TOMORROW starts here.

Cisco Connect

TECH-SDN-SP: Software Defined Networking v prostředí SP
TECH-SDN-API: Software Defined Networking a aplikační rozhraní

Cisco Connect

společná část – dopolení sekce:

9:00 – 10:20 Část 1: Základy SDN

- SDN definice a historie – Martin Diviš (Cisco) [30m]
- Openflow – Josef Ungerman (Cisco) [25m]
- Openstack – Jiří Chaloupka (Cisco) [25m]

10:40 – 12:00 Část 2: Cisco ONE

- Nexus 1000v – Martin Diviš [20m]
- vPE/Mozart – Jiří Chaloupka [25m]
- OnePK – Martin Diviš [15m]
- OpenDaylight, XNC – Josef Ungerman [20m]

TECH-SDN-SP: Software Defined Networking v prostředí SP

odpolení sekce

13:00 – 14:30 Část 3: SP SDN

- SDN WAN (PCEP, BGP-LS, BGP-FS, nLight, Segment Routing) – David Jakl [30m]
- Netconf, RESTconf, Yang – Martin Kramoliš [25m]
- NfV koncepce a využití – Martin Slinták [35m]

14:45 – 17:00 Část 4: SP Demo

- WAN Orchestrace, WAE demo – Stanislav Kraus [30m]
- Cisco Modeling Labs (VIRL) – Stanislav Kraus [10m]
- Představení demo – Cisco Team [15m]

16:00 Demo Fair

- 1. OpenDaylight – Josef Ungerman
- 2. OpenStack – Jiří Chaloupka
- 3. Netconf/Yang – Martin Kramoliš
- 4. Cisco Modeling Labs – Stanislav Kraus
- 5. Segment Routing – David Jakl
- 6. vCPE NfV – Martin Slinták

TECH-SDN-API: Software Defined Networking a aplikační rozhraní

odpolední sekce:

13:00 – 14:30 Část 3: API v rodině Nexus

- Úvod do API datacentrových platforem – Martin Diviš
- Rozhraní Netconf na Nexusech, praktické použití – Jiří Novák (Netsystem)

14:45 – 17:00 Část 4: API v rodině Nexus

- Python – zbraň admina na Nexusech – Martin Diviš
- Nexus 9000 a NX-API – Martin Diviš
- Nexus 1000V a REST API – Martin Diviš
- Guestshell – nativní aplikace na Nexusech – Martin Diviš
- Getting Started with OnePK v DC – Martin Diviš

SDN Introduction

Martin Divis, CSE, mdivis@cisco.com

SDN → **S**till **D**on’t k**N**ow

SDN → **S**tanford **D**efined **N**etworking

SDN → **S**till **D**oes **N**othing

Industry trends

Cloud (2010)

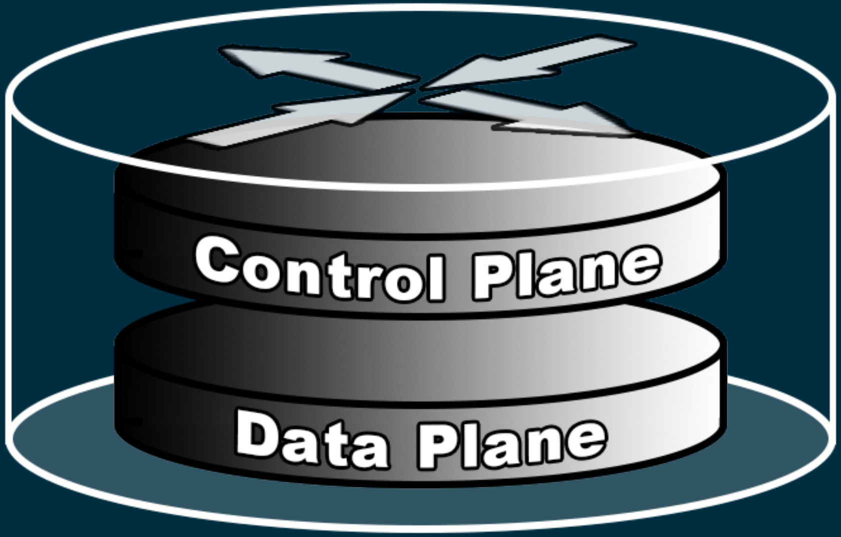
OpenFlow Networking (Stanford clean slate) (2011)

Software Defined Networking (2012)

Software Defined Datacenters (2013)

OPEN DAYLIGHT www.opendaylight.org Open Daylight Project (2013)

Policy Infrastructure Controller (2014)

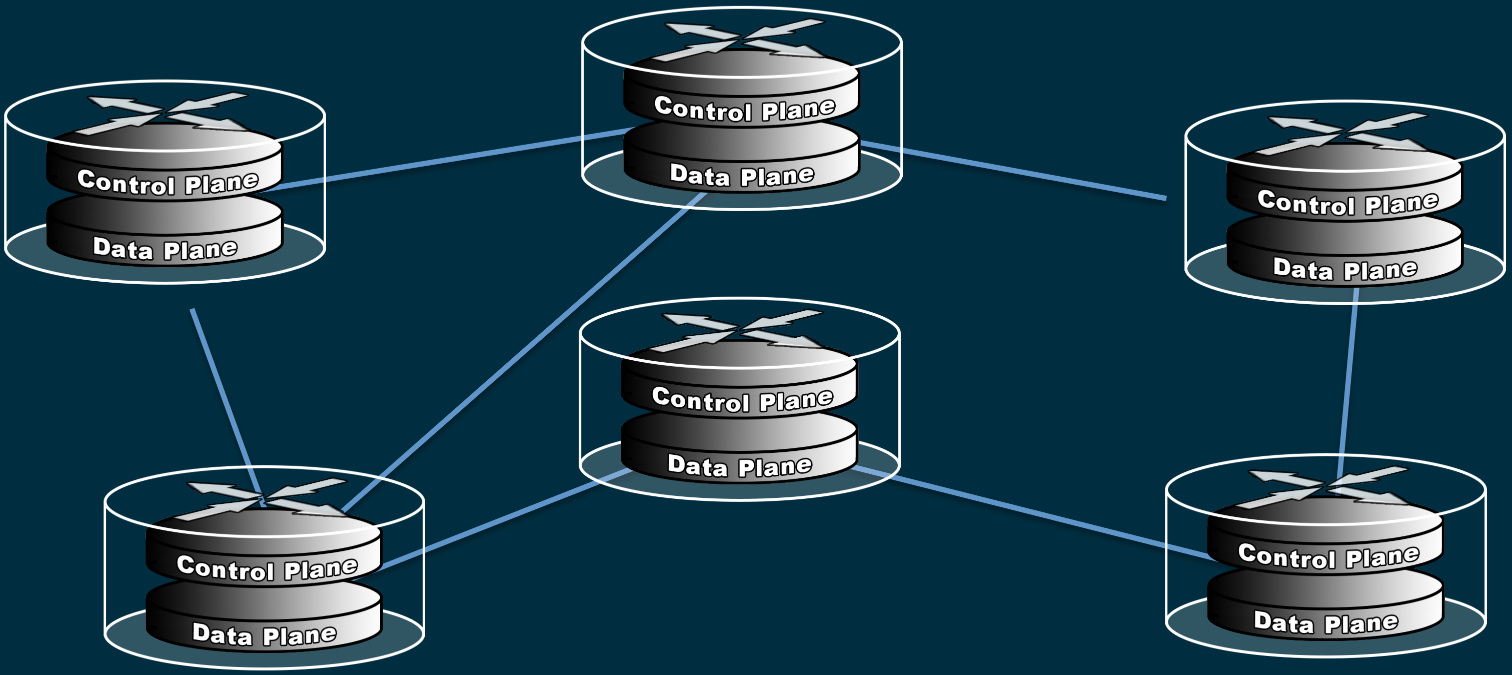


Control and Data Plane resides within Physical Device

Processing Plane	Where it runs	How fast these processes run	Type of processes performed
Control Plane	Switch CPU	In the order of thousands of packets per second	Routing protocols (i.e. OSPF, IS-IS, BGP), Spanning Tree, SYSLOG, AAA (Authentication Authorization Accounting), NDE (Netflow Data Export), CLI (Command Line interface), SNMP
Data Plane	Dedicated Hardware ASIC’s	Millions or Billions of packets per second	Layer 2 switching, Layer 3 (IPv4 | IPv6) switching, MPLS forwarding, VRF Forwarding, QOS (Quality of Service) Marking, Classification, Policing, Netflow flow collection, Security Access Control Lists

Control Plane and Data Plane

Two fundamental terms to begin understanding the concepts around SDN



Devices are independent, intelligence is distributed

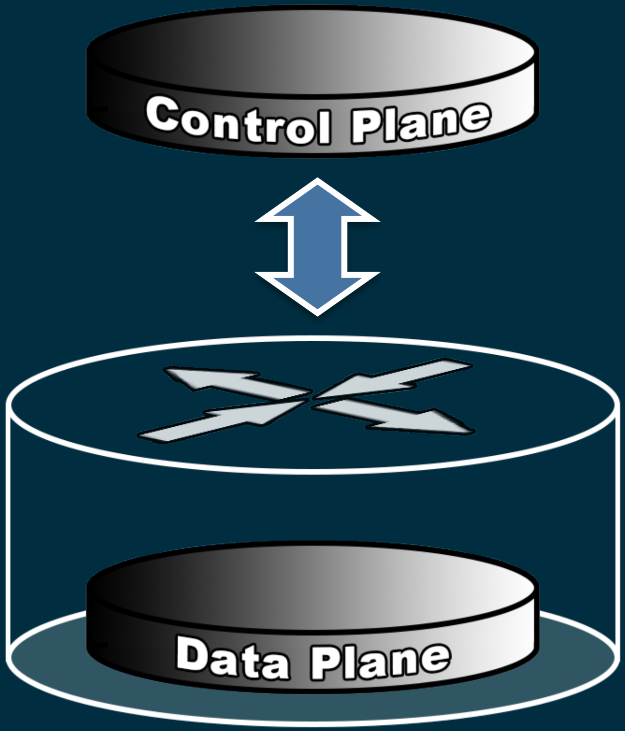
What is SDN?

per ONF definition

https://www.opennetworking.org/sdn-resources/sdn-definition

Software defined networking (SDN) definition:

The physical separation of the network control plane from the forwarding plane,
and where a control plane controls several devices.



In other words…

In the SDN paradigm, not all processing happens inside the same device

"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"

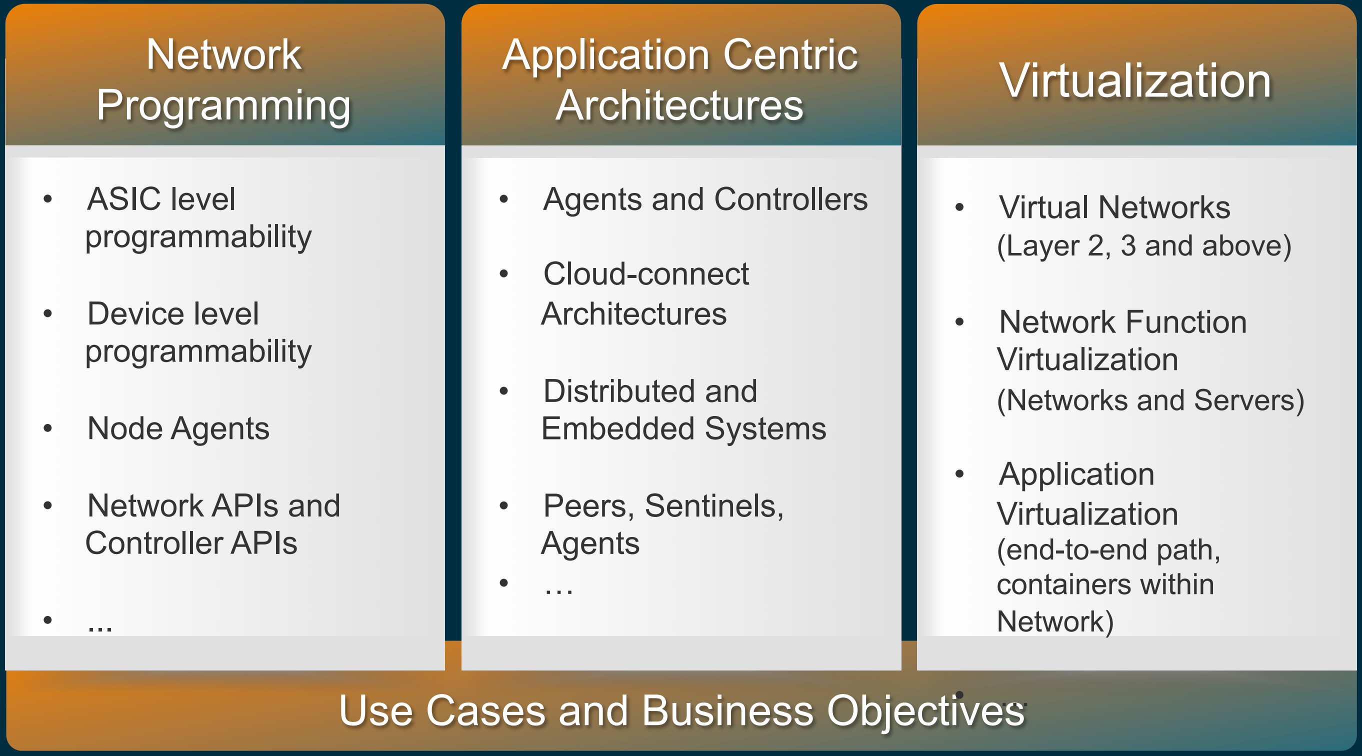
"A way to distribute policy/intent, e.g. for DDoS prevention, in the network"

Within SDN concepts, We Need to Define

- How to access the devices
- How to program/configure/inspect the devices
- Level of autonomness of a device and SDN coexistence
- Featues and functions of a controller
- Controller Northbound API

Proliferation of 3 Main Concepts

- Common across SDN approaches
- Enabling capabilities
- Proliferating across domains



Network Programming

- ASIC level programmability
- Device level programmability
- Node Agents
- Network APIs and Controller APIs
- ...

Application Centric Architectures

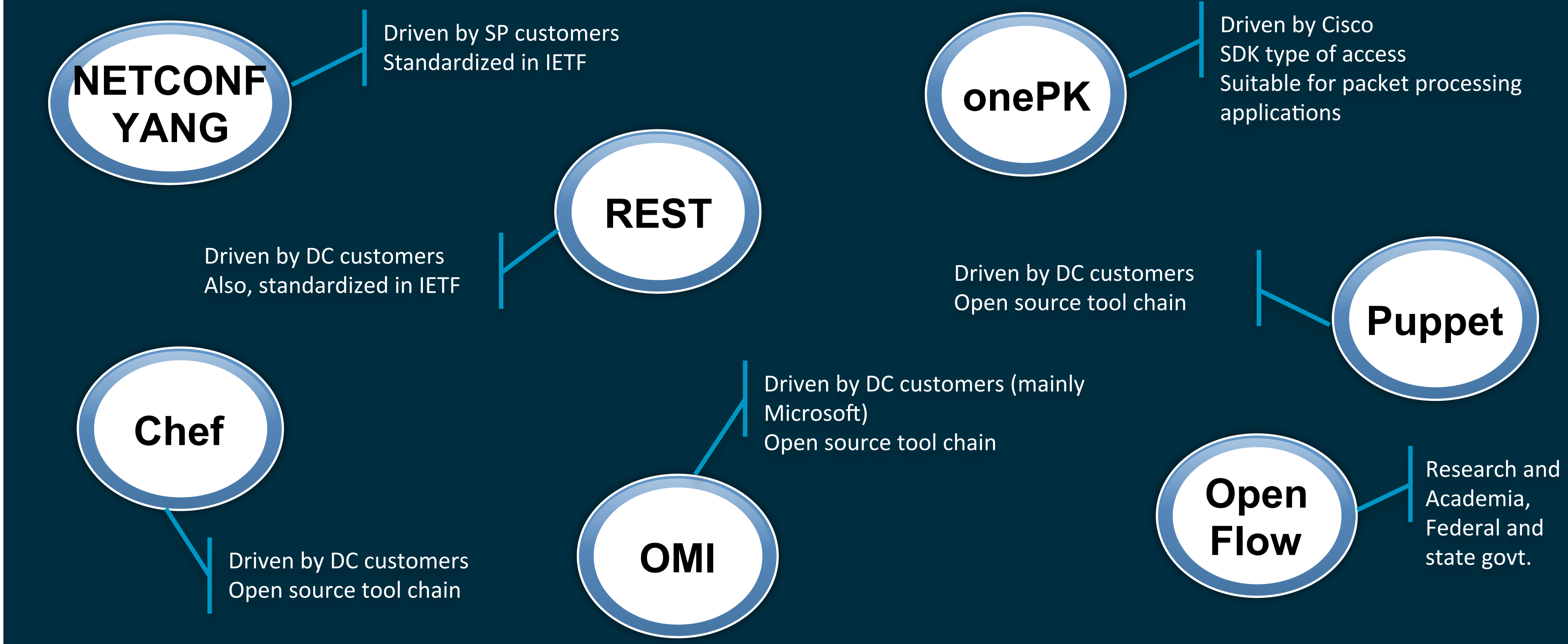
- Agents and Controllers
- Cloud-connect Architectures
- Distributed and Embedded Systems
- Peers, Sentinels, Agents
- …

Virtualization

- Virtual Networks (Layer 2, 3 and above)
- Network Function Virtualization (Networks and Servers)
- Application Virtualization (end-to-end path, containers within Network)

Use Cases and Business Objectives

One Size does not fit ALL



NETCONF YANG

Driven by SP customers
Standardized in IETF

onePK

Driven by Cisco
SDK type of access
Suitable for packet processing applications

REST

Driven by DC customers
Also, standardized in IETF

Puppet

Driven by DC customers
Open source tool chain

Chef

Driven by DC customers
Open source tool chain

OMI

Driven by DC customers (mainly Microsoft)
Open source tool chain

Open Flow

Research and Academia, Federal and state govt.

Extremely resource intensive to scale without a common data model

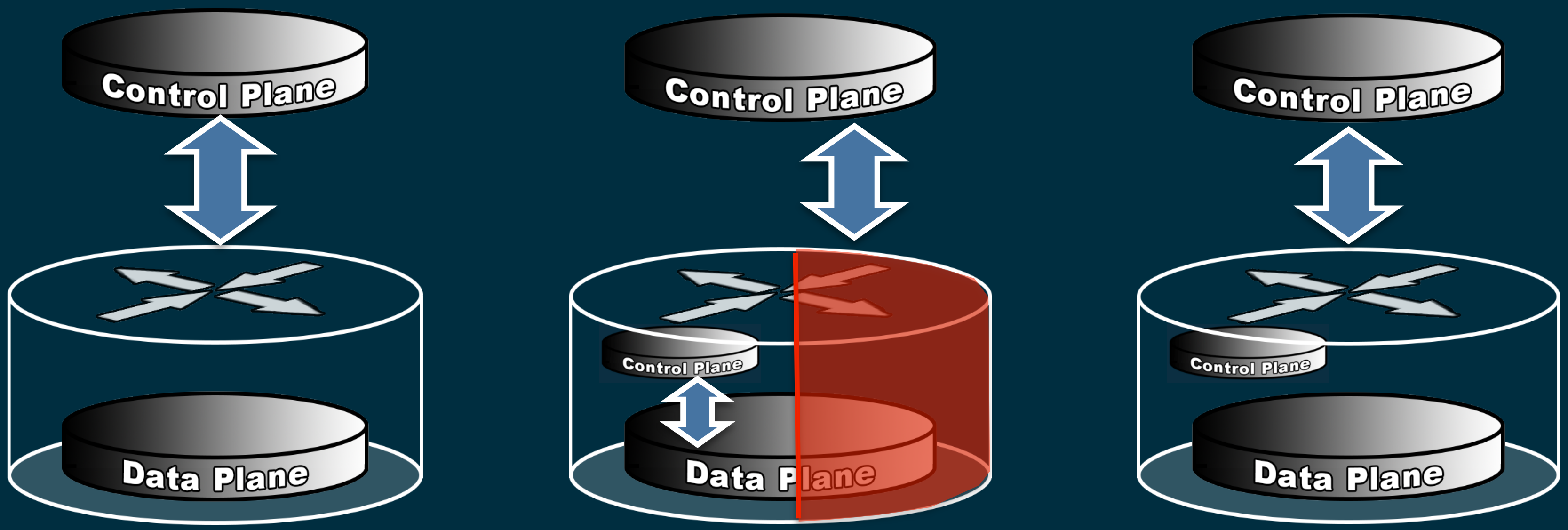
Device autonomy model

Hybrid model

Centralized Control

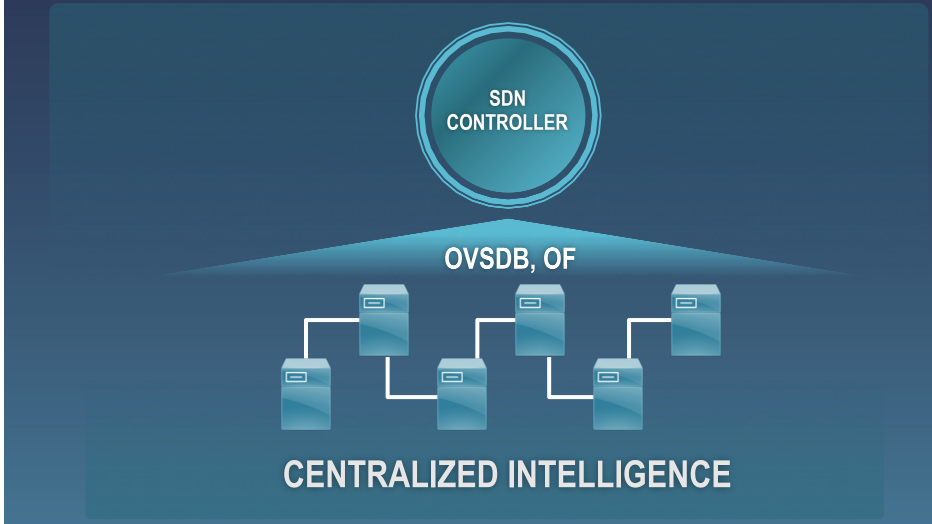
Dedicated resources

Integrated resources



Cisco ACI vs Traditional SDN

TRADITIONAL SDN MODEL



SDN CONTROLLER

OVSDB, OF

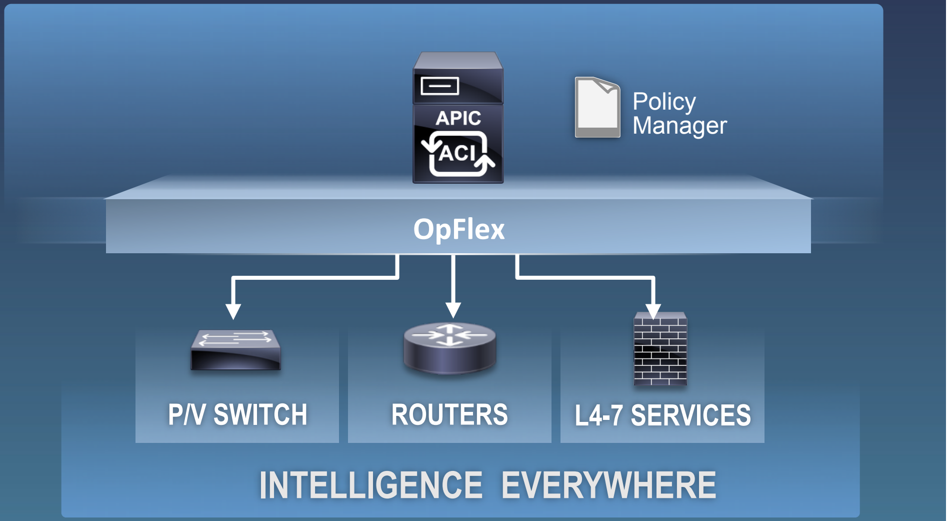
CENTRALIZED INTELLIGENCE

Different Policy Models for DC, WAN, Access

Separate Policy / Mgmt for Physical and Virtual Infrastructure

Inconsistent Security

ACI MODEL



APIC

ACI

Policy Manager

OpFlex

P/V SWITCH

ROUTERS

L4-7 SERVICES

INTELLIGENCE EVERYWHERE

Common Policy Model across DC, Cloud, WAN, Access

Same for both Physical and Virtual infrastructure

End-to-End Consistent Security

OpFlex – policy sharing standard proposal

- http://tools.ietf.org/html/draft-smith-opflex-00
- Cisco, Citrix, IBM, Microsoft, Sungard
- Policy definition, repository & exchange
- Managed object status monitoring
- JSON documents for exchange
- Simple RPC for interation

```
"name": <URI>
"properties": [{"name":<string>, "data": <string>}*]
"children": [<mo>*]
"statistics": [<mo>*]
"from_relations": [<mo>*]
"to_relations": [<mo>*]
"faults": [<mo>*]
"health": [<mo>*]
```

Prosíme, ohodnoťte tuto přednášku

Děkujeme

CISCO™