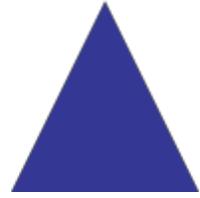


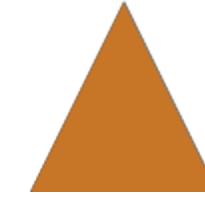
OPEN HARDWARE.



OPEN SOFTWARE.



OPEN FUTURE.



Exploring Existing Open Source Network Operating Systems & APIs

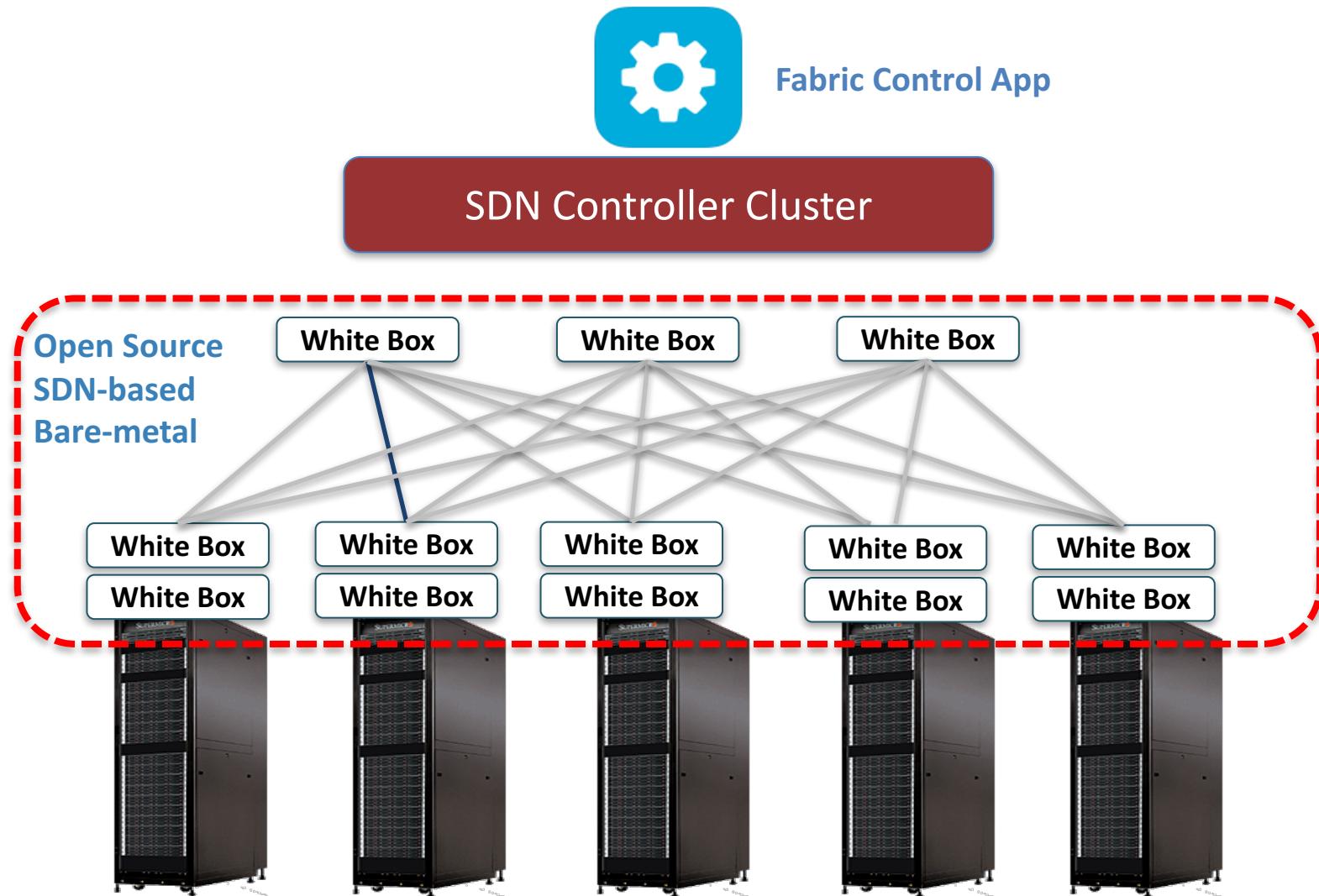
Ahmed M. Abdelmoniem

Senior Researcher, FNTL, Huawei

Outline

1. Open Network Linux (ONL)
2. Software for Open Networking in the Cloud (SONiC)
3. Switch Abstraction Interface (SAI)
4. Open Network Switch Library (OpenNSL) &
OpenFlow DataPath Abstraction (OF-DPA)

Open-Source Leaf-Spine Fabric



ON.LAB



Edge-core
NETWORKS

Why Bare Metal Switches?

Cost

400,000 servers
20,000 switches
\$5k vendor switch = \$100M
\$1k bare-metal switch = \$20M
Savings in 10 data centers = \$800M

Control

Tailor network to applications
Proprietary behavior
Quickly debug
No vendor lock-in

ONL, SwitchLight,
PicOS, FBOSS,
SONIC,
OpenSwitch,
Cumulus, Pigeon,
FastPath,
SnapRoute

Optional Remote API

Accton
Delta
Dell
Wedge
Quanta
Interface Masters
Celestica

OF-DPA, OpenNSL, SAI, P4

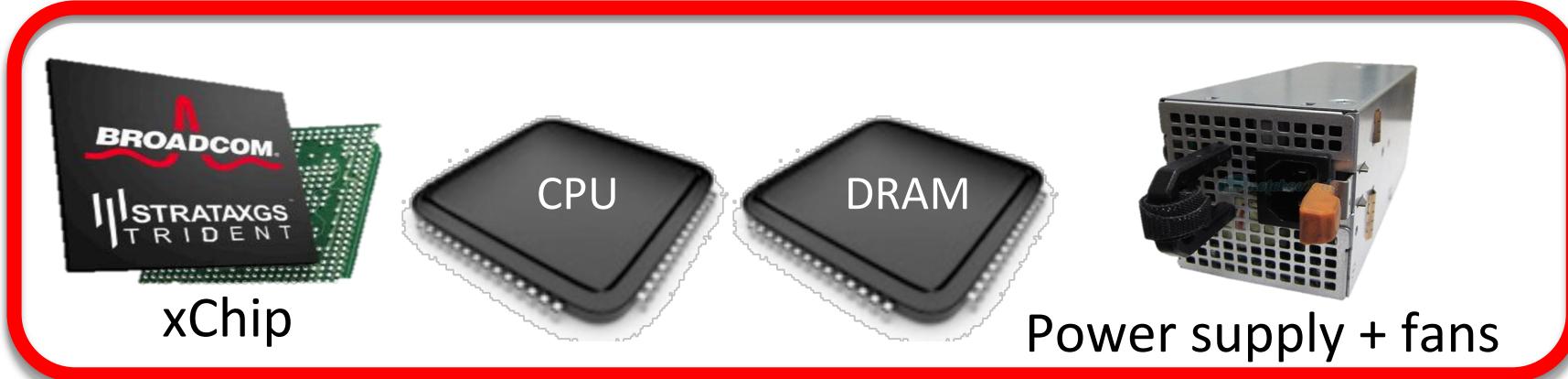
API/Driver

Boot Loader

ONIE



Linux



Open Network Linux (ONL)

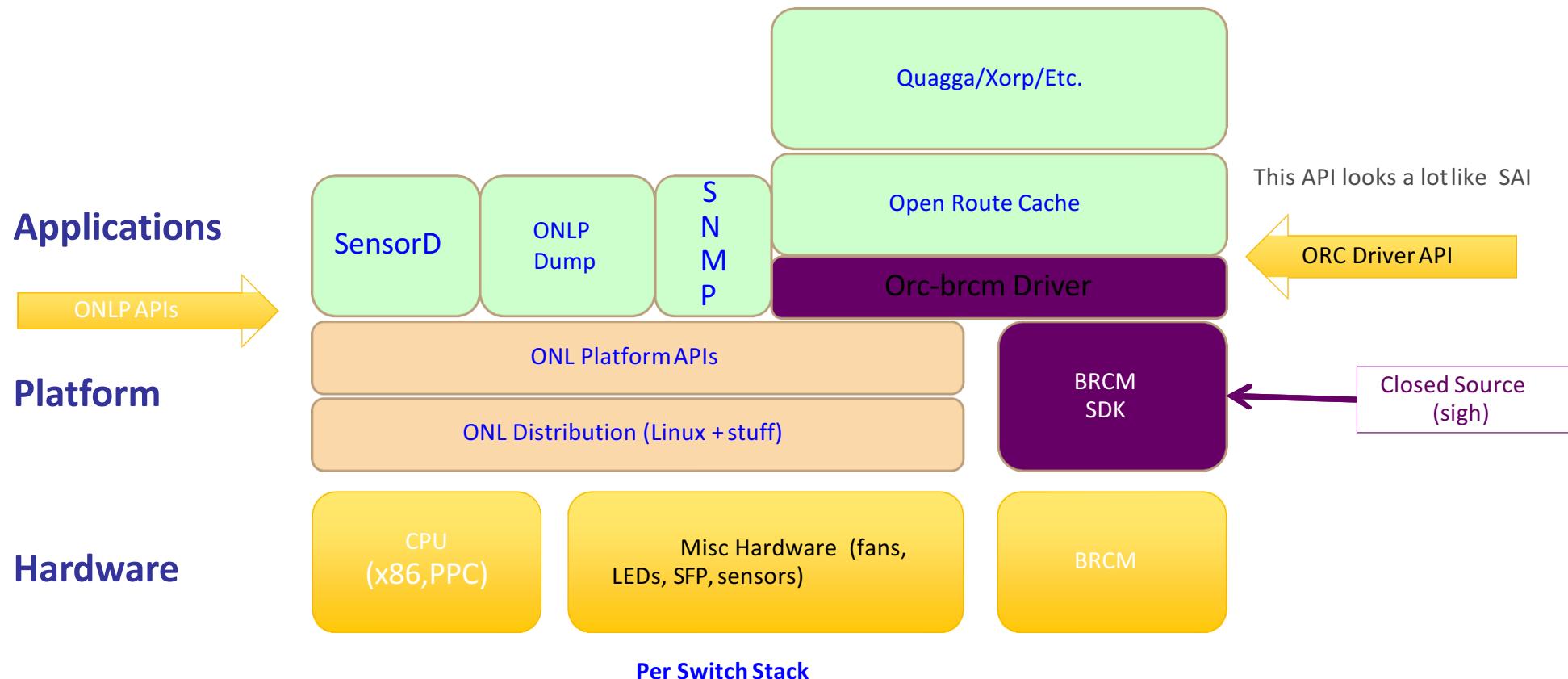
What is Open Network Linux?

- Reference NOS for the Open Compute Project (OCP)
- A collection of software packages, utilities, drivers, and abstractions to run on OCP Switch hardware
- i.e., a “NOS” that ONIE would install
- Why not use an existing Linux distribution?
 - Does build on existing distribution – Debian Wheezy
 - Need to create ONIE installers for many platforms
 - Need to manage switch-specific hardware (e.g.,SFPs, FANs, Sensors)
 - Switches are very similar to servers, but not quite

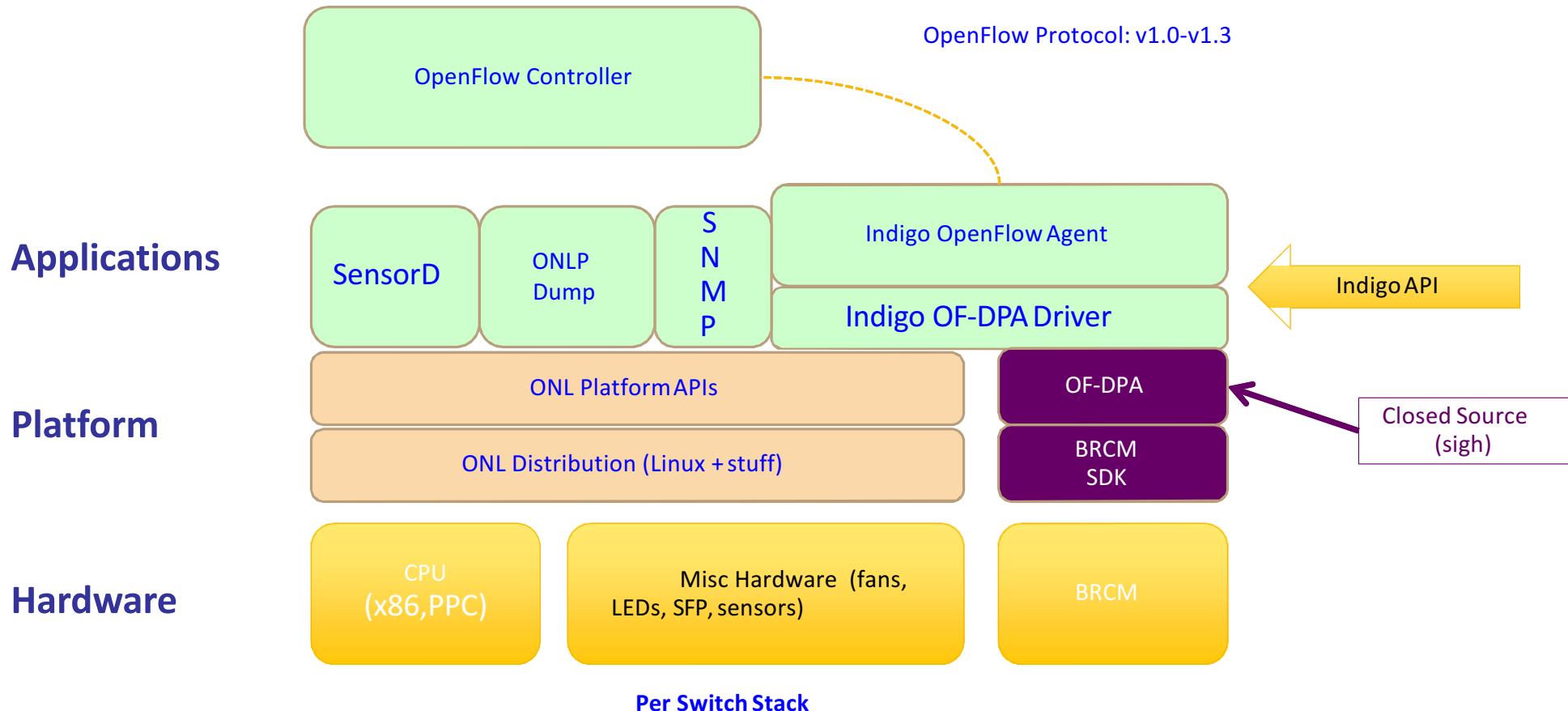
Why Use ONL?

- Help ecosystem focus on innovation
 - Many annoying software details to run an OCP switch
 - Building platform drivers not high value asset; should be common
- Enables a reference NOS implementation
 - Hardware without software is not useful
 - Package up details and best practices into one place
- Help bootstrap the Open ecosystem and OCP adoption
 - Allows commercial companies and DIY-folks to build OCP-based products faster

Example: ONL + Open Route Cache



Example 2: ONL + OpenFlow Agent



ONL Early Contributors

Adds
Forwarding
Agents



Open
Network
Linux



Additional
Drivers

Provides
Platform
Drivers

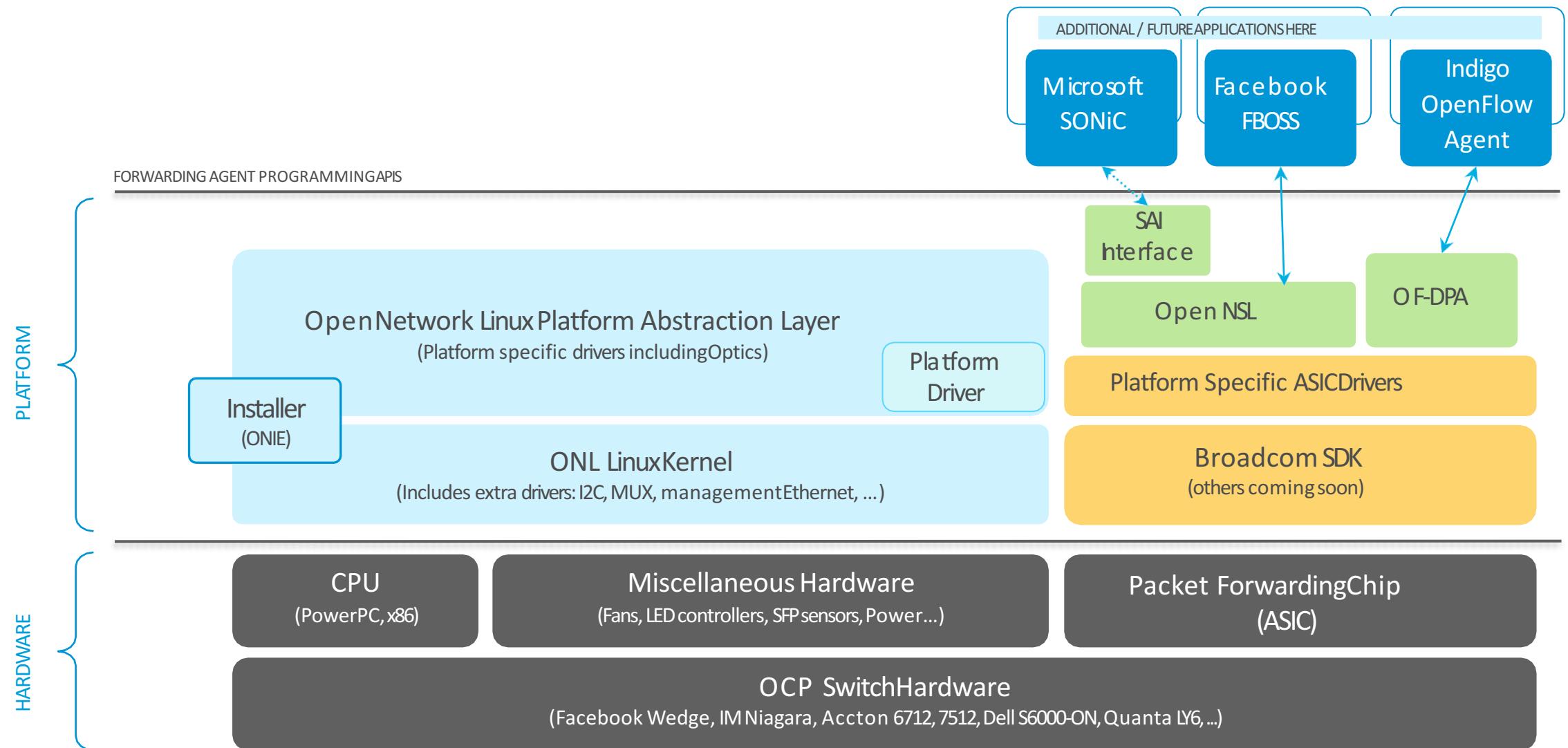


Interface Masters
TECHNOLOGIES
Innovative Network Solutions



OTHERS

ONL: 3+ FORWARDING AGENTS; 20+ PLATFORMS



EXAMPLE OF INDUSTRY USE CASES



Project Atrium with Indigo OpenFlow Agent



FBOSS – Built on Broadcom's OpenNSL Library

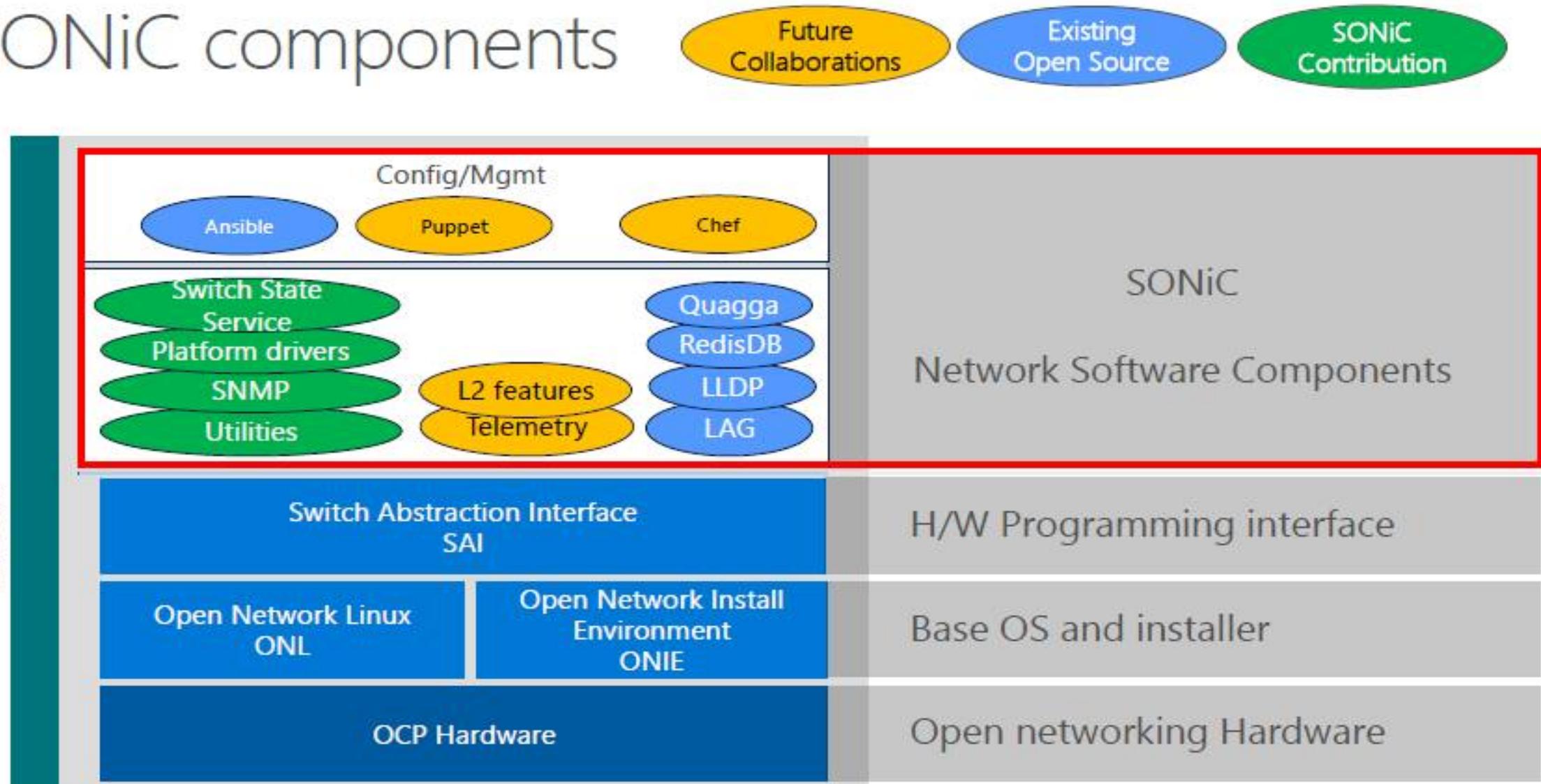


L3 Routing using GoBGP + ORC

Software for Open Networking in the Cloud (SONiC)

SONiC Architecture

SONiC components



Existing SONiCEcosystem

Application,
management,
tools

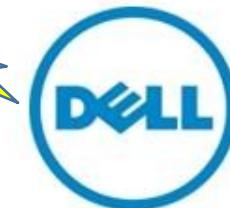
Switch Platform

Merchant
Silicon

Monitoring, Management, Deployment Tools, Cutting Edge SDN



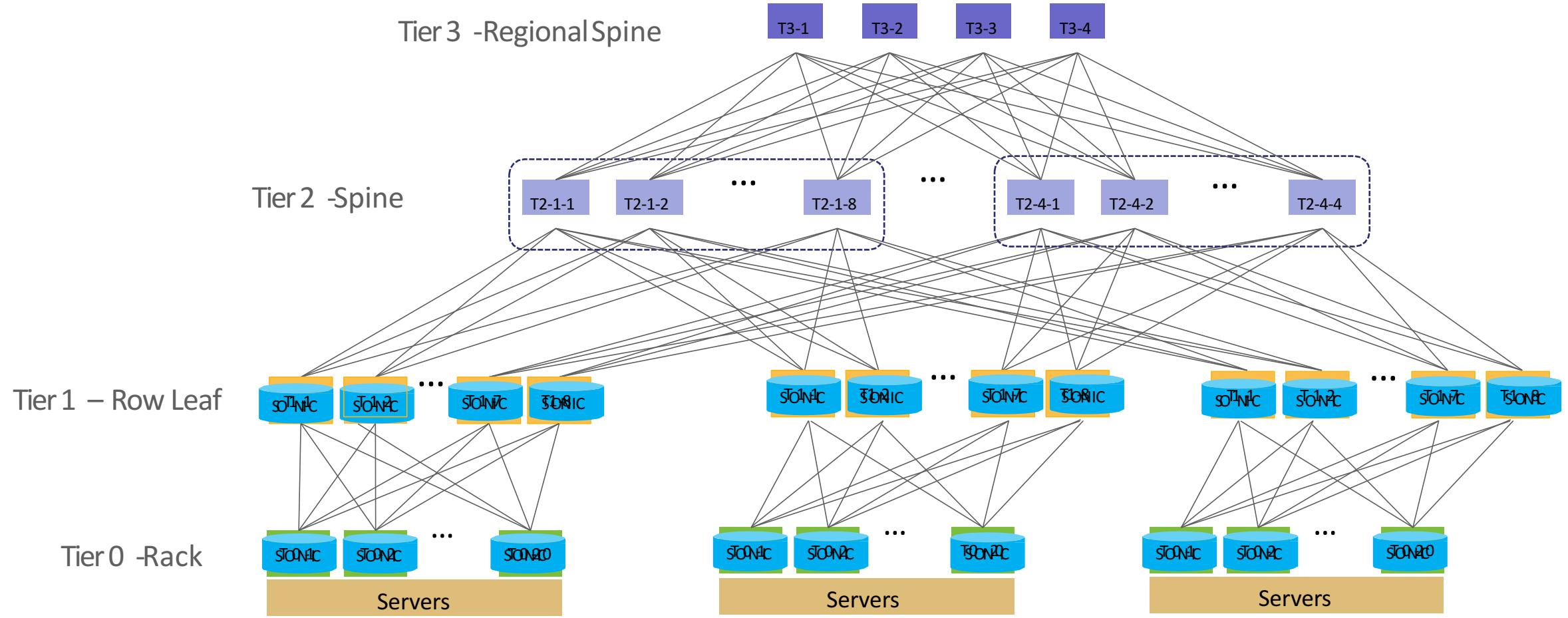
SONiC



Switch Abstraction Interface (SAI)



SONiC is the NOS Powering Tier 0 & 1 in Microsoft Azure



Large-Scale Network for Cloud Services

Cloud-scale creates the following requirements on switch firmware:

- Disaggregate firmware features
 - Minimize the baggage – switch firmware loaded with only features needed
 - Operators can select the best building blocks for the job
- Hitless upgrade so updates don't cause customer impact
 - Fix bugs or apply security patches in hours
 - Rollout features in days
- Development environment
 - Realistic emulation environment

SONiC Containerization

Container Strength

- Clean isolation
- Easy deployment
- Transactional
- Run universal

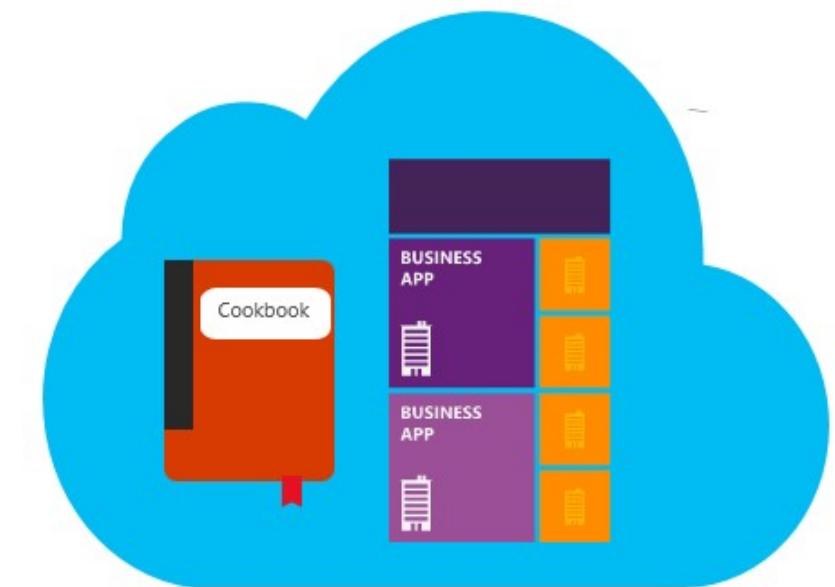
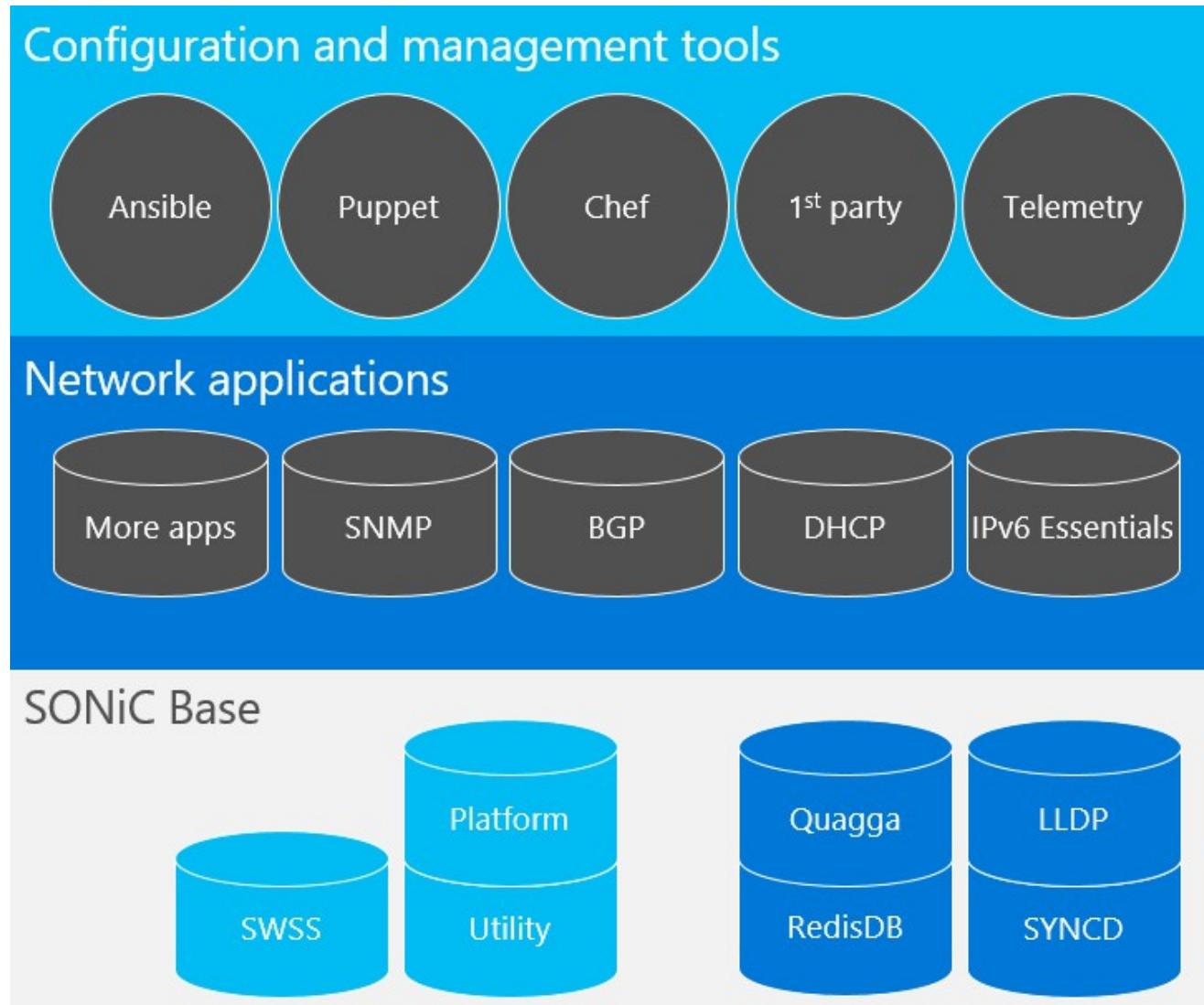


SONiC Benefits

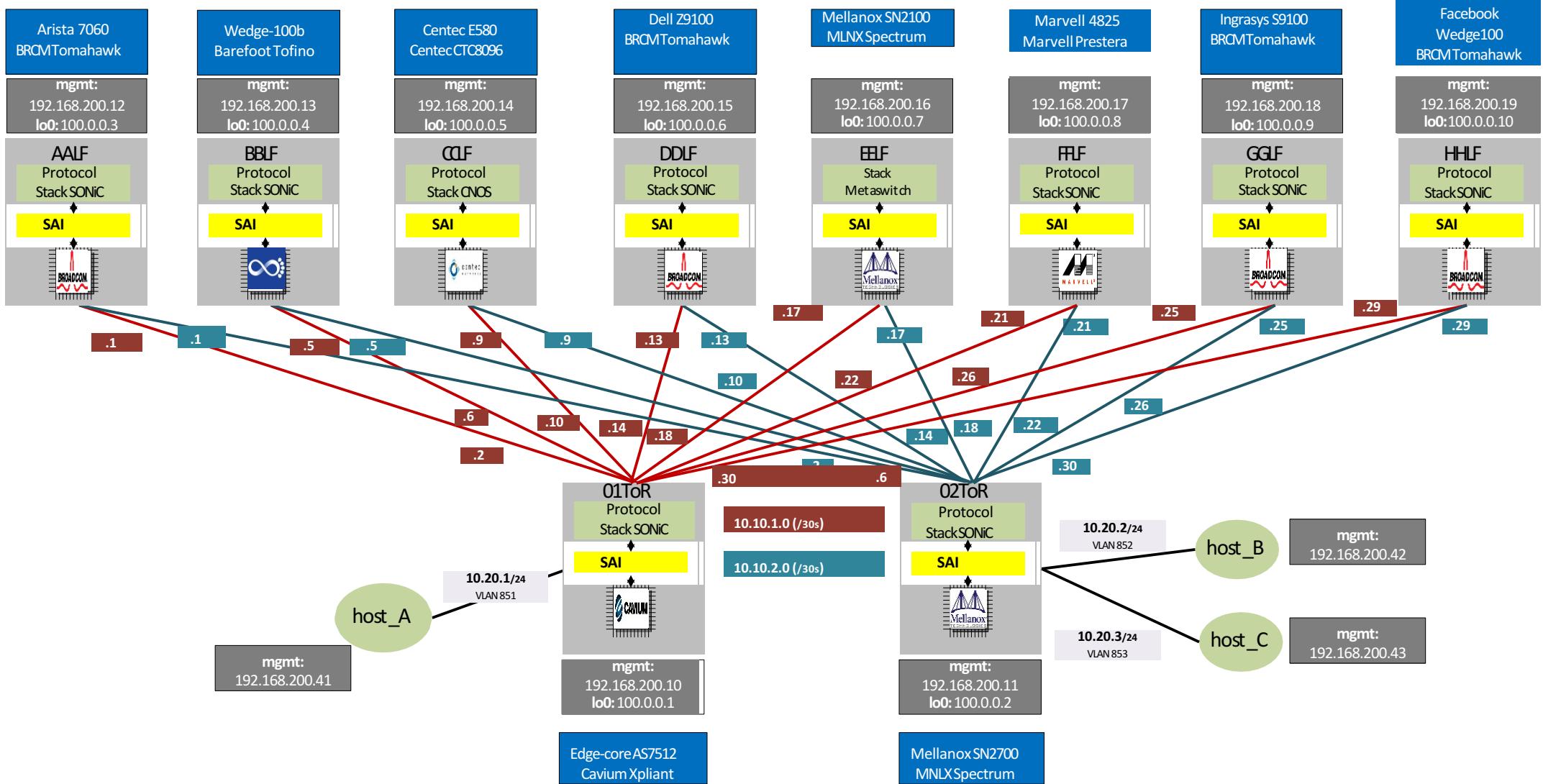
- Serviceability
- Extensibility
- Development agility
- Cross-platform

What Would Containerized SONiC Enable?

SONiC – Containerized

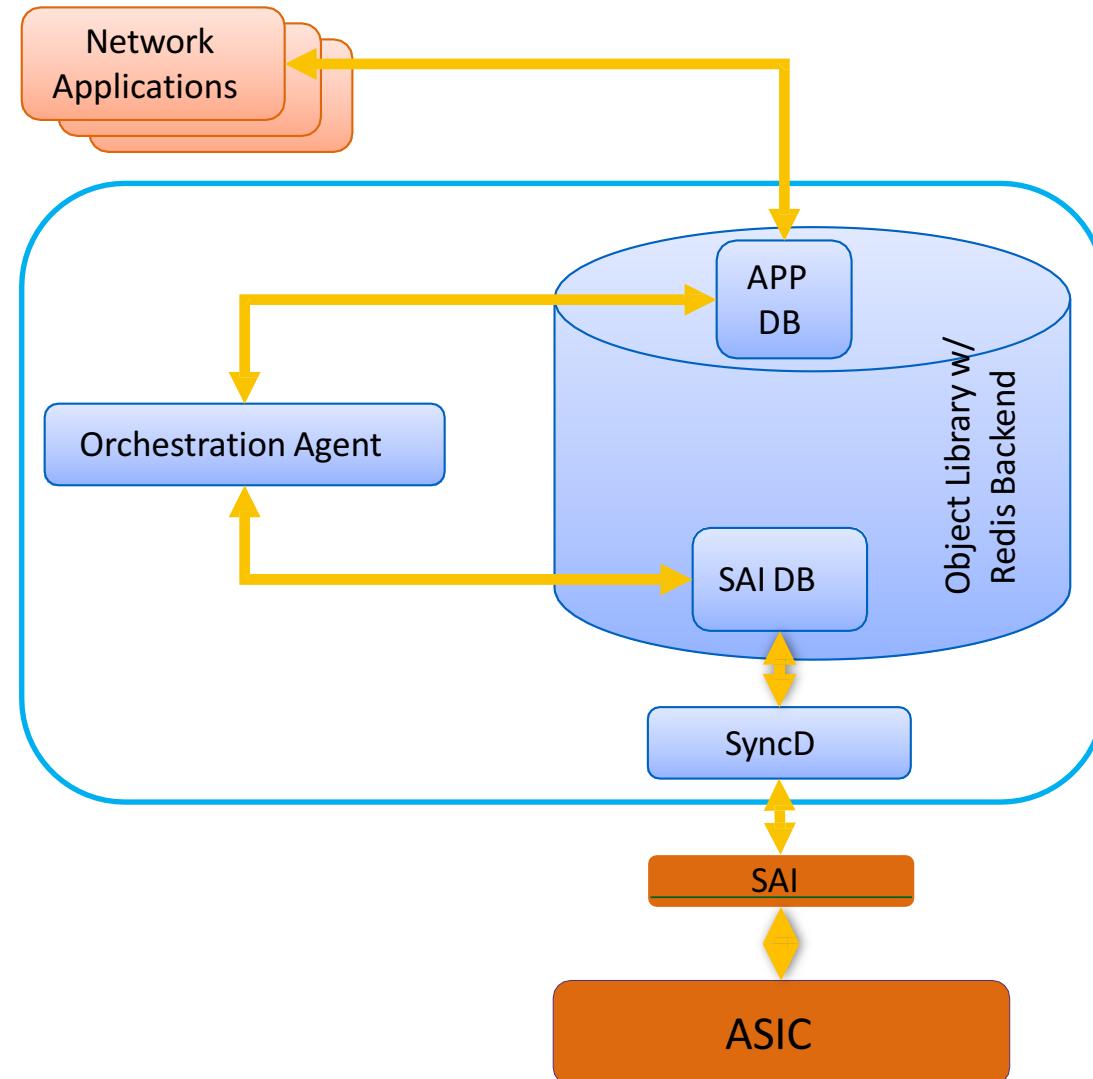


Vendor-less Network OS



SONiC Architecture - Switch State Service (SWSS)

- SAI DB: persist SAI objects
- App DB: persistApp objects
- DBbackend: redis with object library
- SyncD: sync SAI objects between software and hardware
- Orchestration Agent: translation between apps and SAI objects, resolution of dependency and conflict
- Key Goal:
 - Evolve components independently
 - Enable easy integration of containers
 - Enable large scale debugging



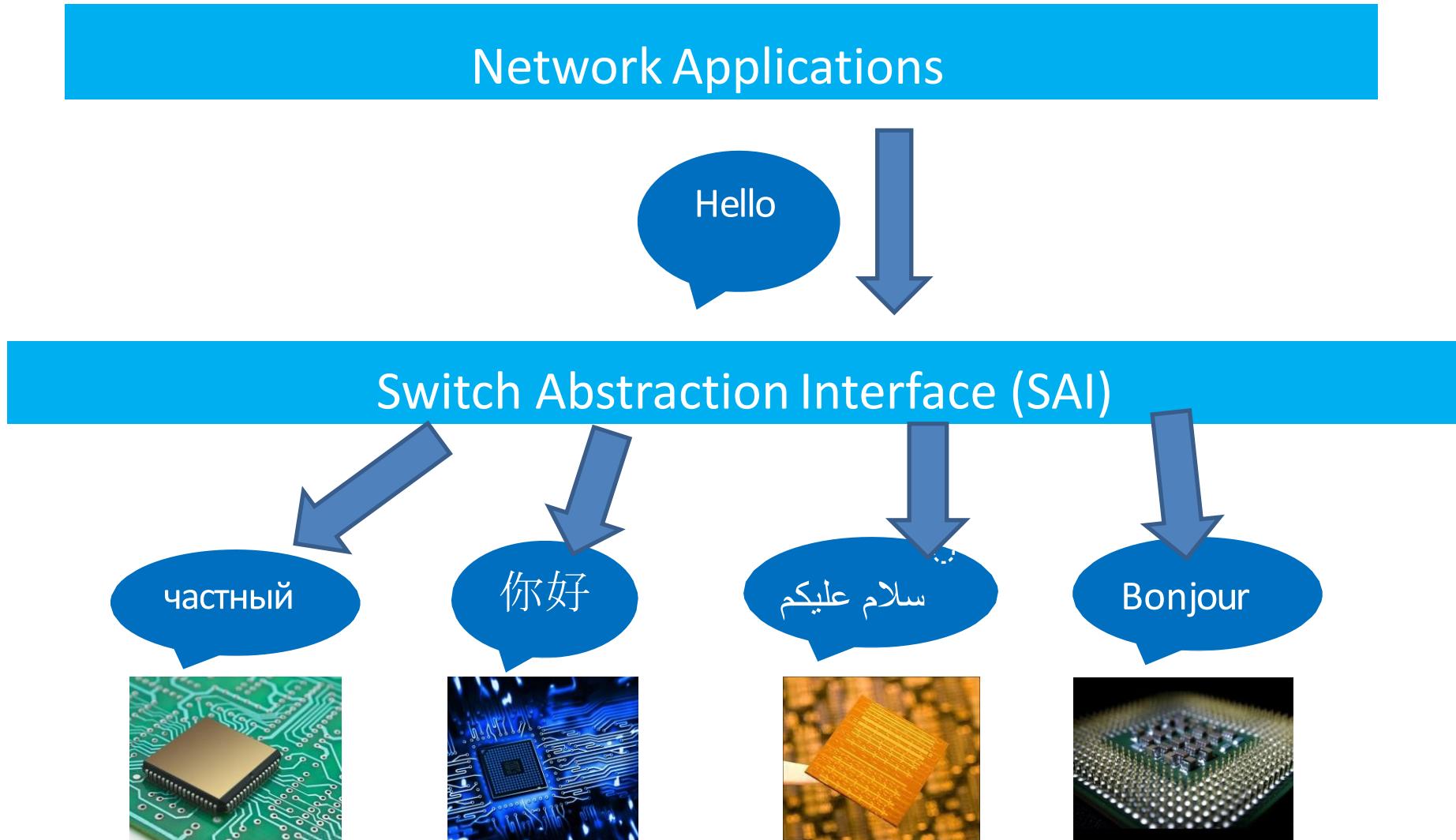
SoNIC Features Status

SONiC.201705	5/15/2017	0.9.4	BGP
			ECMP
			LAG
			LLDP
			QoS - ECN
			QoS - RDMA
			Priority Flow Control
			WRED
			COS
			SNMP
			Syslog
			Sysdump
			NTP
			COPP
			DHCP Relay Agent
			SONiC to SONiC upgrade
			Multiple Images support

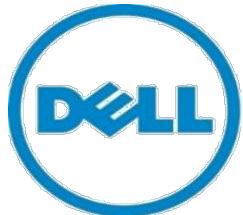
SONiC.201709	9/15/2017	0.9.4	VLAN
			ACL permit/deny
			IPv6
			Tunnel Decap
			Mirroring
			Post Speed Setting
			BGP Graceful restart helper
			BGP MP
SONiC.201712	12/15/2017	1.0	Fast Reload
			PFC WD
			SONiC Support SAI 1.0
			TACACS+
			MAC Aging
			LACP Fallback
			MTU Setting
			Vlan Trunk
			IPv6 ACL
			BGP/Neighbor-down fib-accelerate
			Port breakout
			Dynamic ACL Upgrade
			SWSS Unit Test Framework (best effort)
			ConfigDB Framework

Switch Abstraction Interface (SAI)

Switch Abstraction Interface (SAI)



CNOS FlexSwitch Metaswitch OS10 OPX SONiC



Tofino, P4



Goldengate



Trident, Tomahawk



Prestera



XPliant



Taurus

Spectrum



77
members

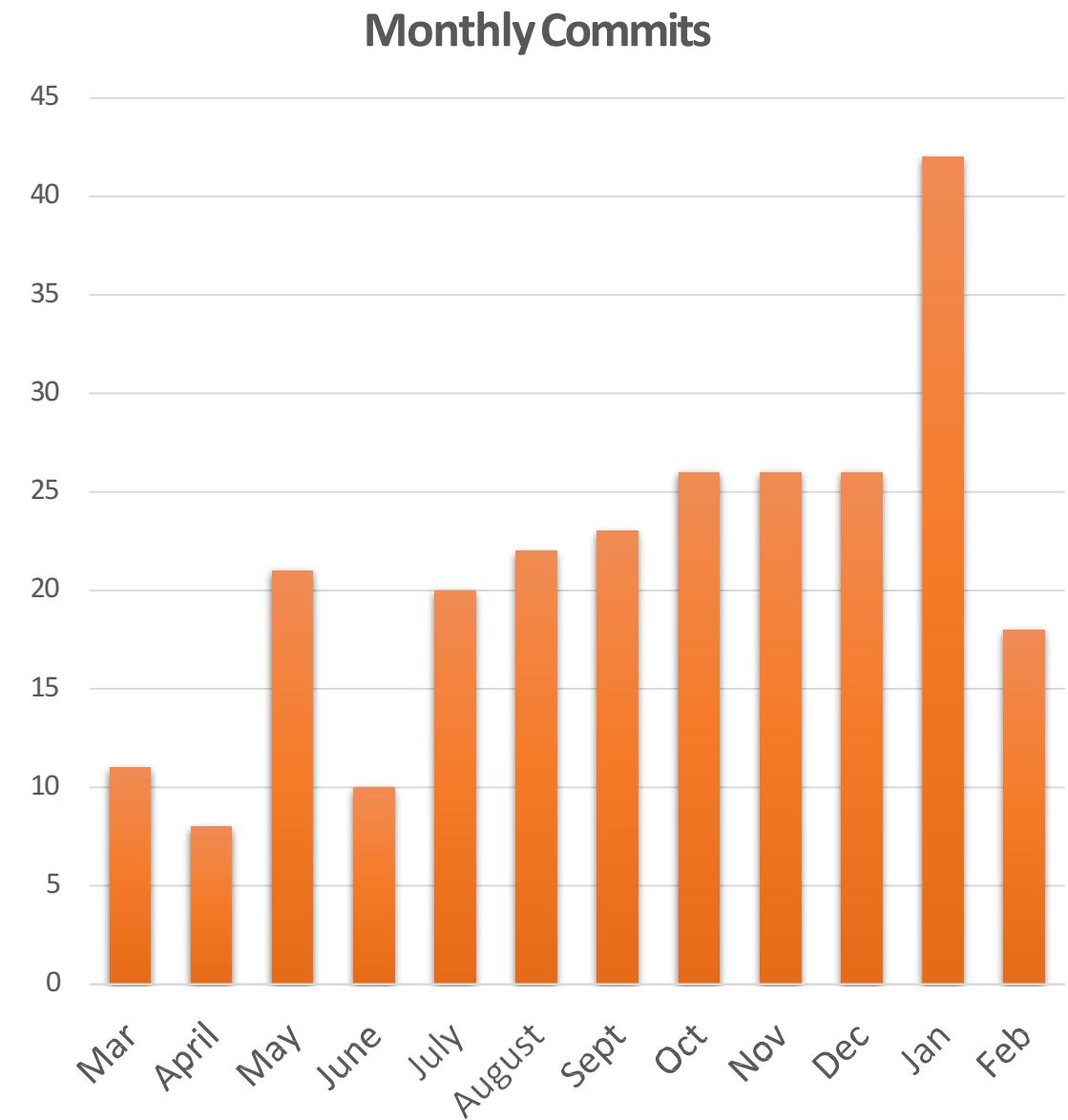
48
Contributors

472
Commits

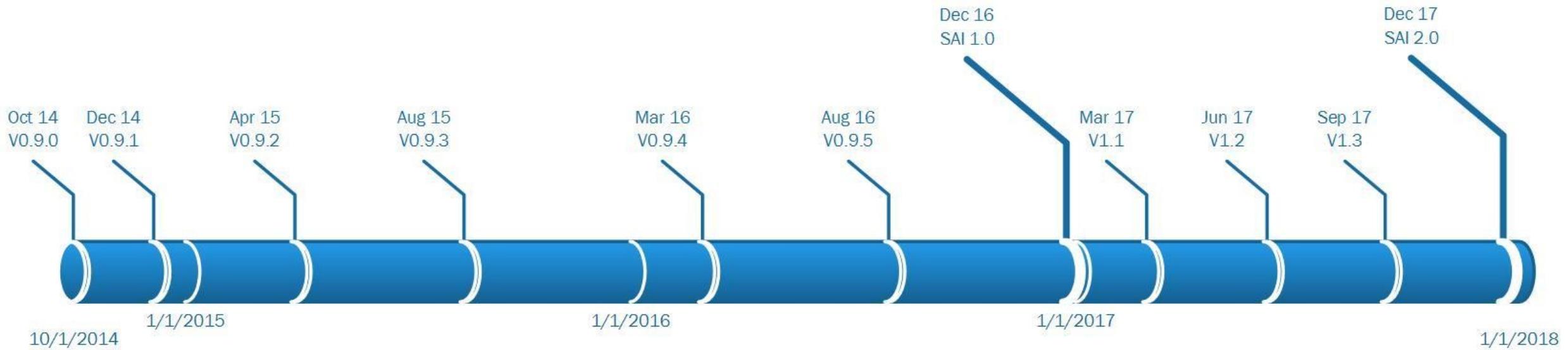
>60
meetings
2016

6
Releases

37
Proposals



SAI Releases



The Roadmap of SAI in 2017

Monitoring

TAM [Broadcom]

Microburst
[Marvell]

Critical Resource
Monitoring [MSFT]

INT [Barefoot]

Protocol Support

MPLS [Mellanox]

802.1BR [Dell]

Segment Routing
[Cavium]

Open flow Extension
[Cavium]

Reliability/QoS

L3 FastReroute
[Metaswitch]

BFD [Dell]

ECN [Dell]

Infrastructure

SAI P4 Model
[Mellanox]

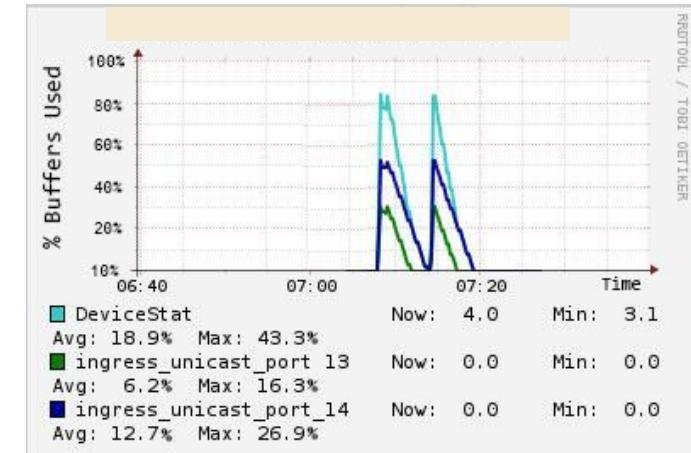
Multi-NPU [Dell]

Capability Query
[MSFT]

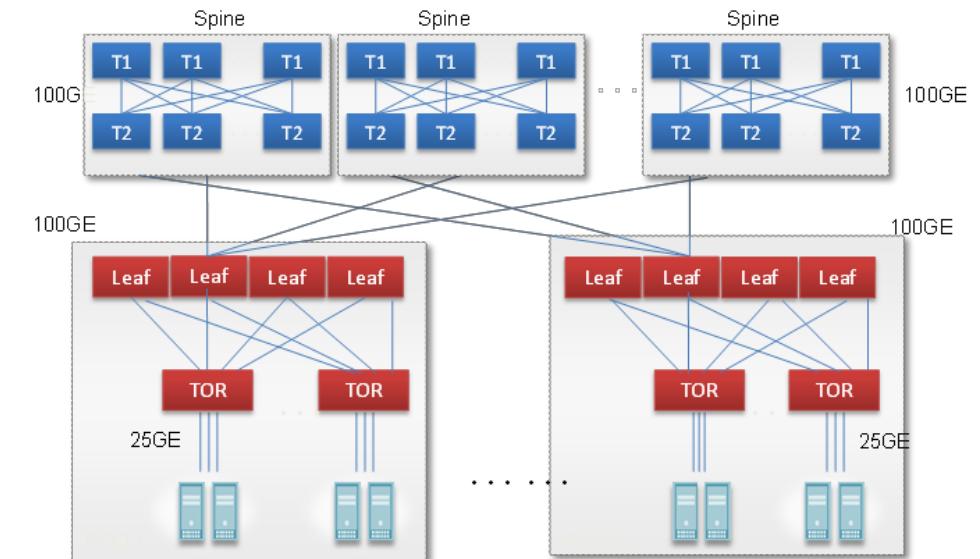
SAI Ext API [Dell]

Scalable Monitoring of Data Center Networks

“How to monitor buffer occupancies in a large scale data center networks in a scalable way?”



Use case : Typically internet traffic flows from Spine to leaf and then to host. When multiple streams destined to servers connected through the same leaf/spine switch, they could create a congestion scenario.



BROADCOM

Proposal Details

- TAM is an API for monitoring and controlling buffer occupancies.
- TAM facilitates real-time microburst detection through threshold alerts
- TAM enables tracker objects to track multiple statistics
- TAM supports multiple snapshot objects for simultaneous capturing of different sets of statistics
- TAM uses transporter objects for delivering snapshots at a desired location
- TAM can be easily customized for underlying hardware

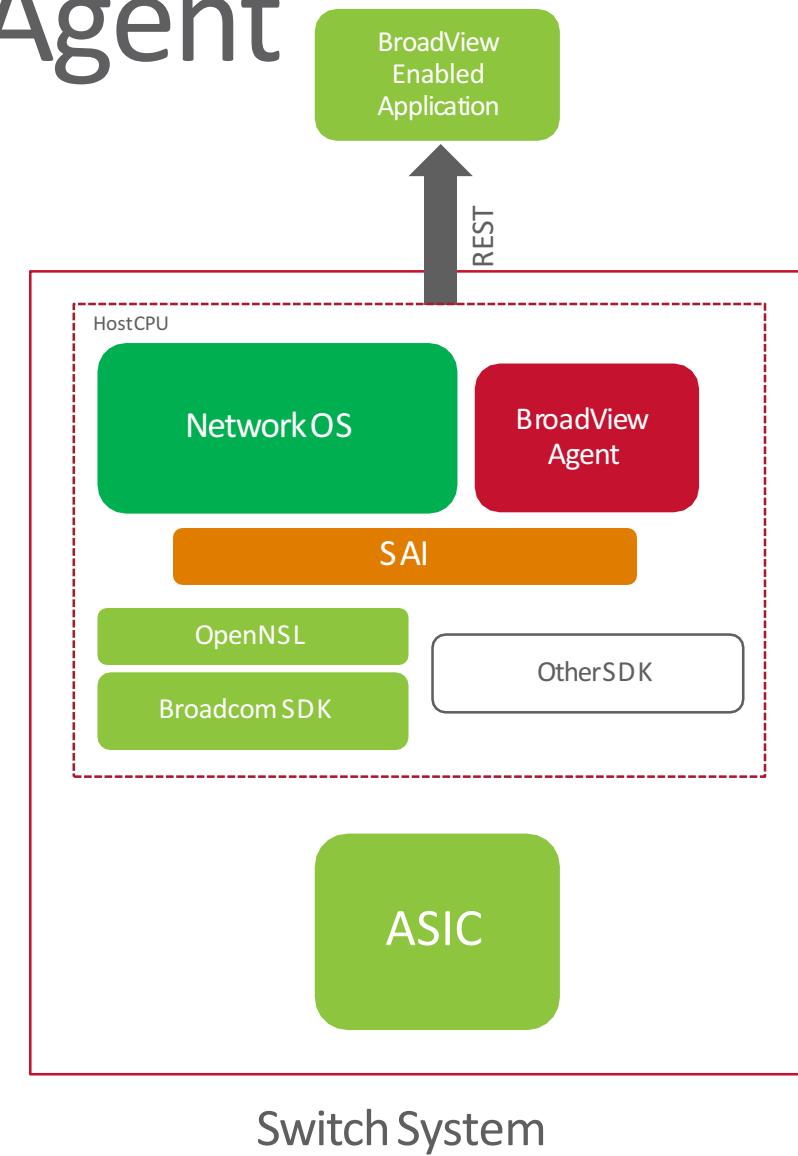
Broadview Instrumentation Agent

Platform agnostic agent for advanced analytics

Light weight with high scalability

Working in progress to integrate into SONiC

Pre-integrated with Open Ecosystem projects





BROADCOM APIs

OpenFlow DataPath Abstraction (OF-DPA) &

Open Network Switch Library (OpenNSL)

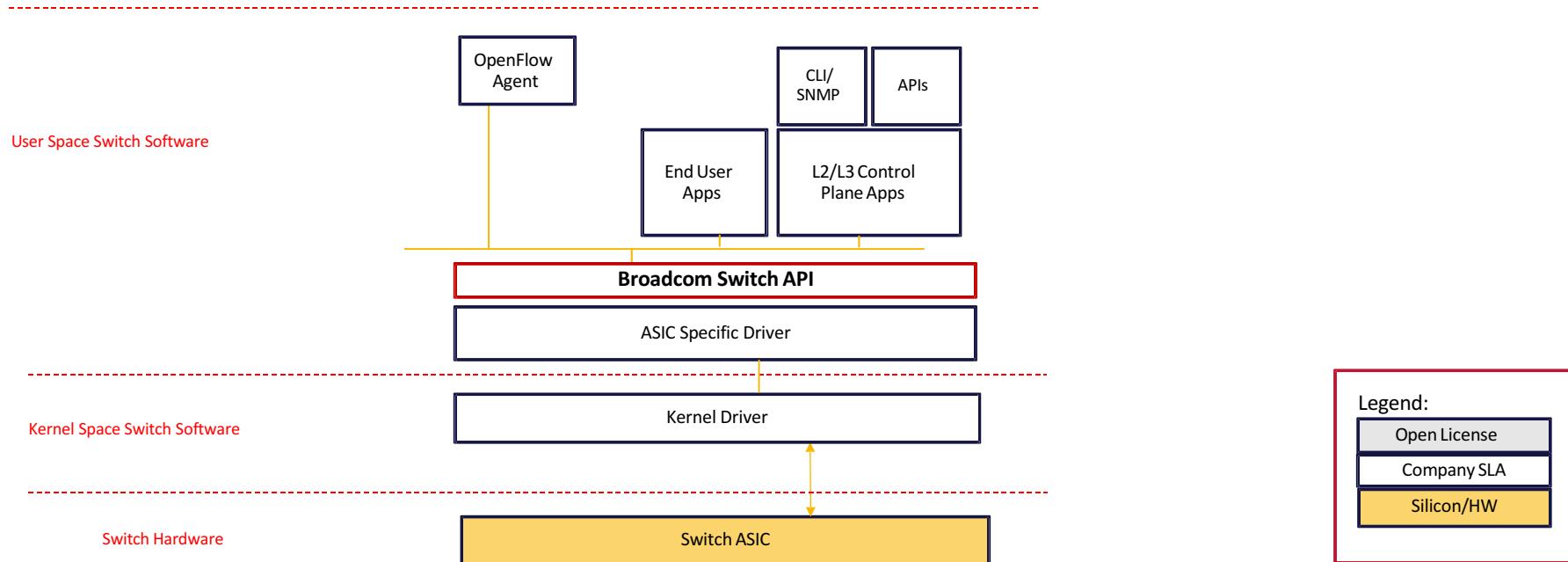
ONL/OFDPA Use Cases

User survey and feedback driving requirements

- Enable open source innovation on switch ASICs
 - Through OCP, GitHub
- Open Switch API must not break compatibility
 - Broadcom switch SDK and API used widely
 - OEM, OSV/ISV, Operator network operating systems
- Enable operator innovation on OEM/ODM switches
 - Access to open switch APIs for operator-developed apps
 - Operates side-by-side with OEM, OSV/ISV control plane

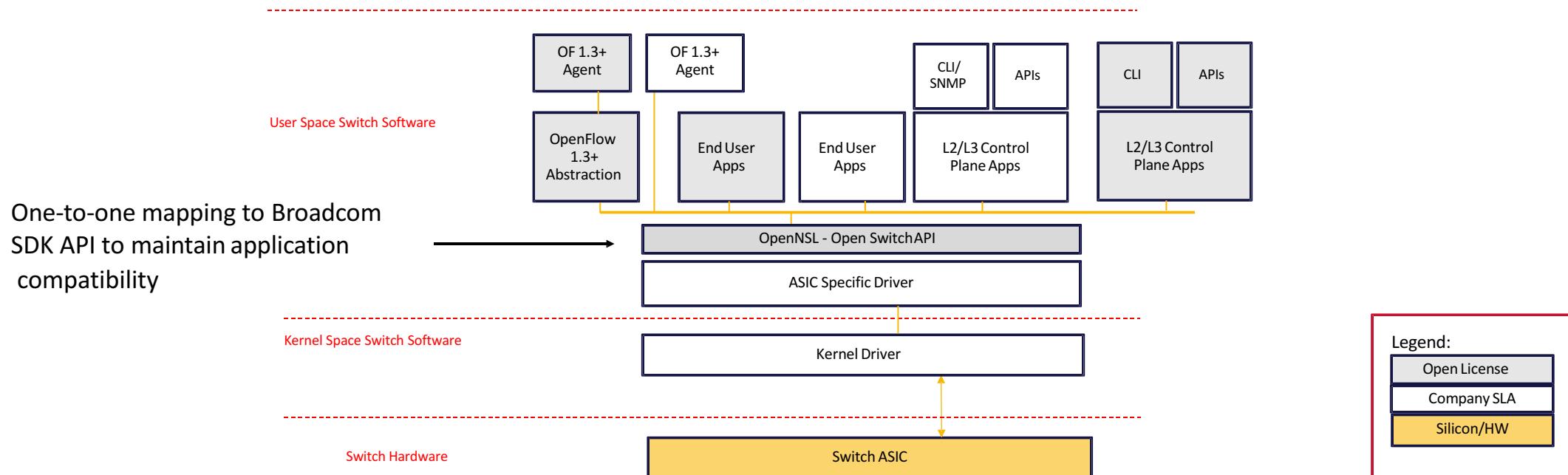
The Broadcom Switch API

Many OEM, OSV & End User Apps on Switch API Apps
above Switch API are closed source



The OpenSL API

Enable Open Source Apps on Switch API Co-exist with Proprietary Apps



Sample 1-1 Mapping

Broadcom SDK APIs

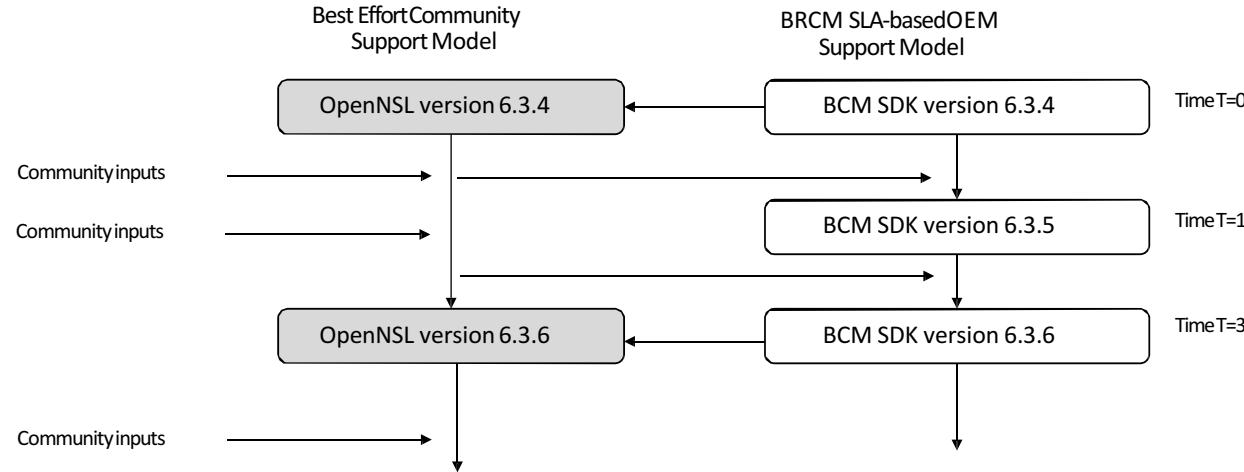
```
int bcm_rx_start(...)  
    Initialize the Rx subsystem (start a Rx thread)  
int bcm_rx_register(...)  
    Register an upper Rx handler  
int bcm_tx_init(...)  
    Initialize the Tx (start a Tx thread)  
int bcm_tx(...)  
    Send a packet  
int bcm_linkscan_register(...)  
    Register an upper linkscan handler  
...
```



OpenSL APIs

```
int opennsl_rx_start(...)  
    Initialize the Rx subsystem (start a Rx thread)  
int opennsl_rx_register(...)  
    Register an upper Rx handler  
int opennsl_tx_init(...)  
    Initialize the Tx (start a Tx thread)  
int opennsl_tx(...)  
    Send a packet  
int opennsl_linkscan_register(...)  
    Register an upper linkscan handler  
...
```

Keeping Versions in Sync



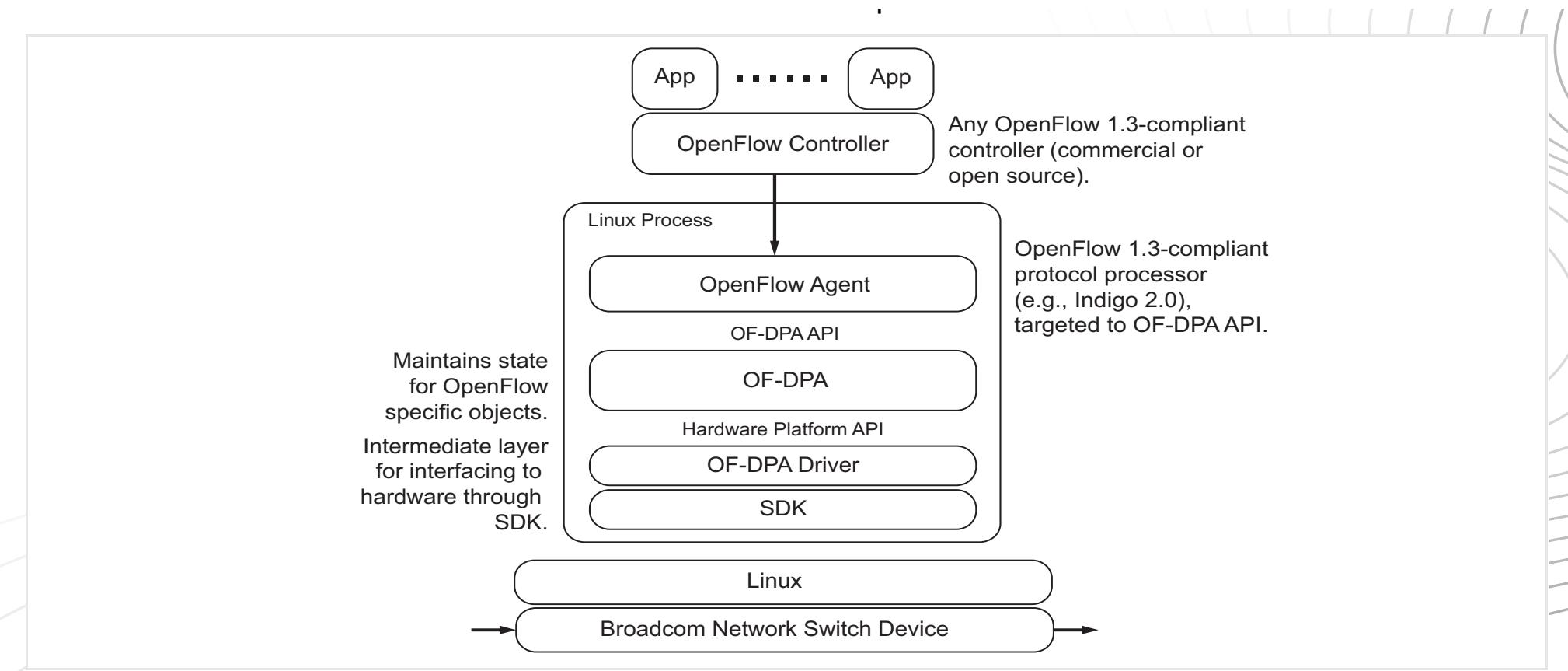
OpenNL releases are tied to Broadcom SDK releases.

Same versioning nomenclature to be used to avoid confusion

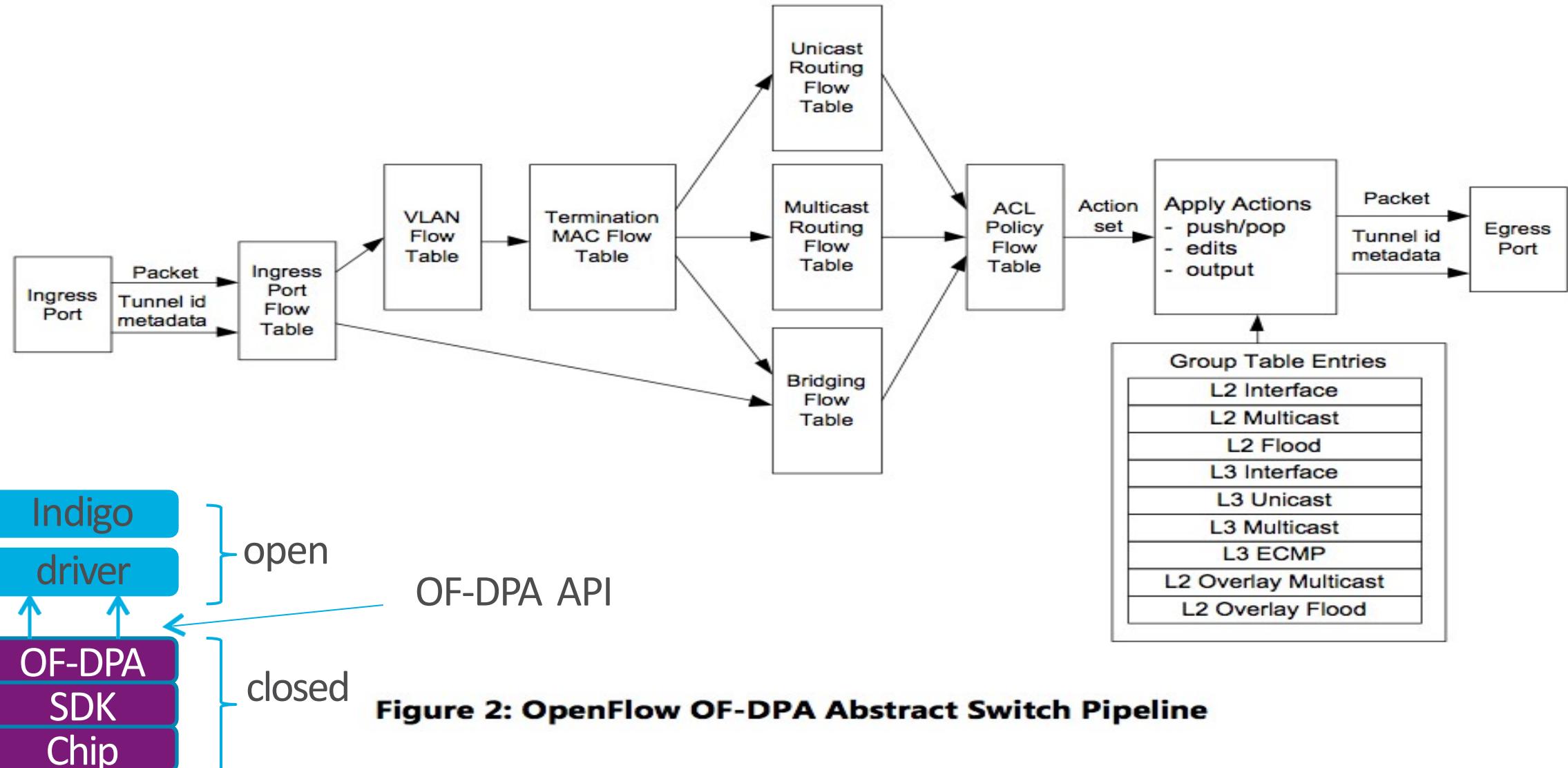
Not all SDK releases might have a corresponding OpenNL release.

OFDPA Component Layers

- It is a software component that provides a hardware adaption layer between an OpenFlow agent and Broadcom network switch devices

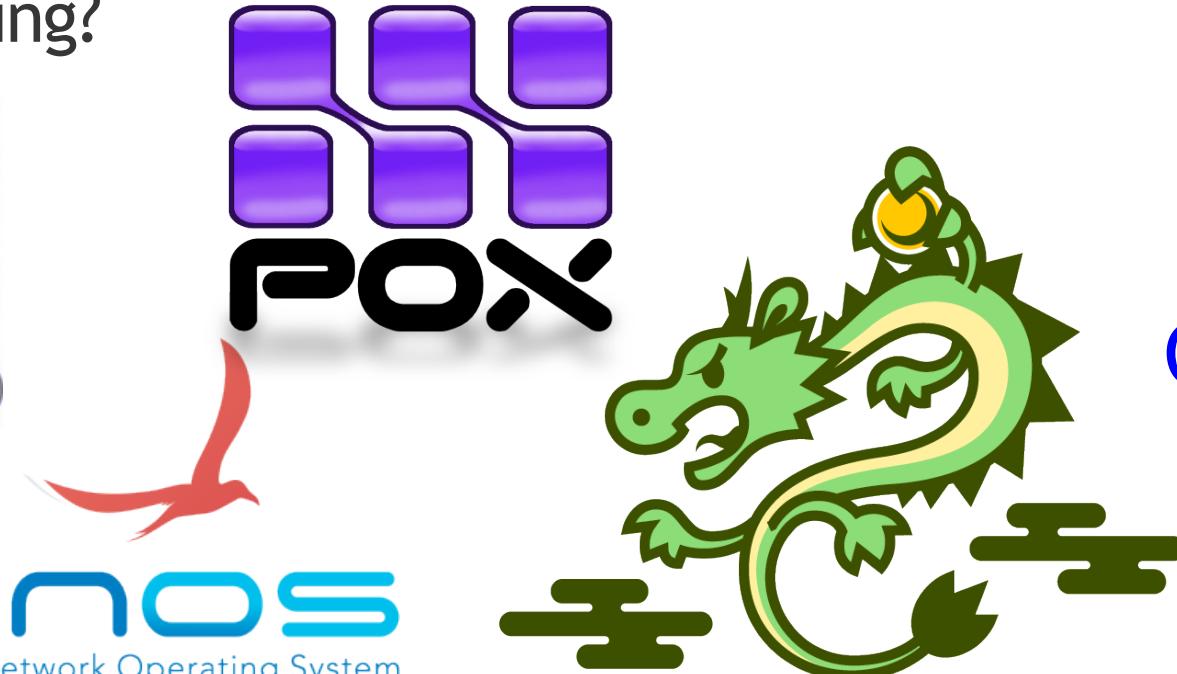


OpenFlow Datapath Abstraction:OF-DPA

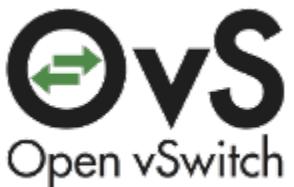


Lots of open source controllers

- But what are they controlling?



Open
Source
SDN
Controllers



SDN OSS Software Switches



SDN OSS
Hardware
Switches

DISCUSSION: SAI VS. OPENNSL VS. OFDPA

- All of them are evolving at the same pace
- Closed vs. Open Source
- SAI gets you chip vendor independence (in theory)
 - Broadcom, Mellanox, Cadvium, Barefoot (via software)
- OpenNSL is closest to existing BCM APIs
- OF-DPA exposes other functionality like OAM and MPLS
- There is a move to merge these APIs
 - e.g., SAI implemented on OpenNSL, OF-DPA implemented on OpenNSL

References

1. <https://azure.microsoft.com/en-us/blog/sonic-the-networking-switch-software-that-powers-the-microsoft-global-cloud/>
2. <https://opennetlinux.org/Open%20Network%20Linux%20-%20A%20Programmers%20View.pdf>
3. <https://github.com/opencomputeproject/SAI/tree/master/doc>
4. <http://files.opencompute.org/oc/public.php?service=files&t=ba2fbceb1b920ce7d87aa68d7248e5f&download&path=/broadcom-openssl.pdf>
5. <https://docs.broadcom.com/docs-and-downloads/collateral/pb/OF-DPA-PB100-R.pdf>
6. <https://sites.google.com/view/iu-whitebox-project/hands-on/sonic>
7. <http://onosproject.org/wp-content/uploads/2016/07/fabric-press-trellis-guru.pptx>