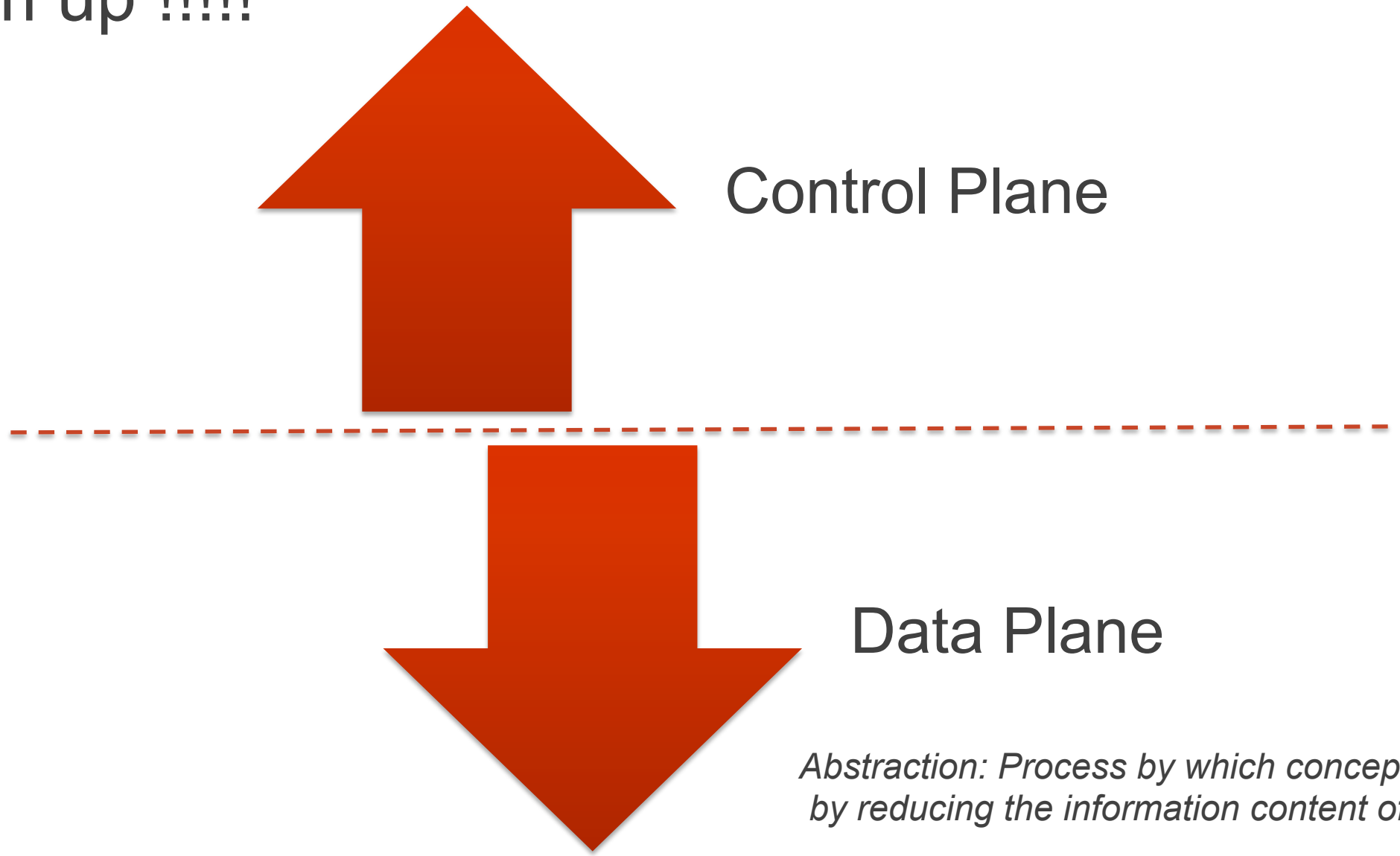
Warm up !!!!!



Abstraction: Process by which concepts are derived by reducing the information content of a concept …

© 2012 Cisco and/or its affiliates. All rights reserved.

Cisco Connect

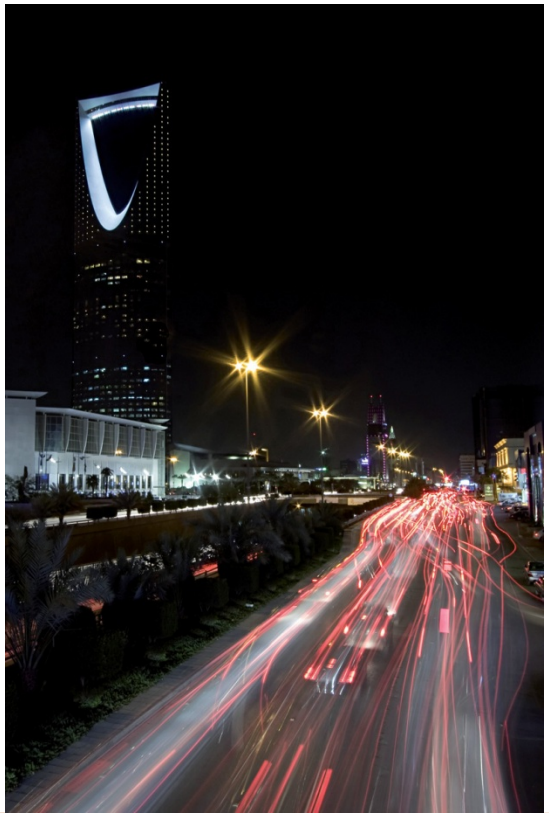
Riyadh,Saudi Arabia
Febuary 5th , 2013

Software Defined Networking

The Future of Networks & Protocols

Zubair Khan

Solutions Architect



cisco

"…In the SDN architecture, the control and data planes are decoupled, network intelligence and state are logically centralized, and the underlying network infrastructure is abstracted from the applications…"

https://www.opennetworking.org/images/stories/downloads/white-papers/wp-sdn-newnorm.pdf

"…open standard that enables researchers to run experimental protocols in campus networks. Provides standard hook for researchers to run experiments, without exposing internal working of vendor devices……"

http://www.openflow.org/wp/learnmore/

ıtes. All rights reserved.

Headlines

"Google revamps networks with OpenFlow"
—ZDnet

"Prediction: OpenFlow Is Dead by 2014; SDN Reborn in Network Management"
—Mike Fratto, Network Computing

"Will OpenFlow commoditize networks? Impact Cisco margins?"
—Several media publications, Bloggers

".We share a more pragmatic view, noting Cisco (for example) is likely to view SDN as a TAM expansion opportunity…" *—Deutsche Bank Research note, Wired, April 2012*

"Hype around SDN/OpenFlow getting way out of Control. Where have I seen this before…" *—Ethereal mind, Blogger*

"SDN needs a bigger definition"
—Lippis report, 2012

"A way to optimize link utilization in my network, through new multi-path algorithms"

"An open solution for customized flow forwarding control in the Data-Center"

"An open solution for VM mobility in the Data-Center"

"A platform for developing new control planes"

"Develop solutions software speeds: I don't want to work with my network vendor or go through lengthy standardization."

"A way to reduce the CAPEX of my network and leverage commodity switches"

"A way to avoid lock-in to a single networking vendor"

"A solution to build a very large scale layer-2 network"

"A means to do traffic engineering without MPLS"

Diverse Drivers

Common Concepts

Different Execution Paths

"A means to scale my fixed/mobile gateways and optimize their placement"

"A way to define virtual networks with specific topologies for my multi-tenant Data-Center"

"A way to build my own security/ encryption solution, avoiding RSA"

"A solution to build virtual topologies with optimum multicast forwarding behavior"

"A way to configure my entire network as a whole rather than individual devices"

"A way to scale my firewalls and load balancers"

"A solution to get a global view of the network – topology and state"

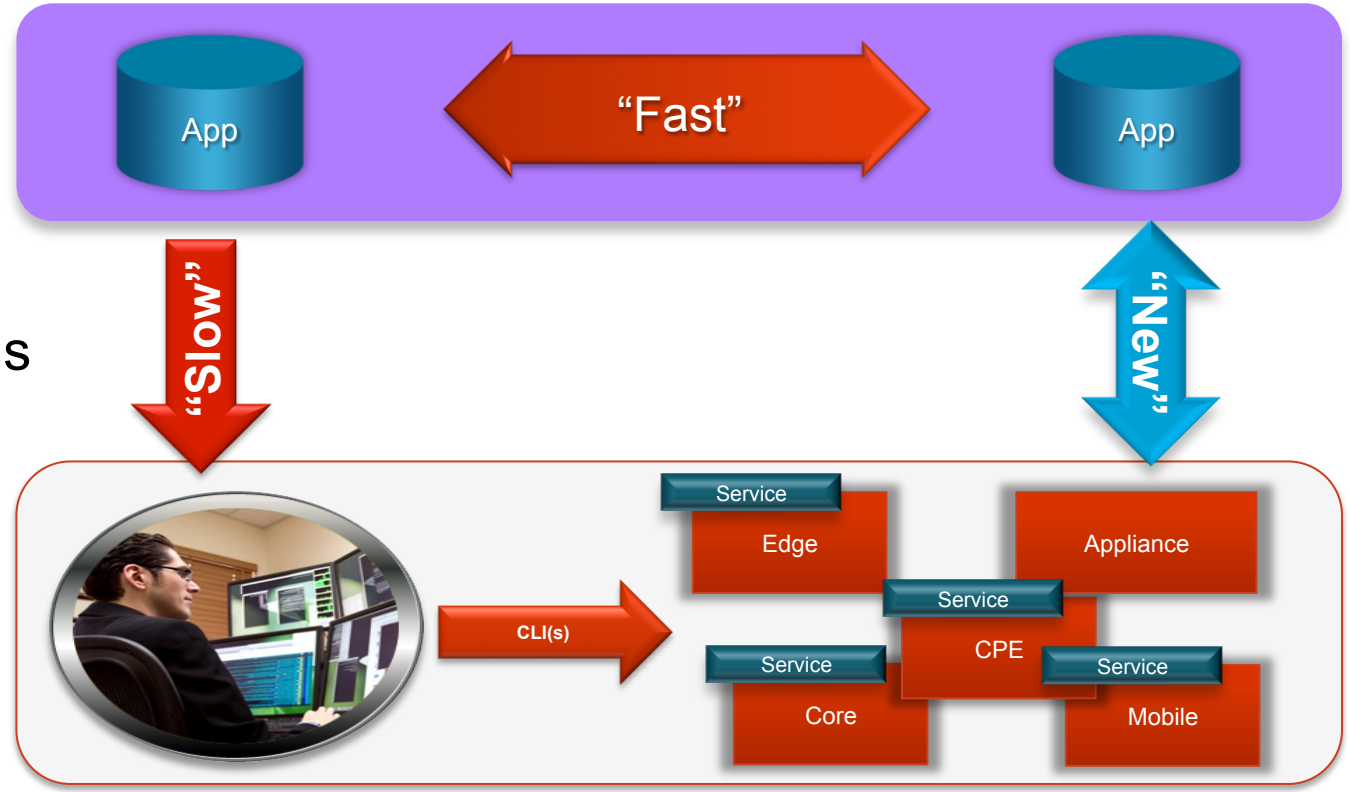
"A way to distribute policy/intent, e.g. for DDoS prevention, in the network"

ıtes. All rights reserved.

Towards Programmatic Interfaces to the Network

Approaching Today's Application Developer Dilemma

- Many Network Applications today:
  - OTT – for speed and agility
  - Avoid network interaction – complex and slow innovation
- New Model for Network Applications
  - Keep speed and agility
  - Full-duplex interaction with the network across multiple planes – extract, control, leverage network state



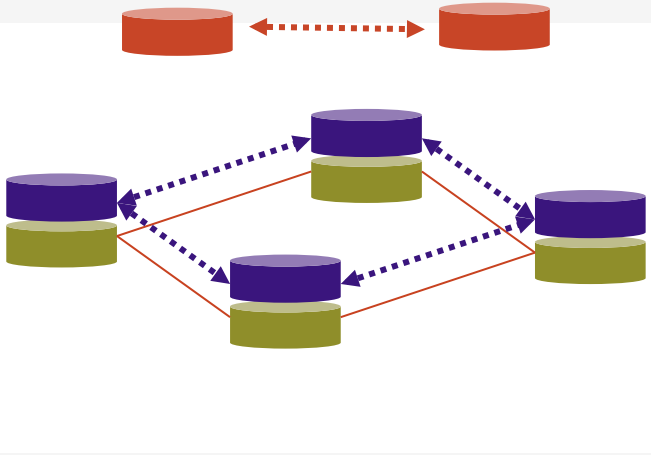
A New Programming Paradigm is Needed

ıtes. All rights reserved.

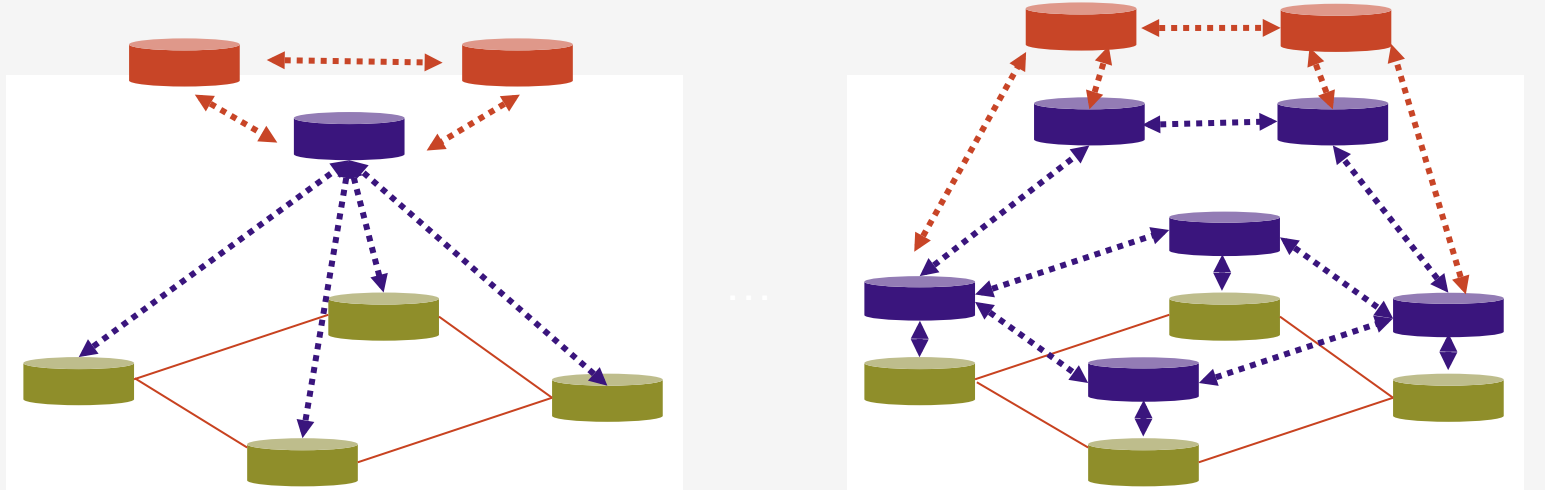
Open Network Environment for SDN

Implementation Perspective: Evolve the Control-Plane Architecture

Traditional Control Plane Architecture



Control Plane Architecture with SDN (Examples)

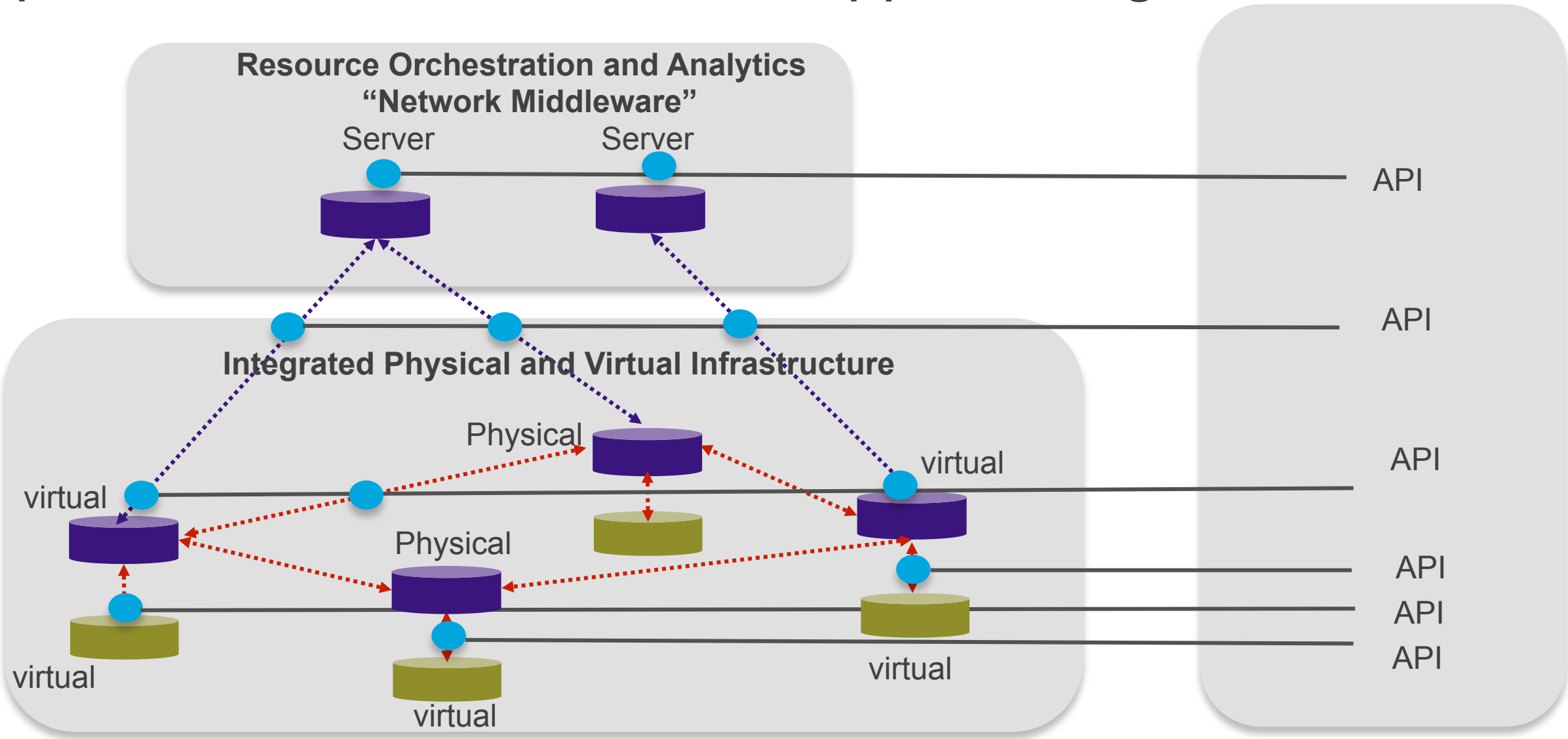


- Enable modularization and componentization of network control- and data-plane functions, with associated open interfaces. This allows for optimized placement of these components (network devices, dedicated servers, application servers) and close interlock between applications and network functions.
- Anticipated benefits include: Closely align the control plane with the needs of applications, enable a clear control-/data-plane split, improve performance and robustness, enhance manageability, operations and consistency

Control-plane component(s) Data-plane component(s) Application Component(s)

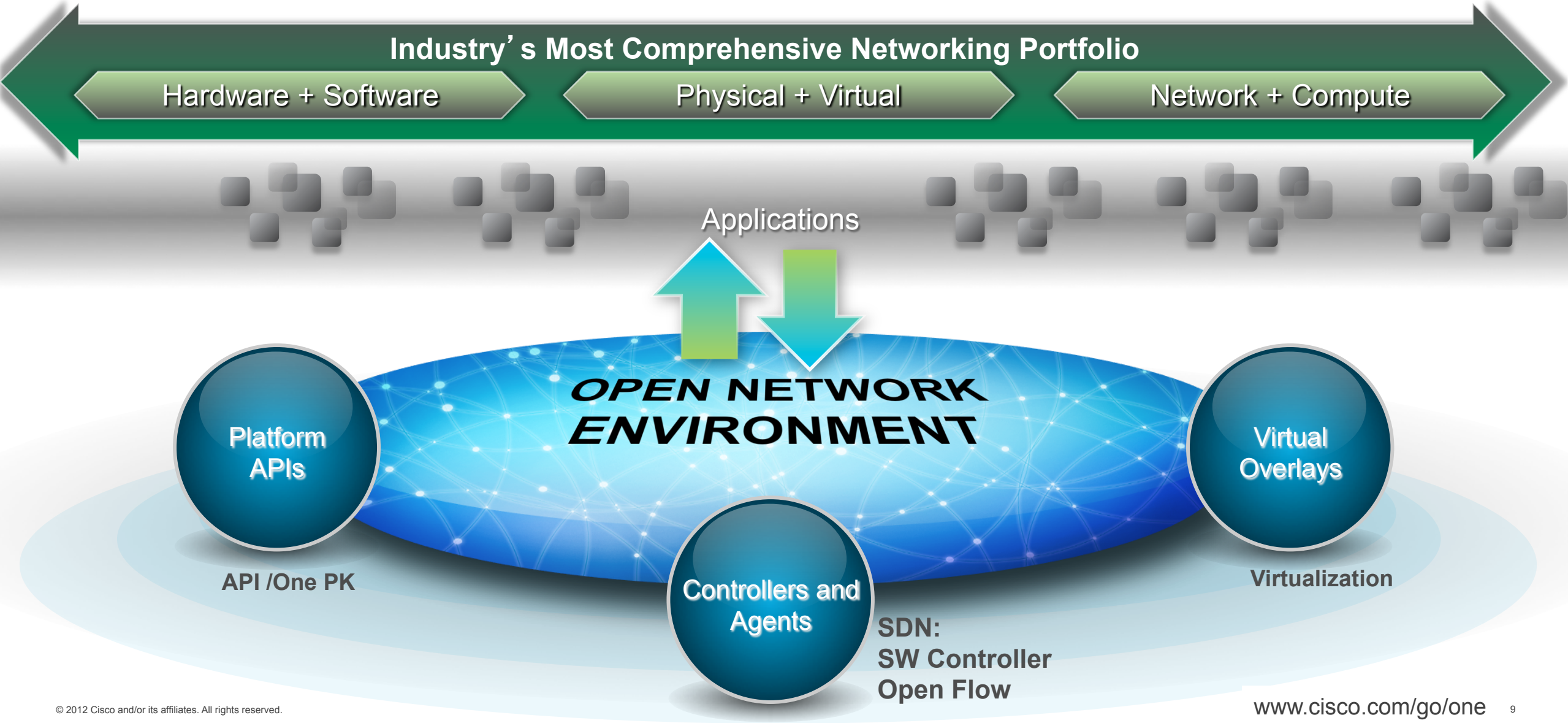
ıtes. All rights reserved.

Open Network Environment – Approaching a Definition



ıtes. All rights reserved.

Cisco Open Network Environment



© 2012 Cisco and/or its affiliates. All rights reserved.

Open Network Environment Qualities Programmatic APIs

The Need for Abstractions

Abstractions in Networking

- Data-plane Abstractions – ISO/OSI Layering
  - Examples
    - Local best effort delivery (e.g., Ethernet)
    - Global best effort delivery (e.g., IP)
    - Reliable byte-stream (e.g., TCP)
  - Data plane abstractions are key to Internet's success
- Abstractions for the other planes (control, services, management, orchestration,..) … are missing
  - Consequences include:
    - Notorious difficulty of e.g. network management solutions
    - Difficulty of evolving software for these planes

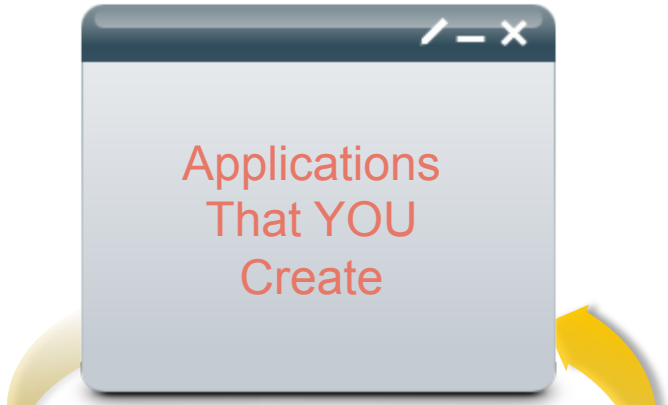
"Modularity based on abstraction is the way things get done"

Barbara Liskov
Turing Award Winner

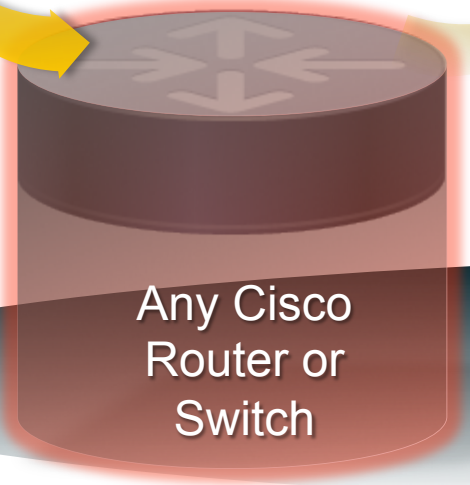
ıtes. All rights reserved.

APIs make Abstractions available to Programmers

Example: Cisco's onePK (one Programming Kit) – Get your build on!



onePK

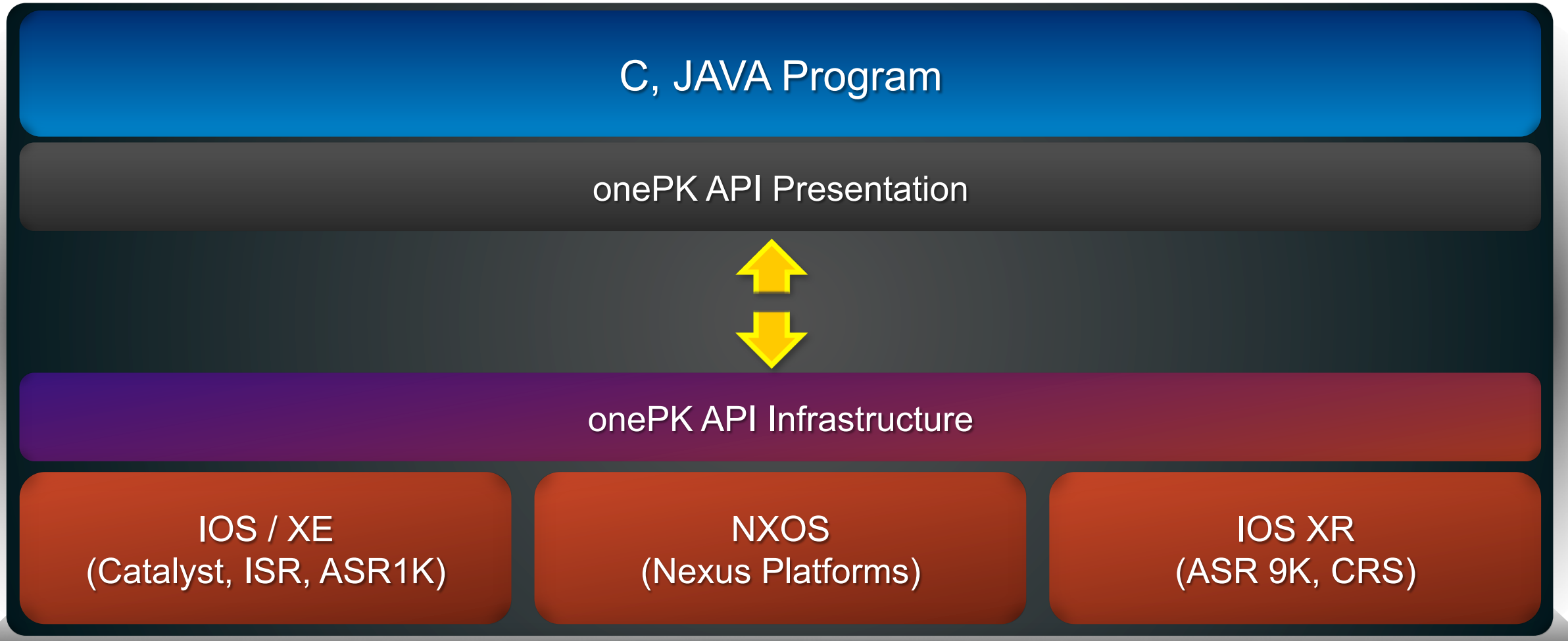


Flexible development environment to:

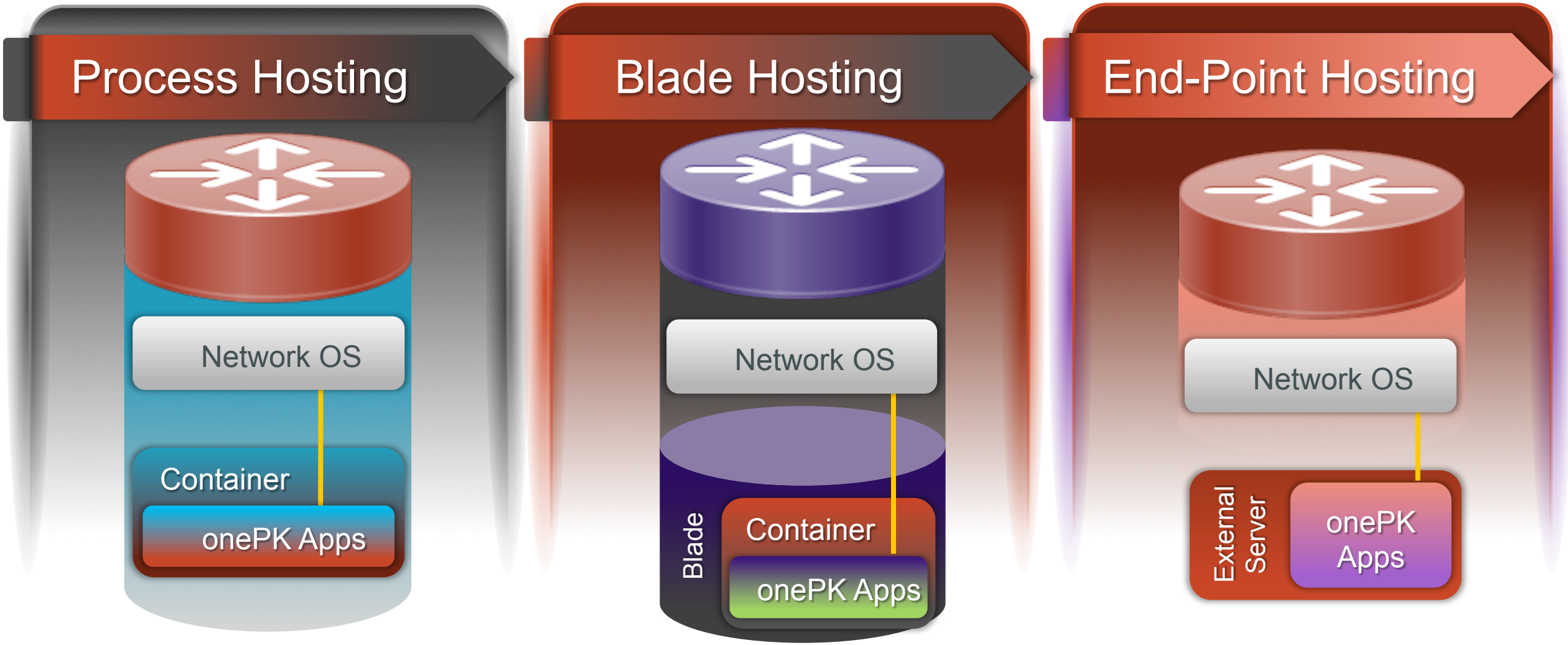
- Innovate
- Extend
- Automate
- Customize
- Enhance
- Modify

© 2012 Cisco and/or its affiliates. All rights reserved.

onePK Architecture



© 2012 Cisco and/or its affiliates. All rights reserved.

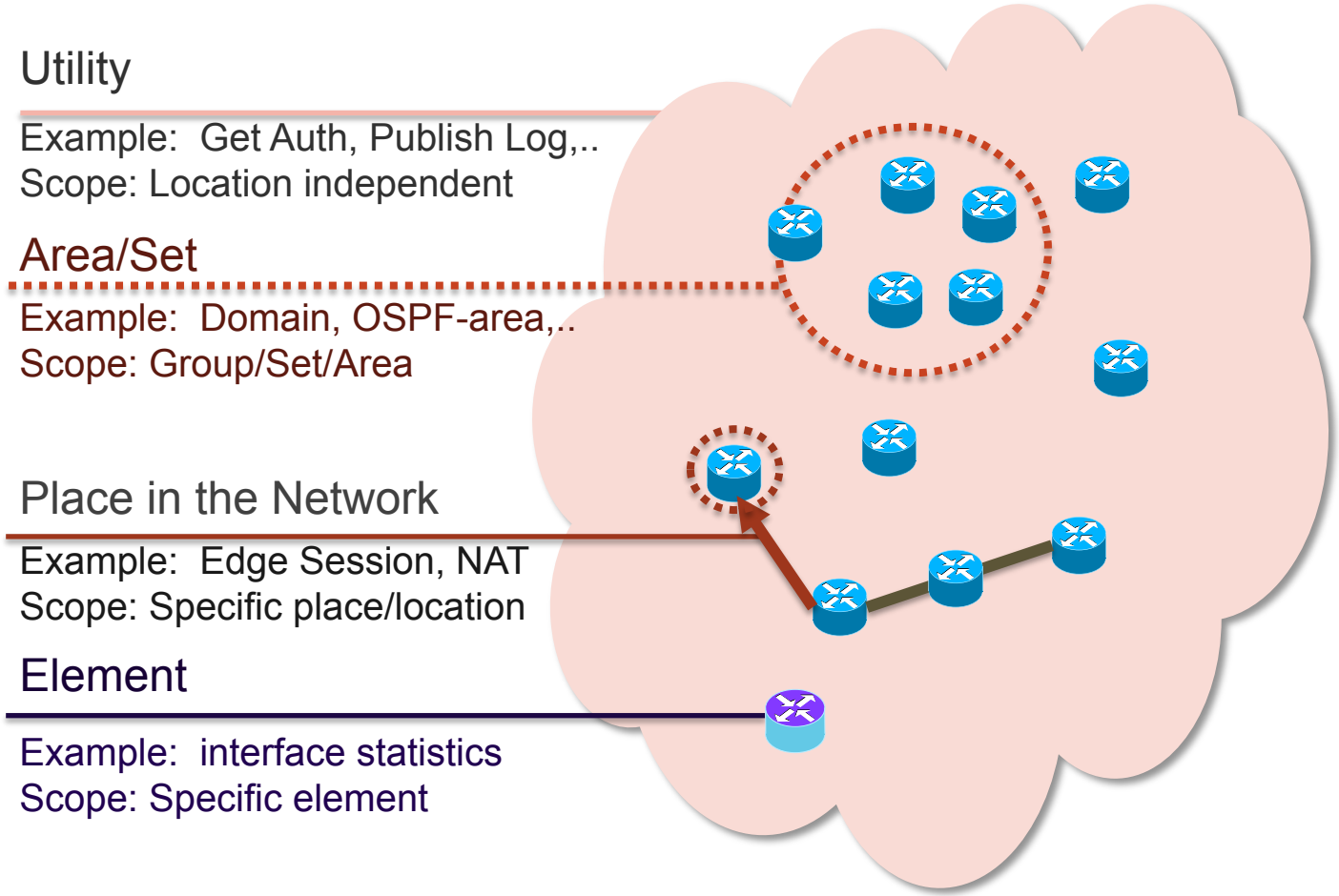


Write Once, Run Anywhere

Not all Networking APIs are created the same

Classes of Networking APIs following their Scope

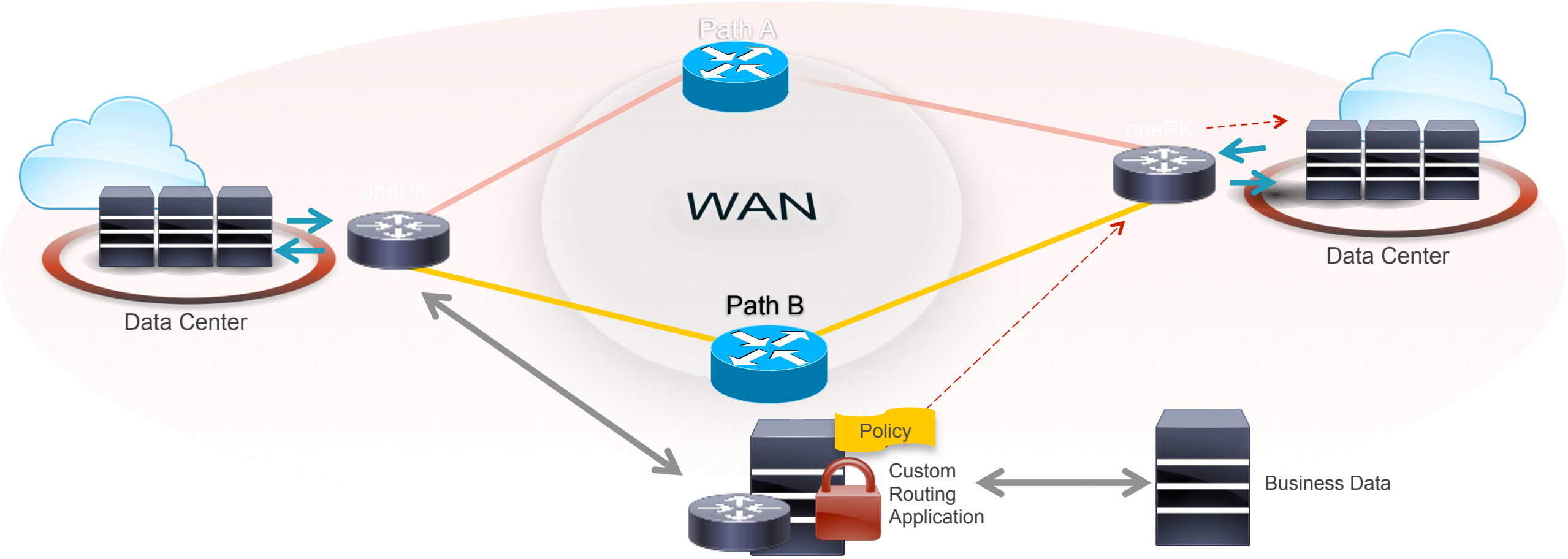
- Classify Networking APIs based on their *scope*
  - API Scopes:
    Location independent; Area;
    Particular place; Specific device
  - Alternate approaches like device/ network/service APIs difficult to associate with use cases
  - Location where an API is hosted can differ from the scope of the API
- Different network planes could implement different flavors of APIs, based on associated abstractions



ıtes. All rights reserved.

Example: Custom Routing

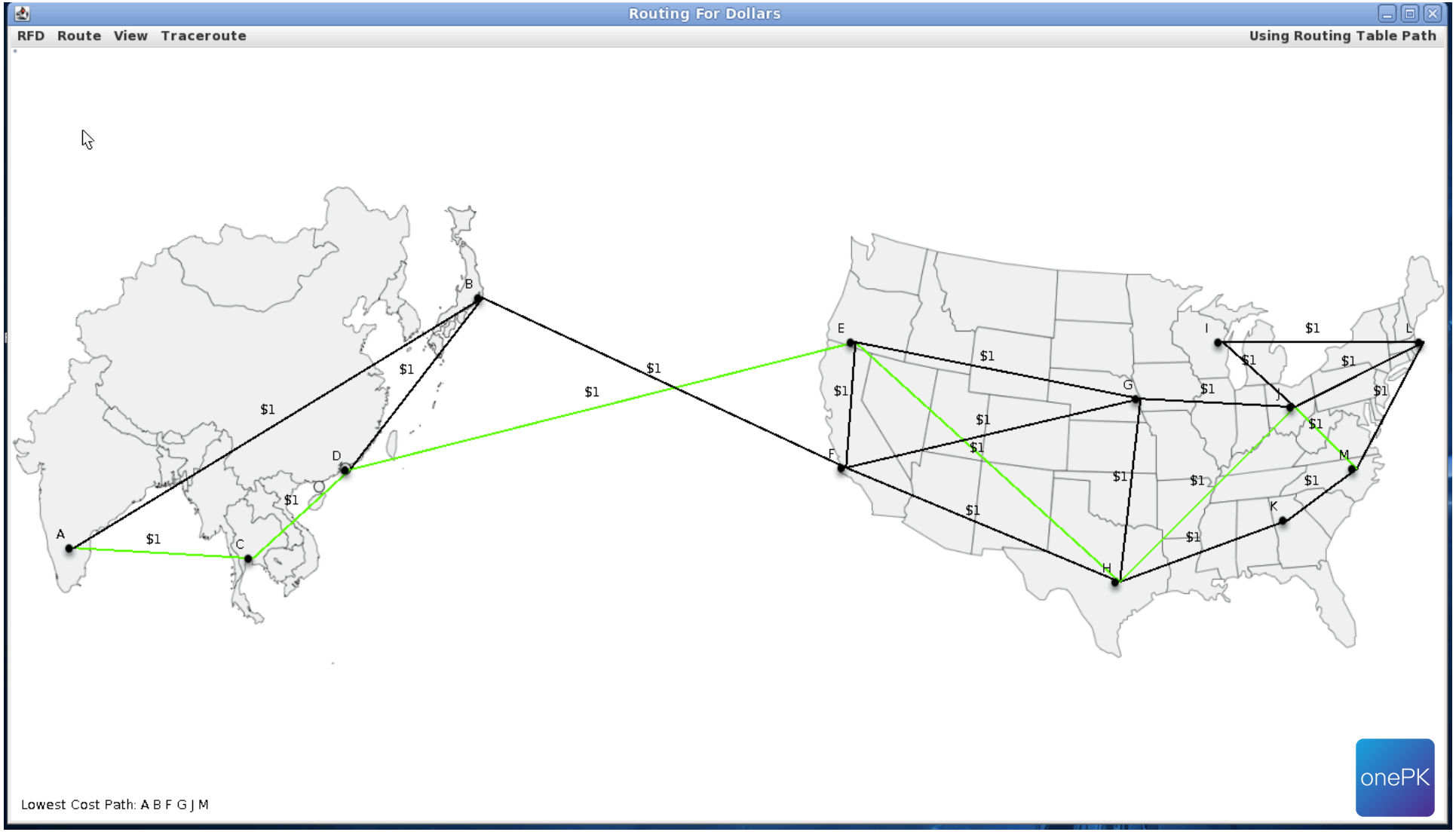
Data Center Traffic Forwarding Based on a Custom Algorithm



Unique Data Forwarding Algorithm Highly Optimized for the Network Operator's Application

Custom Routing

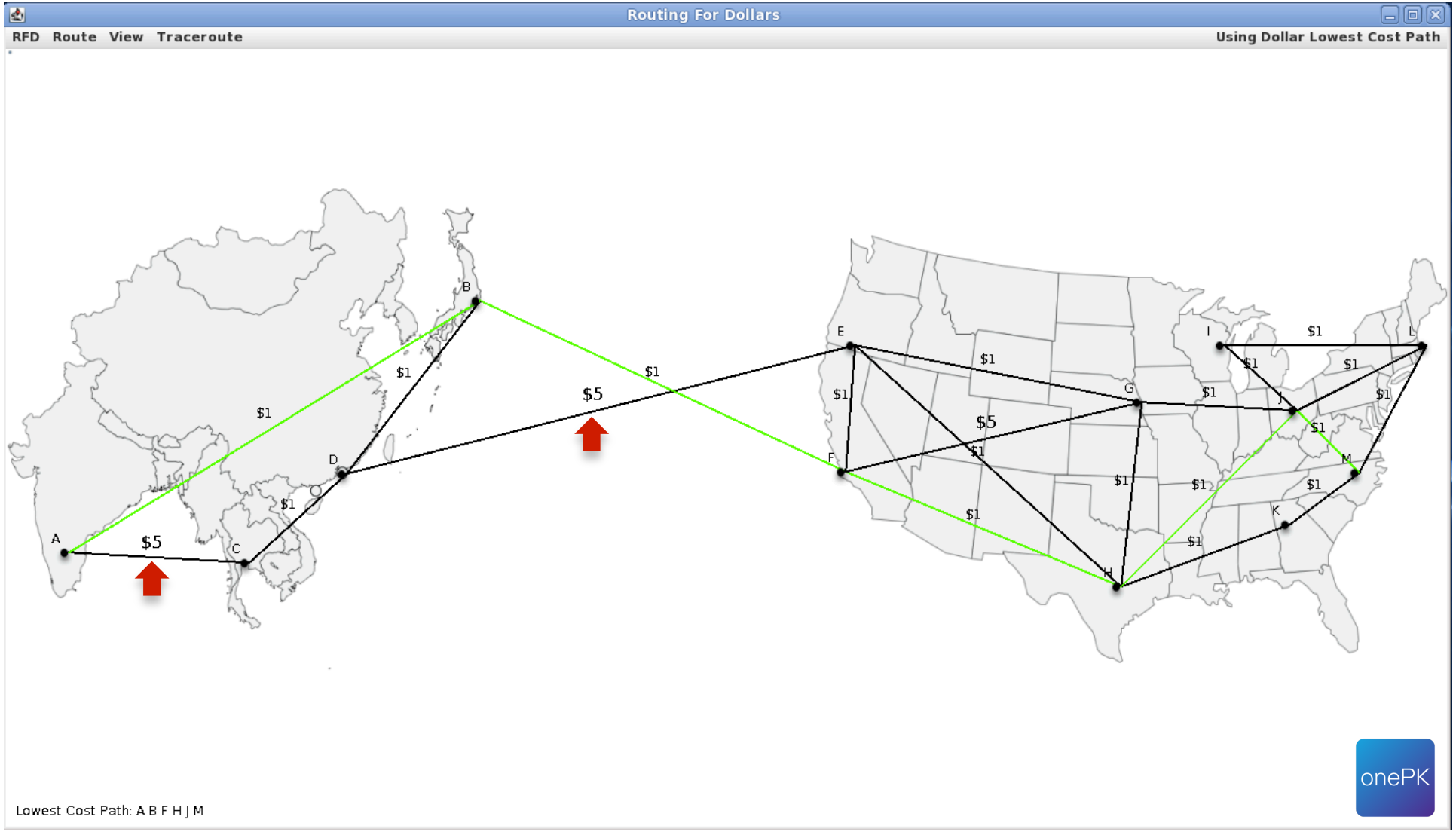
Initial Setup: Default routing using EIGRP



© 2012 Cisco and/or its affiliates. All rights reserved.

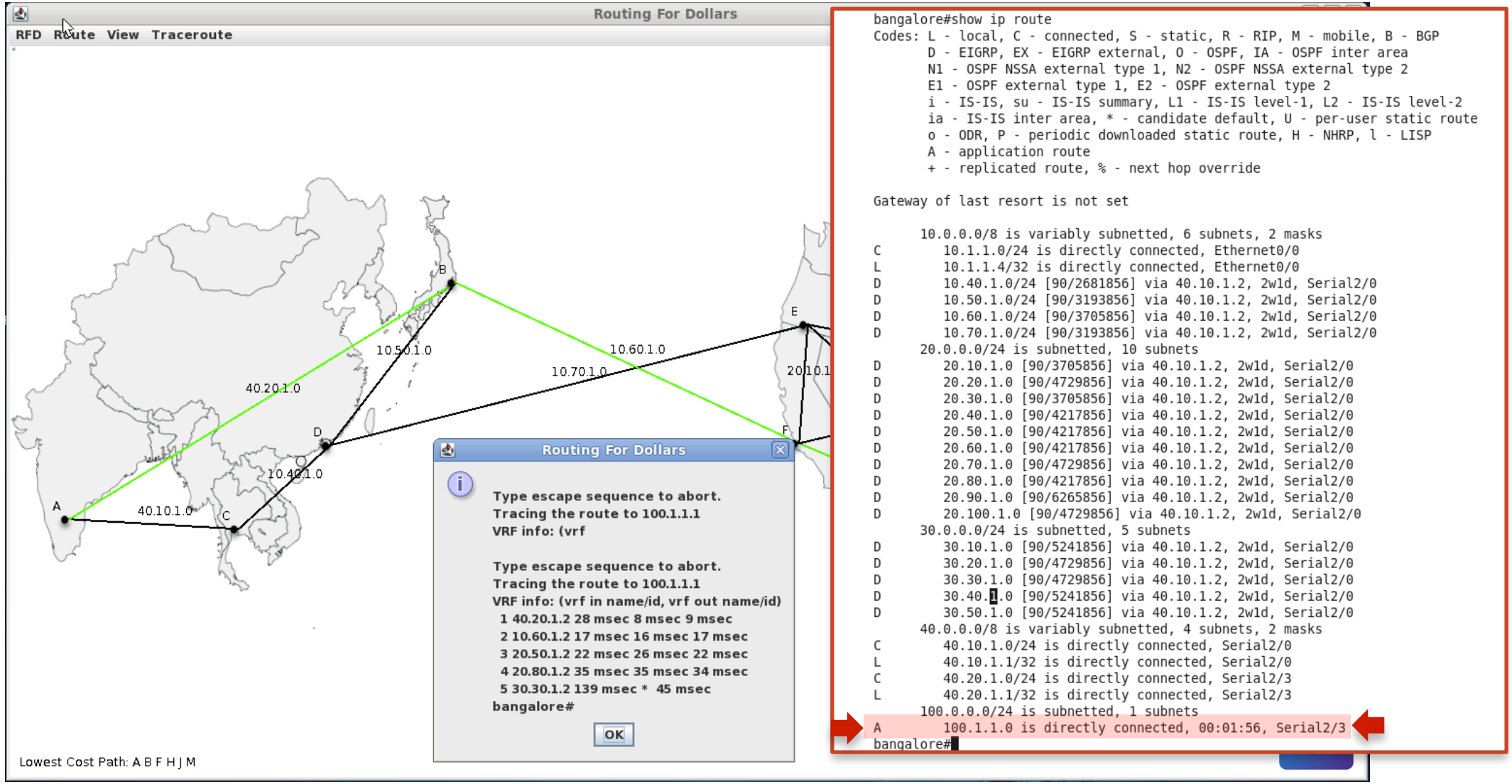
Custom Routing

Routing for Dollars: Application driven routes installed in network



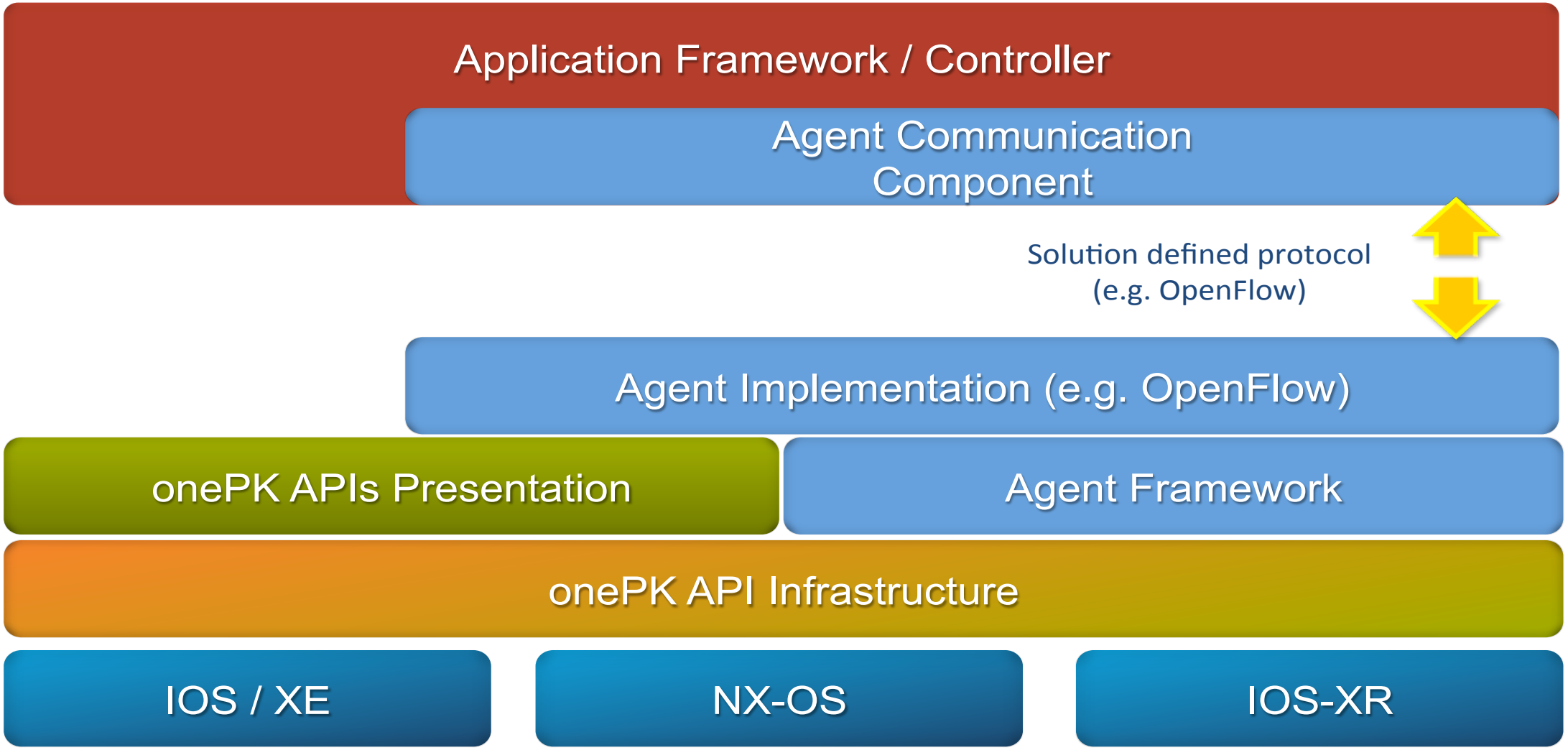
© 2012 Cisco and/or its affiliates. All rights reserved.

Tracing the application installed route – using the developer and element services



© 2012 Cisco and/or its affiliates. All rights reserved.

One PK and Agent Framework (Open Flow,IRS ontop of One PK)



© 2012 Cisco and/or its affiliates. All rights reserved.

Open Network Environment Qualities

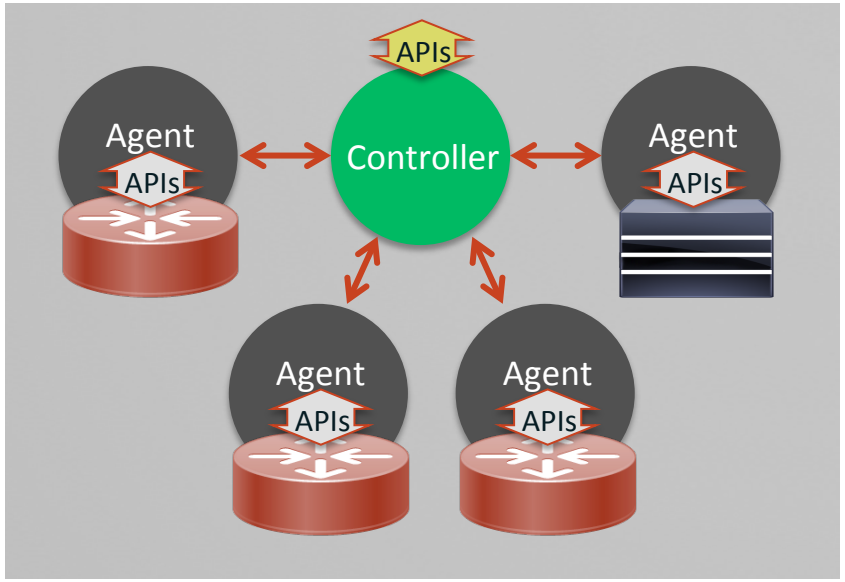
Agents and Controllers:

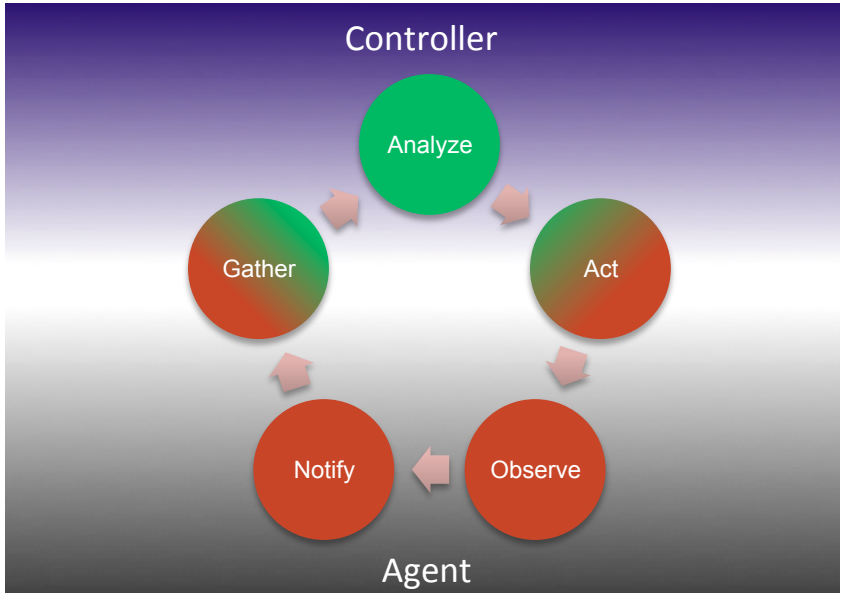
Logically centralized and fully distributed Control

Orchestration: Agents and Controllers

Consolidate State Across Multiple Network Elements

- Some network delivered functionality benefits from logically centralized coordination across multiple network devices
  - Functionality typically domain, task, or customer specific
  - Typically multiple Controller-Agent pairs are combined for a network solution
- Controller
  - Process on a device, interacting with a set of devices using a set of APIs or protocols
  - Offer a control interface/API
- Agent
  - Process or library on a device, leverages device APIs to deliver a task/domain specific function
- Controller-Agent Pairs offer APIs which integrate into the overall Network API suite





© 2012 Cisco and/or its affiliates. All rights reserved.

“Subsidiarity is an organizing principle that matters ought to be handled by the smallest, lowest or least centralized competent authority.”

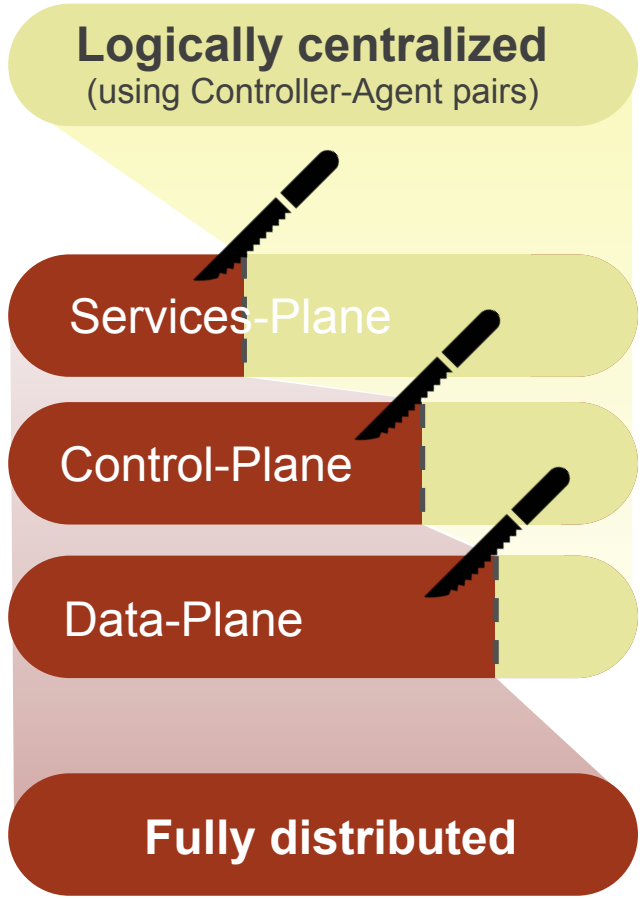
http://en.wikipedia.org/wiki/Subsidiarity

ıtes. All rights reserved.

Deployment Considerations – Applying Subsidiarity to Networking

Exploring the tradeoff between Agents and Controllers – and fully distributed Control

- Control loop requirements differ per function/ service and deployment domain
  - "As loose as possible, as tight as needed"
  - Latency, Scalability, Robustness, Consistency, Availability
  - Different requirements per use case
    - Example:
      Topology for Visualization (Network Management) vs.
      Topology for Path-Computation/Routing
- *How to decide which functionality is well suited a particular control paradigm?*

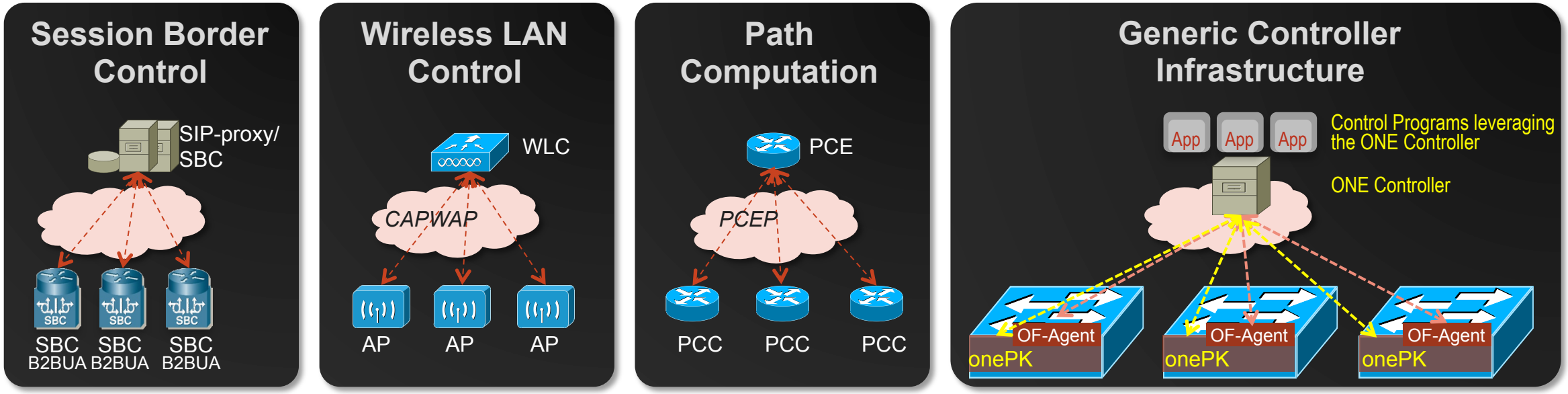


Note: Example only – Not all network planes shown

ıtes. All rights reserved.

Orchestration: Controllers and Agents

Task Specific Solutions and Generic Controller Infrastructure



- Networking already leverages a great breath of Agents and Controllers
  - Current Agent-Controller pairs always serve a specific task (or set of tasks) in a specific domain
- System Design: Trade-off between Agent-Controller and Fully Distributed Control
  - Control loop requirements differ per function/service and deployment domain
  - “As loose as possible, as tight as needed”
  - Latency, Scalability, Robustness, Consistency, Availability

© 2012 Cisco and/or its affiliates. All rights reserved.

Open Network Environment Qualities
Network Infrastructure Virtualization

Network Abstractions support Virtualization

Blurring the lines between physical and virtual entities – networks and services

- Virtual Overlay Networks
  - custom endpoint addressing (e.g. for simple endpoint mobility)
  - custom topologies/segmentation
  - custom service chains
    - Example: vPATH
- Virtual Service Appliances/Gateways
  - VSG, vWAAS

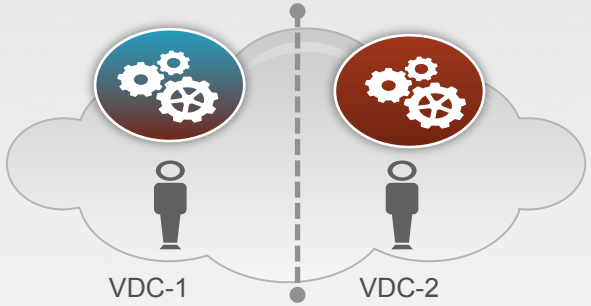
"In computing, **virtualization** is the creation of a virtual (rather than actual) version of something, such as a hardware platform, operating system, storage device, or network resources."

http://en.wikipedia.org/wiki/Virtualization

Common Abstractions and common APIs across physical and virtual network elements

Physical | Virtual | Cloud Journey

PHYSICAL WORKLOAD	VIRTUAL WORKLOAD	CLOUD WORKLOAD
• One app per Server • Static • Manual provisioning	• Many apps per Server • Mobile • Dynamic provisioning	• Multi-tenant per Server • Elastic • Automated Scaling



CONSISTENCY: Policy, Features, Security, Management, Separation of Duties

Switching	**Nexus 7K/5K/3K/2K**	**Nexus 1000V, VM-FEX**
Routing	**ASR, ISR**	**Cloud Services Router (CSR 1000V)**
Services	**WAAS, ASA, NAM**	**vWAAS, VSG, ASA 1000V, vNAM****
Compute	**UCS for Bare Metal**	**UCS for Virtualized Workloads**

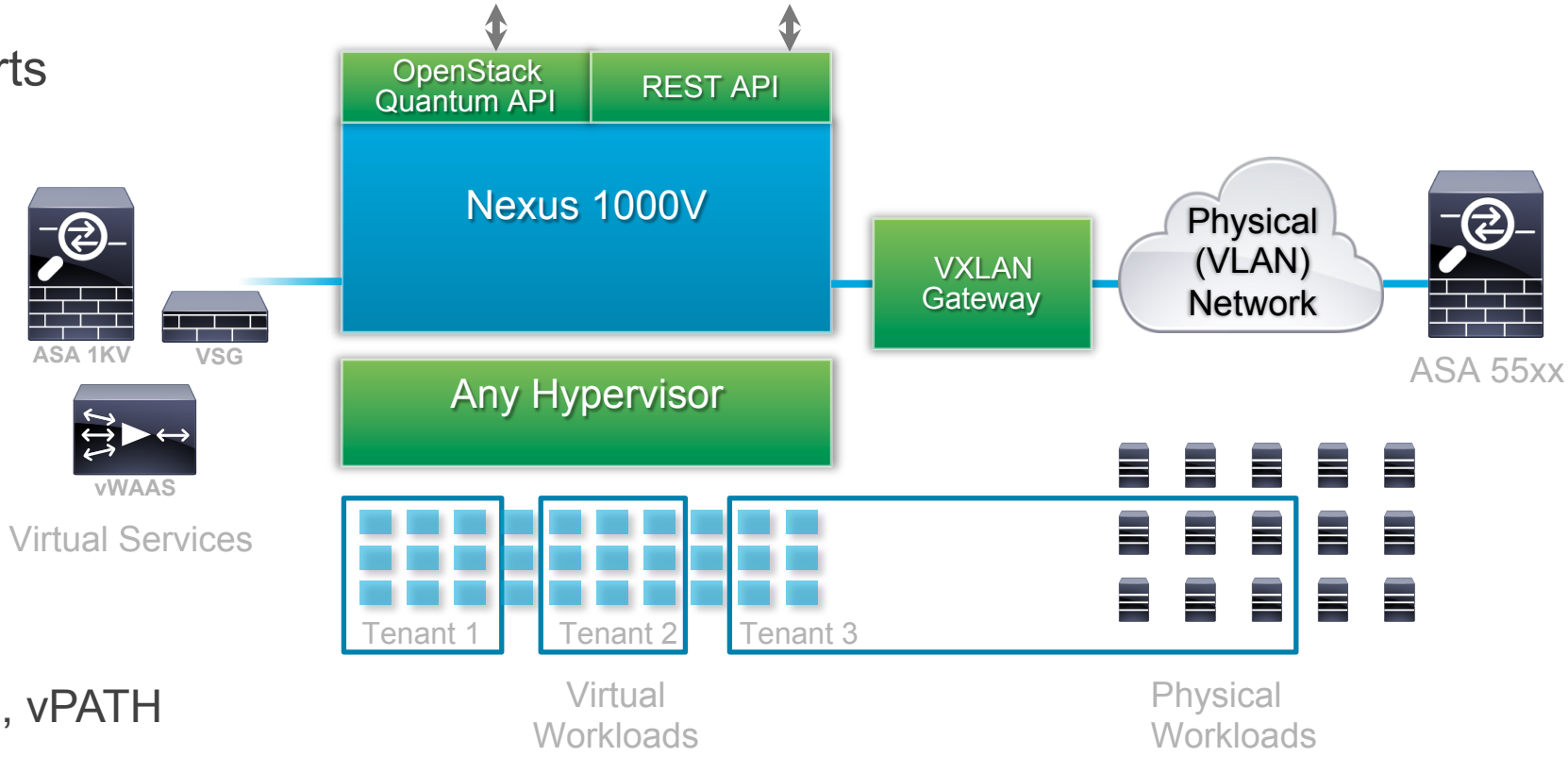
© 2012 Cisco and/or its affiliates. All rights reserved.

** 1H 2013

Virtual Overlay Networks

Example: Virtual Overlay Networks and Services with Nexus 1000V

- Large scale L2 domains:
  Tens of thousands of virtual ports
- Common APIs
  - Incl. OpenStack Quantum API's for orchestration
- Scalable DC segmentation and addressing
  - VXLAN
- Virtual service appliances and service chaining/traffic steering
  - VSG (cloud-ready security), vWAAS (application acceleration), vPATH
- Multi-hypervisor platform support: ESX, Hyper-V, OpenSource Hypervisors
- Physical and Virtual: VXLAN to VLAN Gateway



© 2012 Cisco and/or its affiliates. All rights reserved.

Open Network Environment – Summary

Cisco's approach

- Evolutionary step for networking – evolve the Network Control Plane where needed
- Approach centered around delivering open, programmable environment for real-world use cases
  - No one-size-fits-all
  - Cisco will support Network Virtualization, APIs and Agents/Controllers
  - Joint learning with industry and academia
- Technology-agnostic
  - Not predicated on a particular technology or standard
  - Draw from Cisco technologies and industry standards
- Delivered as incremental functionality
  - Many customers will use hybrid implementations
  - Build upon existing infrastructure with investment protection

For More information

Open Network Environment
www.cisco.com/go/one

One PK
www.cisco.com/go/onepk
www.cisco.com/go/getyourbuildon

ıtes. All rights reserved.

Thank you.