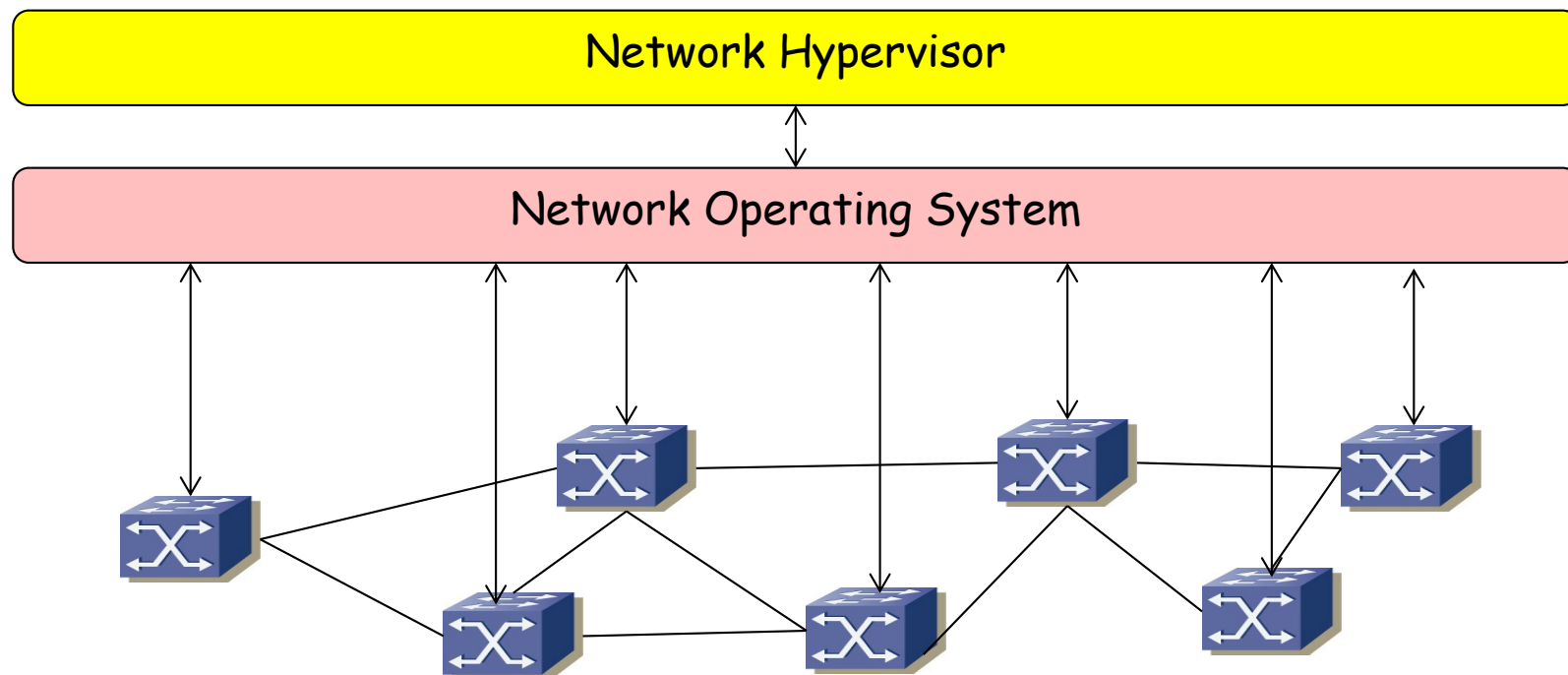
Network Virtualization

- La visión que da el NOS es virtual
- Puede ser la más adecuada para el problema que tenga que resolver el programa de control
- Sobre el NOS un *Network Hypervisor*



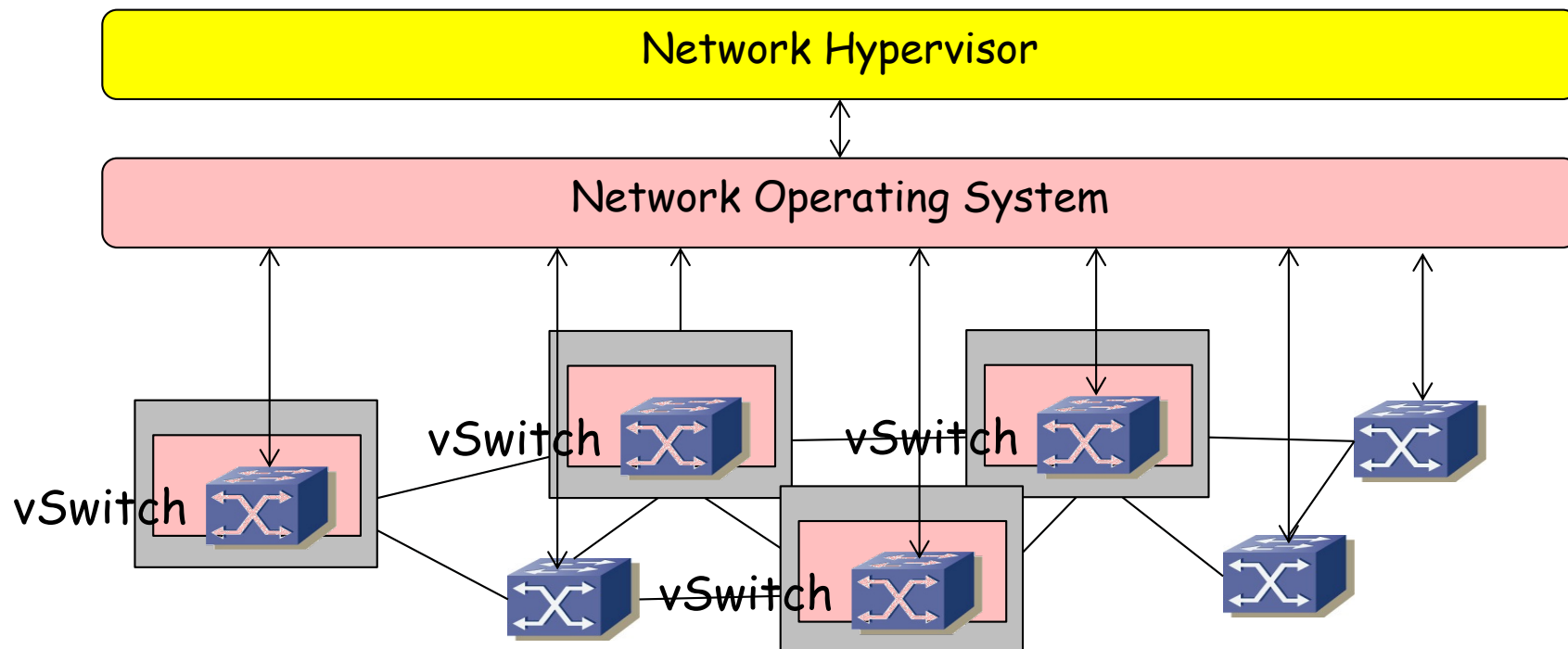
Network Virtualization

- ¿Y esos conmutadores?
- (...)



Network Virtualization

- ¿Y esos conmutadores?
- Ya no son simplemente conmutadores hardware, también vSwitches
- Hoy en día tenemos ya más puertos de hosts virtuales que físicos
- Un core x86 puede reenviar más de 20Mpps IPv4
- 1Mpps de 64bytes = 500Mbps; 1Mpps de 1518bytes = 12Gbps
- La frontera (edge) puede implementarse en software
- Podemos simplificar el core y volver el edge controlado por software



Ejemplo

- Middleboxes: lo más frecuente es que estén basados en arquitectura x86
- Están en general en el camino del tráfico
- Hacen mucho más que simple reenvío capa 2 ó 3
- Y pueden con ello
- Por cierto, ¿hay muchos? (...)

Ejemplo

- Middleboxes: lo más frecuente es que estén basados en arquitectura x86
- Están en general en el camino del tráfico
- Hacen mucho más que simple reenvío capa 2 ó 3
- Y pueden con ello
- Por cierto, ¿hay muchos?

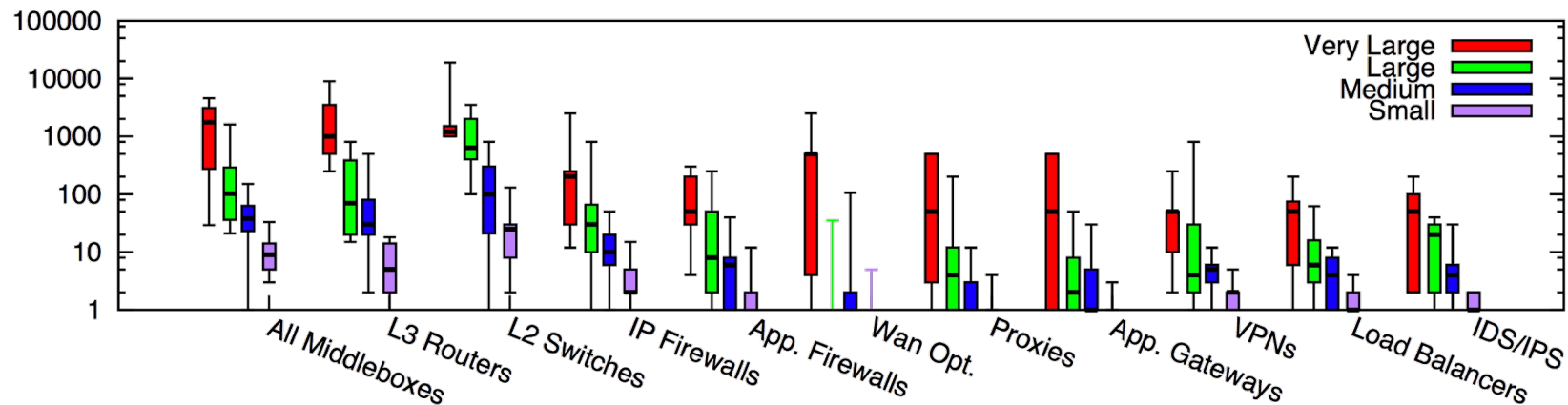


Figure 1: Box plot of middlebox deployments for small (fewer than 1k hosts), medium (1k-10k hosts), large (10k-100k hosts), and very large (more than 100k hosts) enterprise networks. Y-axis is in log scale.

Network Virtualization