



SDN – heat & light

*Joel Obstfeld
Director of Engineering
SP CTO team*

November 2012

How do you define SDN?

Controller architecture that efficiently computes & fits application instances onto the network

31.3%

Switched network w/ more controllers & APIs to help virtualize the network

31.3%

SDN is just about network programability & new APIs

28.1%

Software that makes network orchestration concurrent w/ application deployment

28.1%

Software that derives network topology from an application

12.5%

SDN just adds computation & algorithms to a switched network

9.4%

0.0% 5.0% 10.0% 15.0% 20.0% 25.0% 30.0% 35.0%

Sources: ACG Research, Cisco 2012

SDN – from the providers

- There are multiple SDN definitions which creates complexity – are we even talking about the same thing? - *Asian SP*
- The biggest challenge is how to operationalize a network driven by software. Where would the NOC guys start? - *US SP/Telco*
- SDN's will become interesting when it offers, at a minimum, parity to today's network infrastructure. Until that time, we see no compelling use-case that SDNs enable - *EMEA SP/Telco*
- We know that the network contains a wealth of data. If we can get it, how do we monetize it? - *Asian SP*

Customer Insights



Research/ Academia

Flexibility to support experimental SDN components in a production network environment

Network Slicing



Massively Scalable Data Center

Customize with APIs to provide insight into network traffic and enable optimization

Network flow management



Cloud

Automated provisioning and programmable overlays

Scalable Multi-tenancy



Service Providers

Policy-based control and analytics to optimize and monetize services

Agile service delivery



Enterprise

Virtualization, distribution and orchestration - with security

Private Cloud Automation

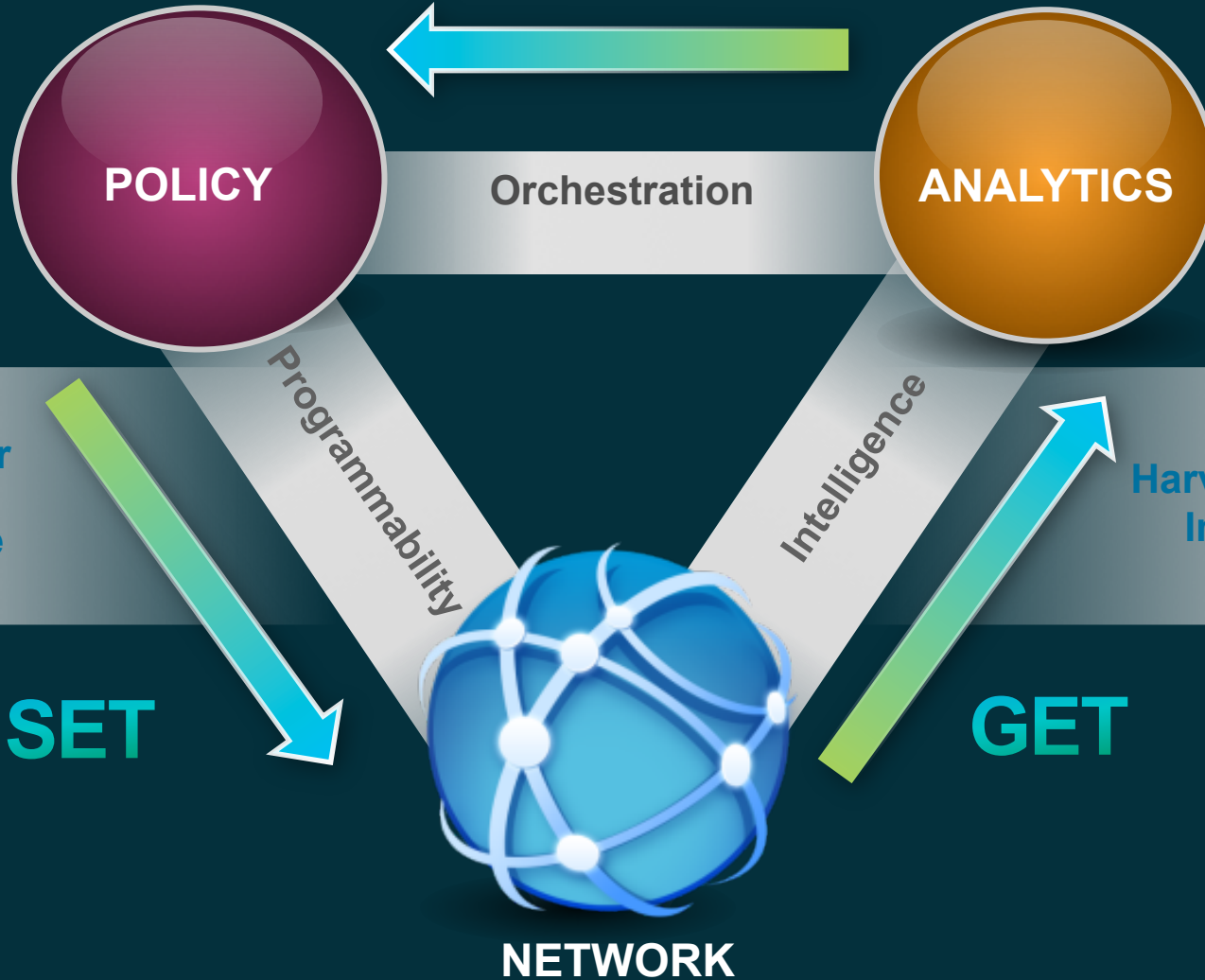
Diverse functionality required across segments

So what are the problems to be solved?

- More efficient usage of network – lower cost per unit while improving customer experience
- Need for faster service implementation
- Operationalize (and monetize) information already being collected in network elements
- Enable new services & revenue sources

All we need is programmability?

CHECK



Approaching Service Abstractions

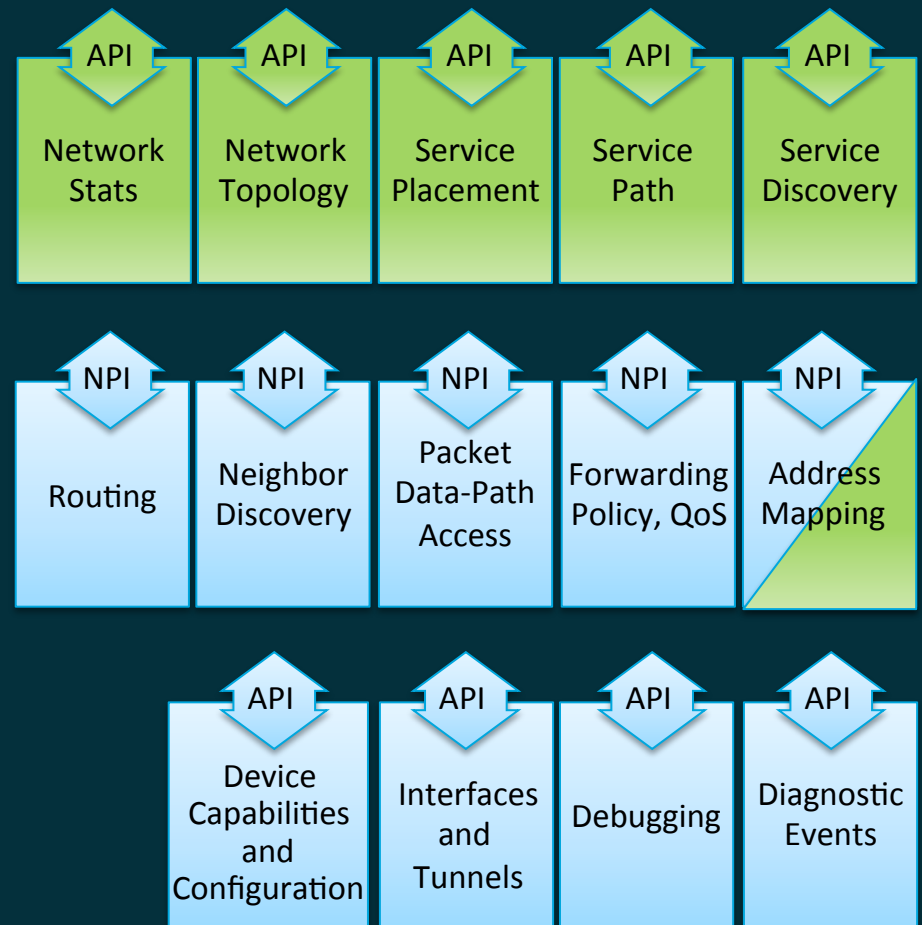
- Abstractions allow the definition of layered APIs and NPIs

Enable multi-layer APIs across all elements, to integrate with operator development environments

Accelerate development of network applications: Integrated stack from device to network

Multiple deployment modes, local and remote APIs

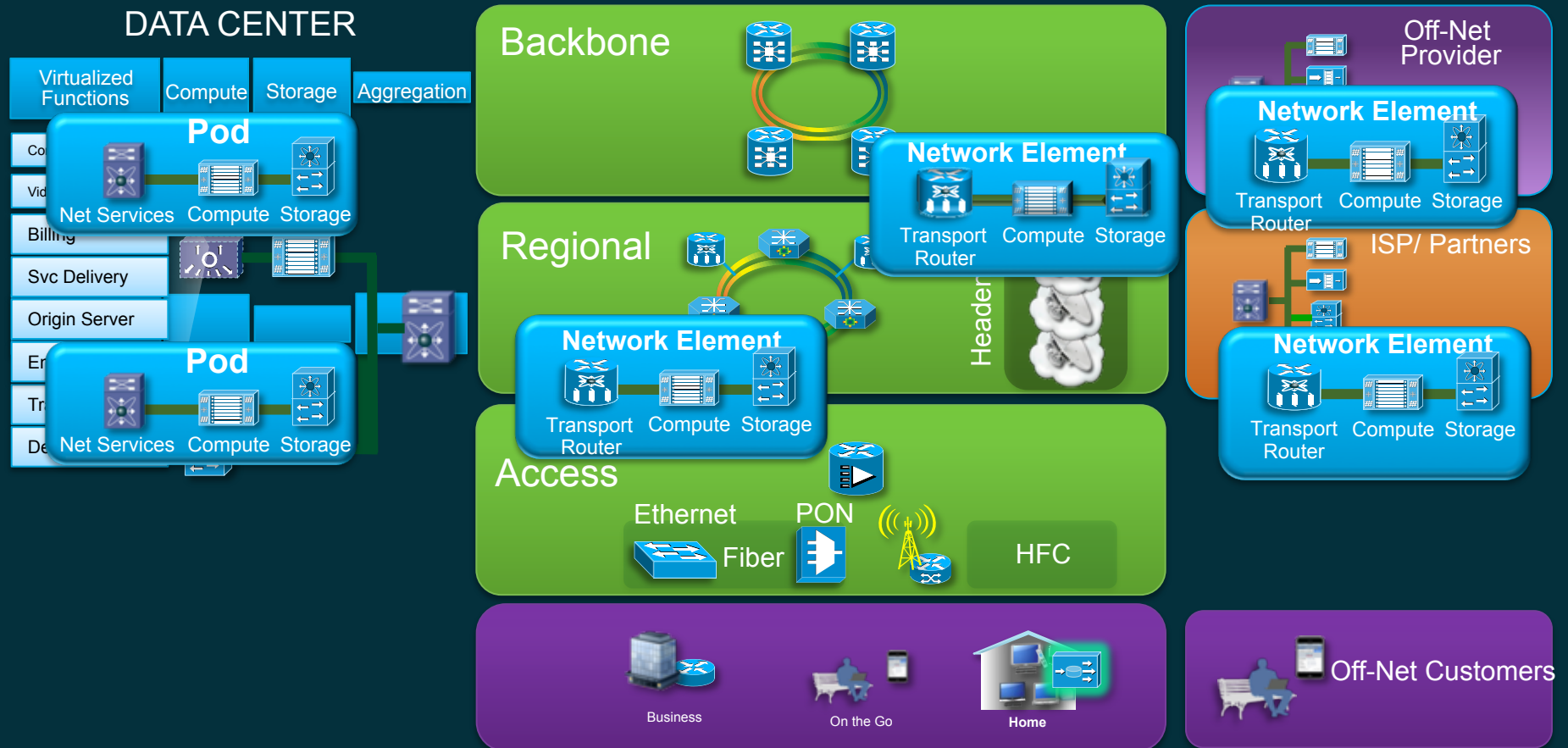
Multiple Language Bindings and Virtualization options



Example Abstractions

■ Device focused abstractions ■ Network focused abstractions

Content, Applications & Resources Where you need them



- Enable optimal resource usage
- Enable higher quality services with increased service velocity

A few elephants in the room...

- Billing systems – if I can't bill, I can't get revenue
How to interface dynamic services with billing systems premised on static services?
- The law – how do the rules apply?
As a customer – eg. where is my data right now?
As a provider – eg. what requirements for traceability must I meet?

Thank You..

<http://www.cisco.com/go/one>