Stratum: An Overview

Brian O'Connor
brian@opennetworking.org
Open Networking Foundation (ONF)

December 3, 2018

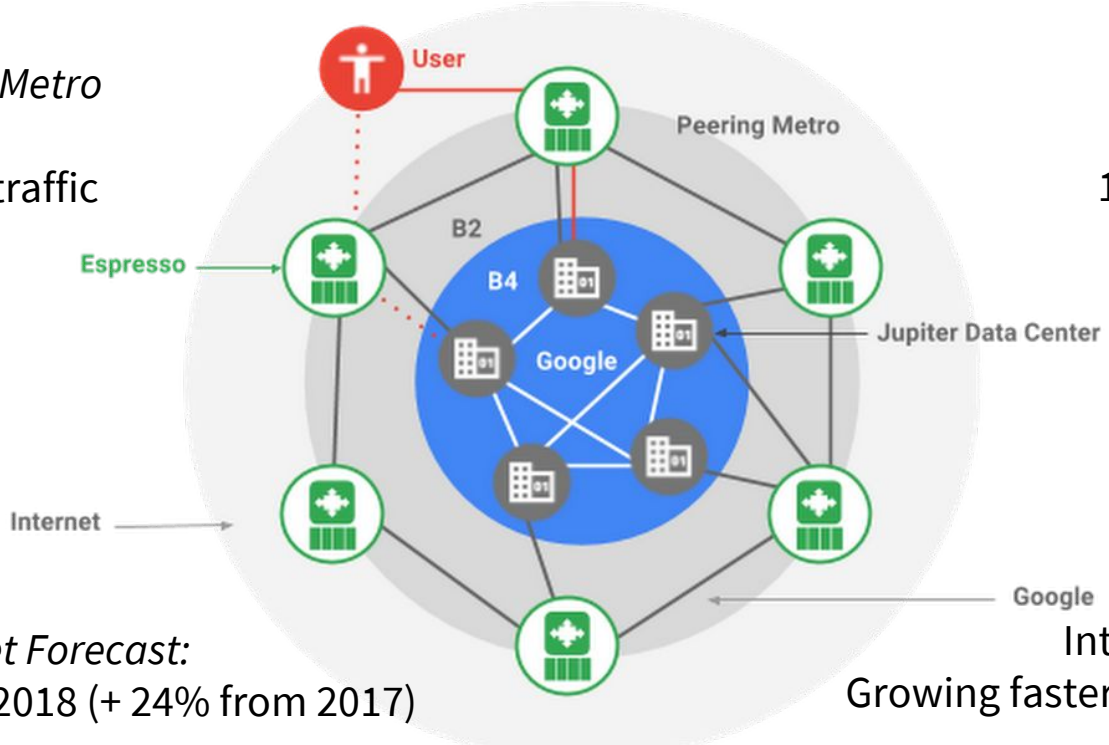
ONF's History

The ONF has a lot of experience building SDN and NFV solutions

Google's History

Google runs SDN networks at scale

Espresso
SDN Peering Edge / Metro
70 metro sites
25% of all Internet traffic



Jupiter
SDN Data Center
1.3 Pbps
100,000+ servers/site

Cisco Global Internet Forecast:
~150 EB/month in 2018 (+ 24% from 2017)

B4
SDN WAN
Inter-datacenter traffic
Growing faster than Internet traffic

https://www.blog.google/topics/google-cloud/making-google-cloud-faster-more-available-and-cost-effective-extending-sdn-public-internet-espresso/
https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf

SDN Provides Many Benefits

- **Fine-grained control** enables support for more complex QoS and load balancing policies
- **Control plane optimizations** difficult to achieve using traditional networking
- **Enhanced network visibility** for troubleshooting, monitoring, and auditing
- New features can be added **by operators** at **software time scales**
- … and the list goes on

So, what’s the catch?

Challenges with Existing SDN

- Programmatic Network Interfaces are Inconsistent and Incomplete
  - OpenFlow provided no data plane pipeline specification; every vendor's pipeline is different
  - Every vendor provides their own proprietary models for configuration or management
  - Differences in protocol implementations require custom handling in the control plane
- Control planes are written and tested against specific hardware
  - Some control planes have worked around this by building their own abstractions to handle these differences, but new abstractions are either **least common denominator** (e.g. SAI) or **underspecified** (e.g. FlowObjectives)
  - Other control planes have exploited specific APIs are essentially **locked in to specific vendors**, which slows innovation cycles

Aside: Future of OpenFlow

- OpenFlow 1.x is not going away (yet)
  - Many vendors have built OpenFlow implementations for their hardware and software switches
  - Plan to continue to support OpenFlow in ONOS
  - Core abstractions work for both OpenFlow and P4 Runtime
- It will take some time before Stratum is generally available
  - *Hopefully, there will be hardware and software support by 2019*
- This work builds on and improves OpenFlow
  - Provides similar interaction model which should make transitioning to Stratum-based solutions easy (ONOS is a proof-point for this)

Challenge: Programmable Devices

- Programmable forwarding chips are here
  - Control protocols are mostly fixed function
  - Extensibility is difficult or takes too long
  - Even “fixed-function” forwarding chips have some degree of programmability and may add new features
- Need to define mutable contract between the hardware vendor and network operator
  - Different operators can have different contracts
  - Operators need to work with vendors to determine limitations
  - Contract must be designed for programmability; APIs must be contract-independent
  - May be useful to have community consensus on reference contracts
- Operators can dynamically define pipeline and dataplane features to enable new use cases and device behaviors/roles in the network
  - Experimentation should require minimal effort and rollout should be seamless

Challenge: Bringing SDN to production at scale

- Operators have hit limitations with existing protocols when introducing them into their networks at scale
  - e.g. OpenFlow, NETCONF
  - Interfaces and services must evolve to meet operational needs
- SDN at scale brings new requirements
  - Frequent updates to all layers of the stack (software AND state)
  - Monitoring, visibility and automation for rollout and operations
  - Design time validation and testing to minimize risk to production

Challenge: Handling Migration

- Widespread, greenfield adoption of new technology is not feasible
  - Ops teams need to build experience and confidence
  - Business teams won't allow infrastructure forklift from capex perspective
- Types of Migration
  - Fixed-function to programmable switching chips
  - Traditional networking to SDN
  - Introduction of new vendor equipment
- Need interfaces, models and operations to be consistent to allow incremental adoption along these axes
  - Ideally, services and hardware can be introduced or migrated independently

Wanted

- New control interface with:
  - Abstraction for different types of switching chips
  - Well defined interfaces and behavior
  - Extensibility
- Common models for configuration and monitoring
- Common interfaces for operations
  - Testing, Debugging, Certificate Management, Software upgrade
- Common platform abstraction (e.g. OCP's ONLP)
- Open source switch stack

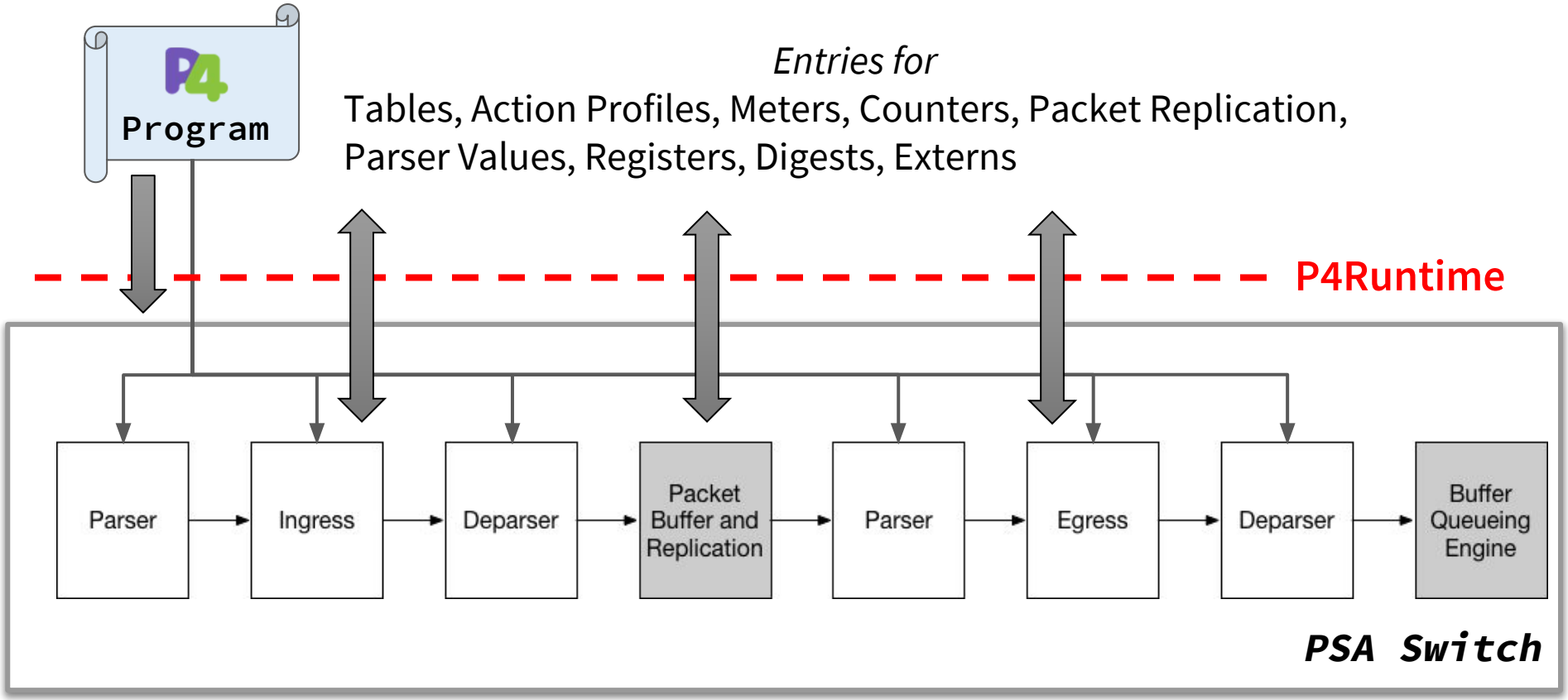
Slide adapted from Google

Requirements for SDN

- Highly scalable
- Highly automated
- High performance interfaces
- Ability to directly drive design at all levels of the software stack

Slide adapted from Google

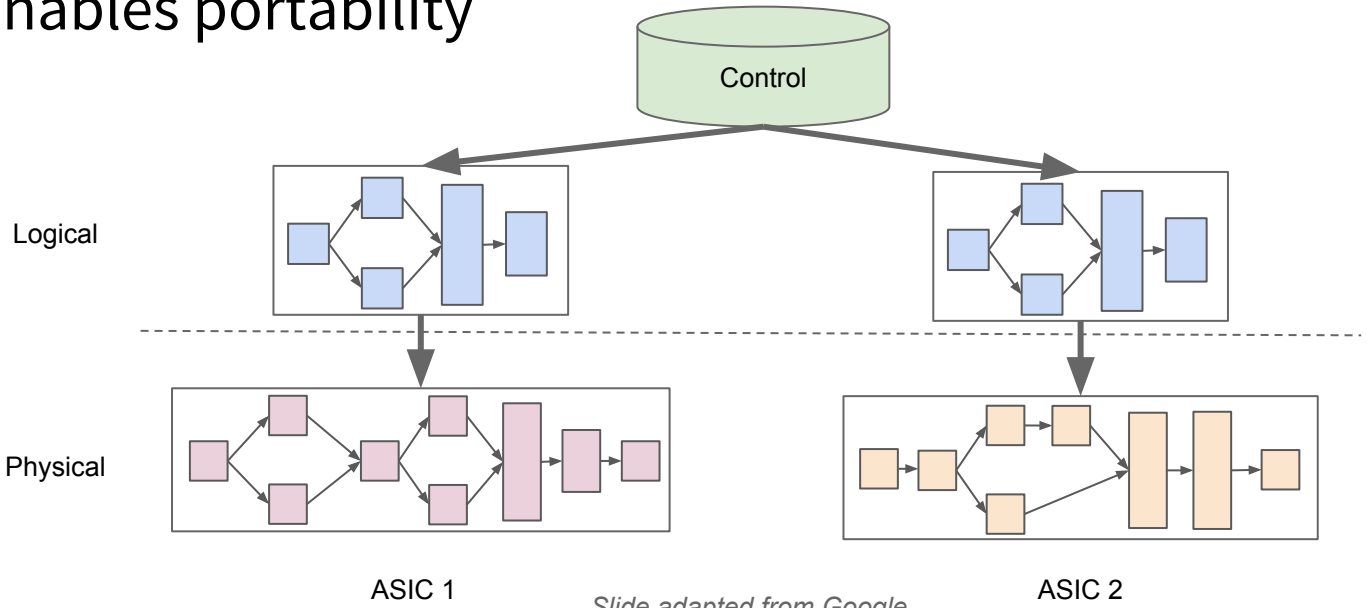
Control Interface: P4Runtime



Slide adapted from P4.org

Role of P4

- Provide clear pipeline definition using P4 tailored to role
- Useful for fixed-function/traditional ASICs as well as programmable chips
- Enables portability



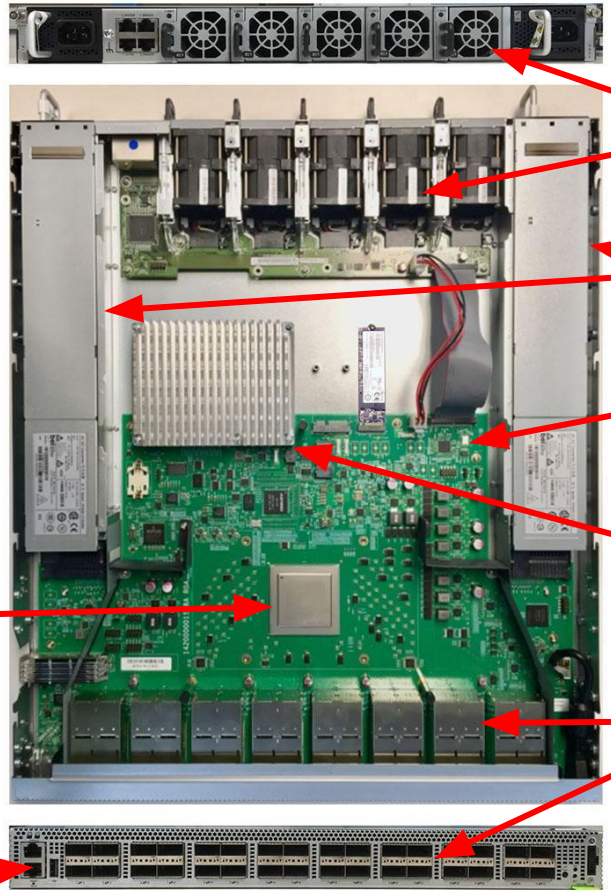
Slide adapted from Google

P4 and P4Runtime are great, but …

Still Missing:

- Configuration
- Monitoring
- Operations

OpenConfig, **gNMI**, and **gNOI** are here to help!



Fan Speed

Power supplies

Monitor Sensors
e.g. temperature

Software Deployment and Upgrade

Port State and Mapping
LED Control

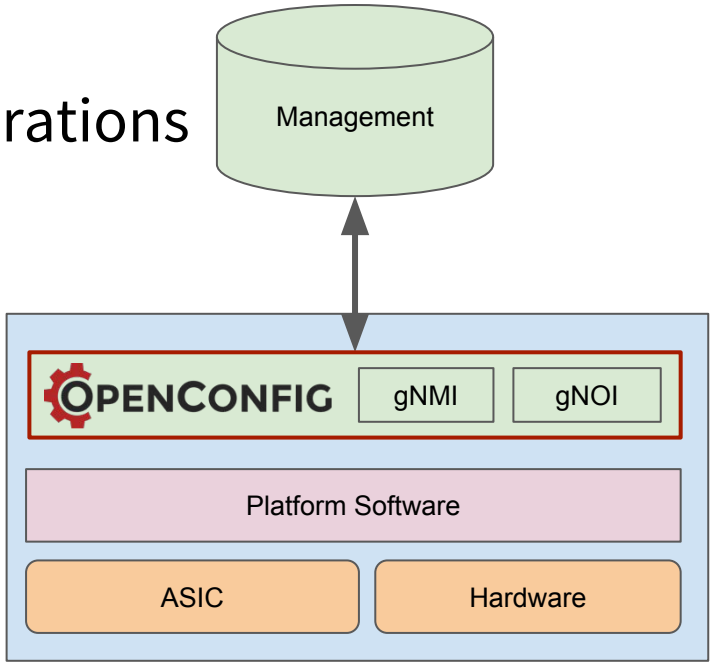
… and the list goes on.

Switch Chip Configuration
QoS Queues and Scheduling
Serialization / Deserialization
Port Channelization

Management Network

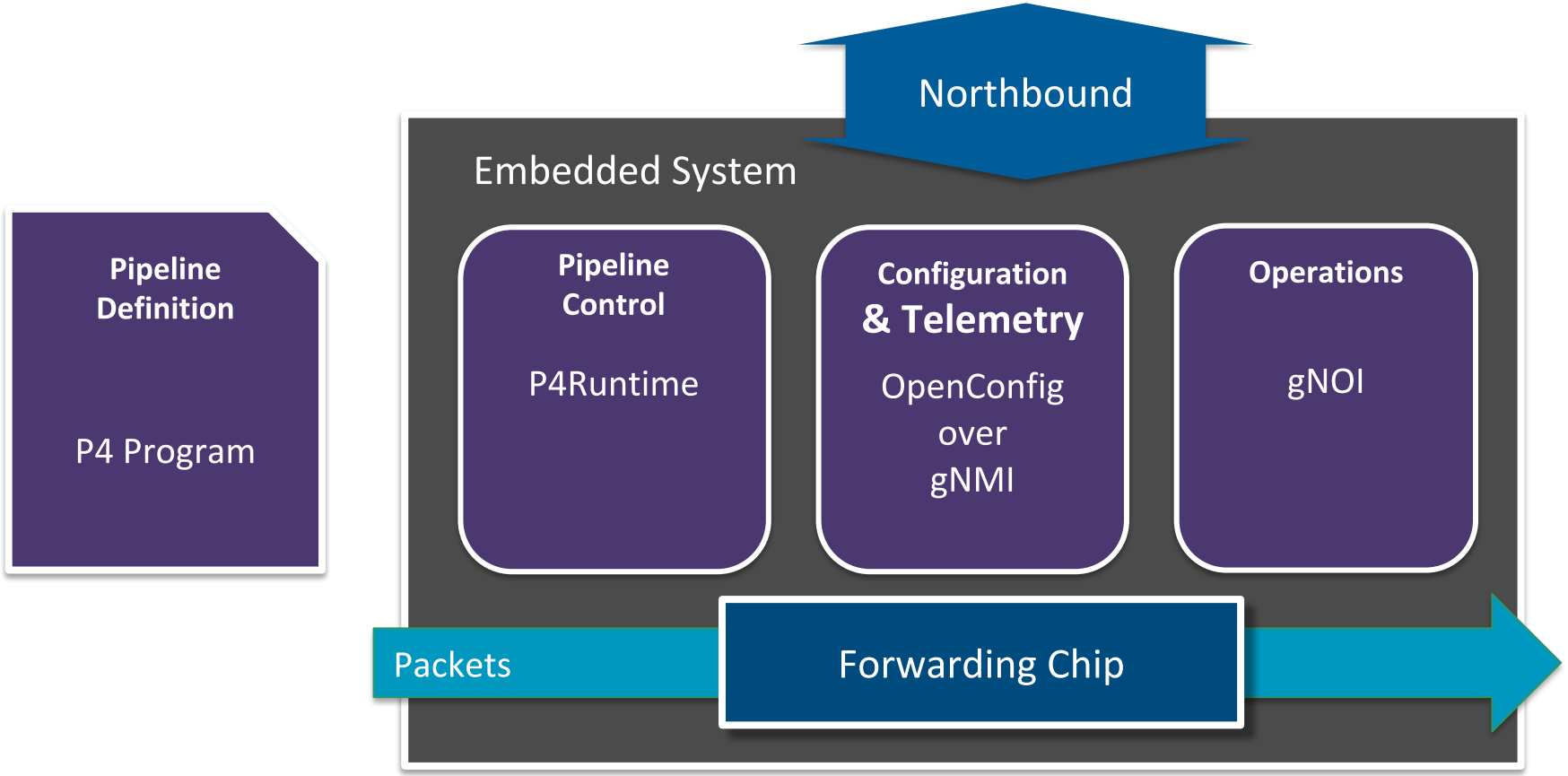
Enhanced Configuration

- Configuration and Management
- Declarative configuration
- Streaming telemetry
- Model-driven management and operations
  - gNMI - network management interface
  - gNOI - network operations interface
- Vendor-neutral data models

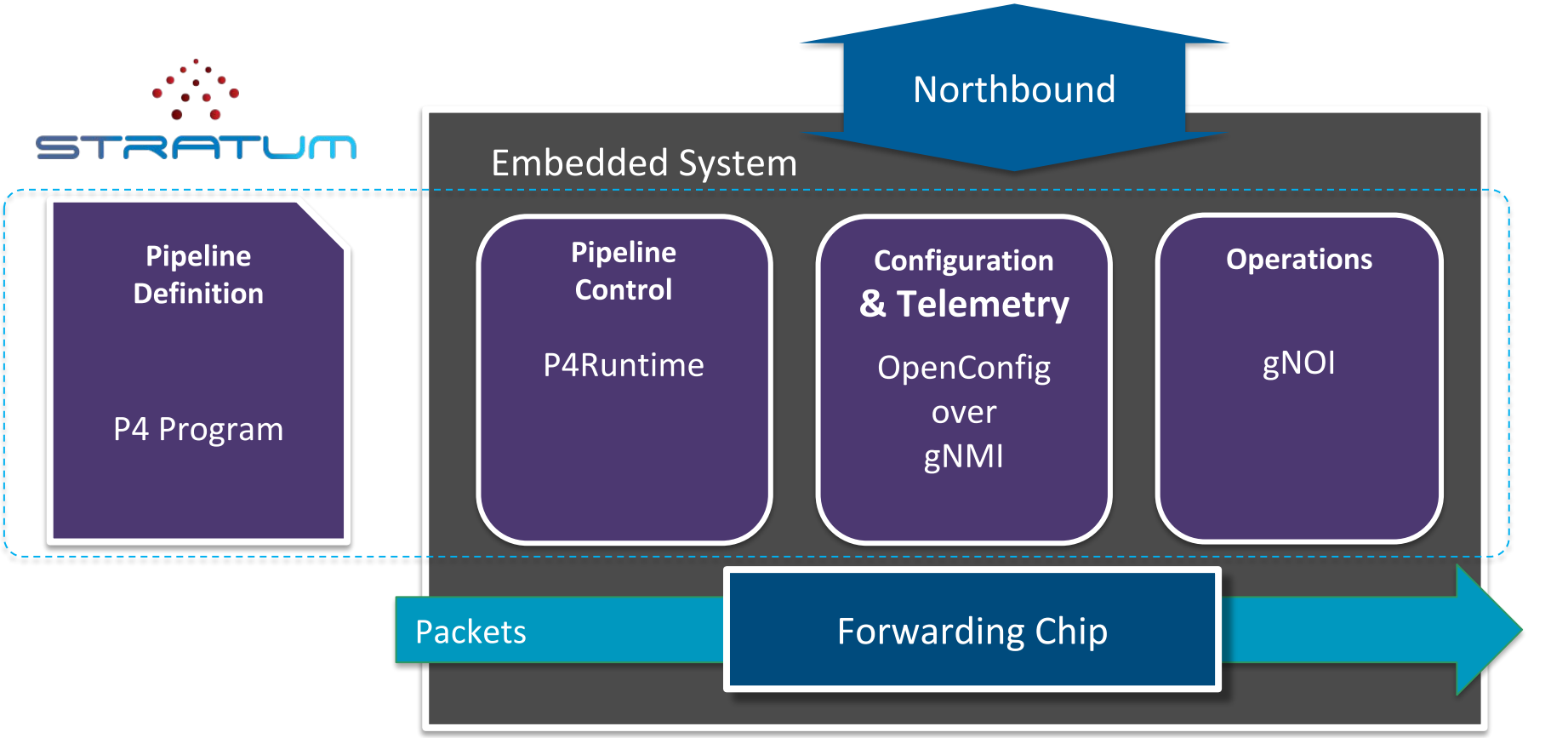


Slide adapted from Google

Next Generation SDN Interfaces



Lightweight and Production-ready Implementation

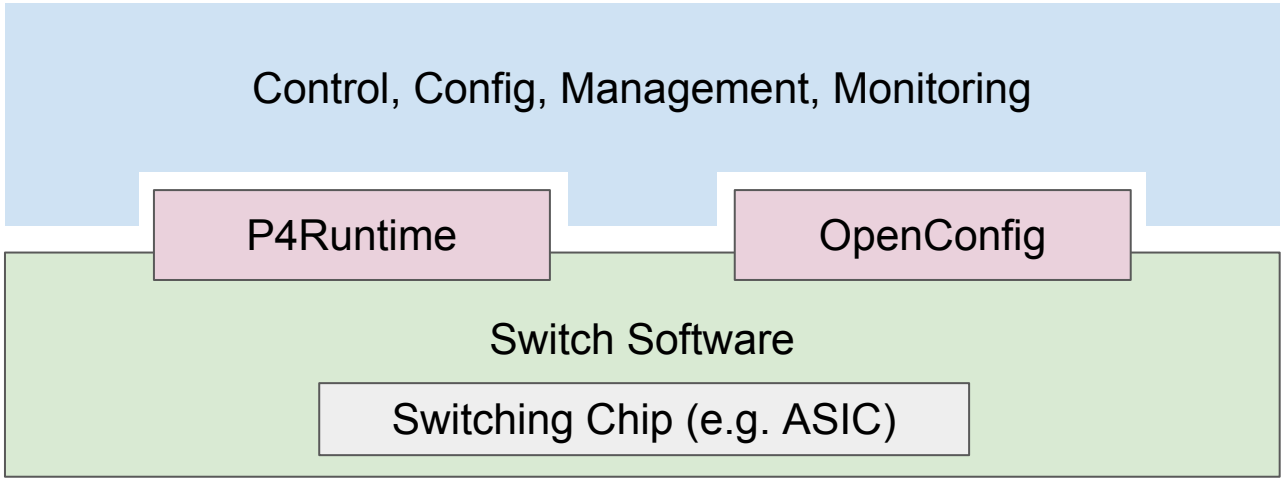


Vision

- Stratum supports multiple silicon products
- Stratum runs on many platforms
- Widely deployed in production SDN fabrics
- Enables P4Runtime and OpenConfig in the industry
- Proprietary (‘blackbox’) support desired
- Share the technology

Slide adapted from Google

Target

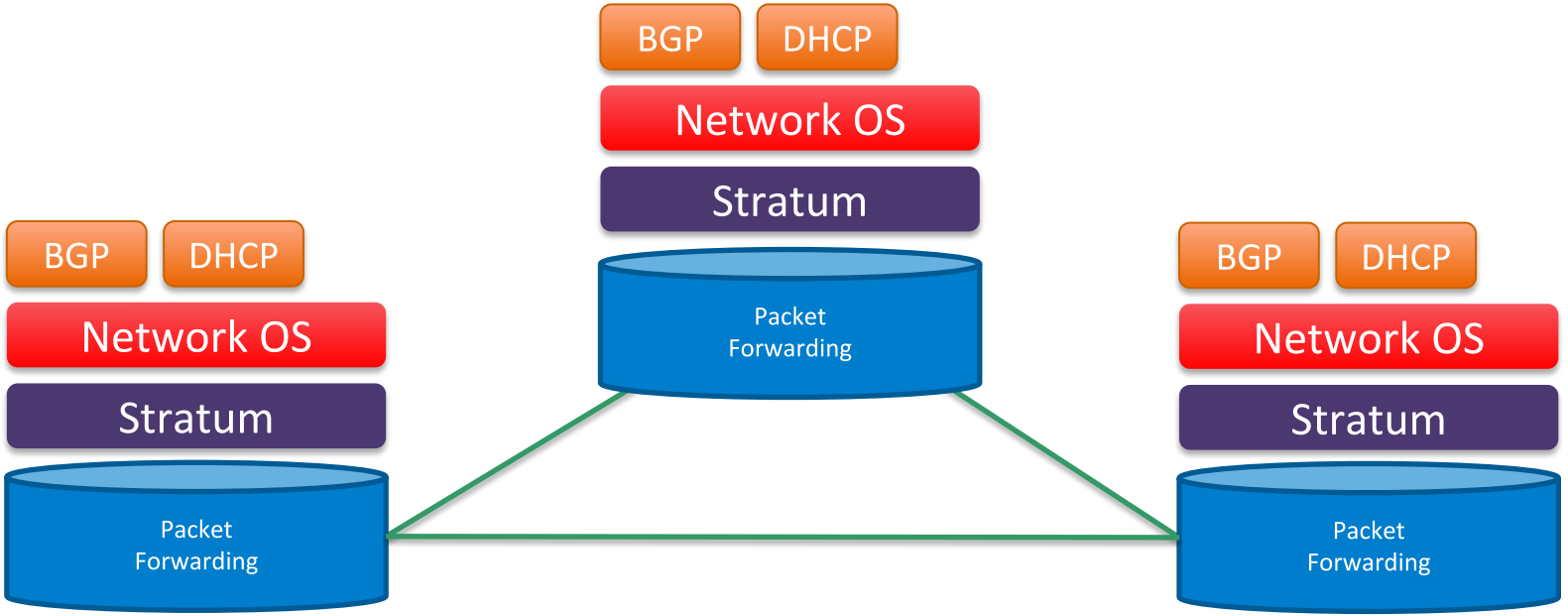


Standard, well defined interfaces allow devices to drop into existing networks (e.g. ONF's Trellis fabric or Google production SDN fabrics)

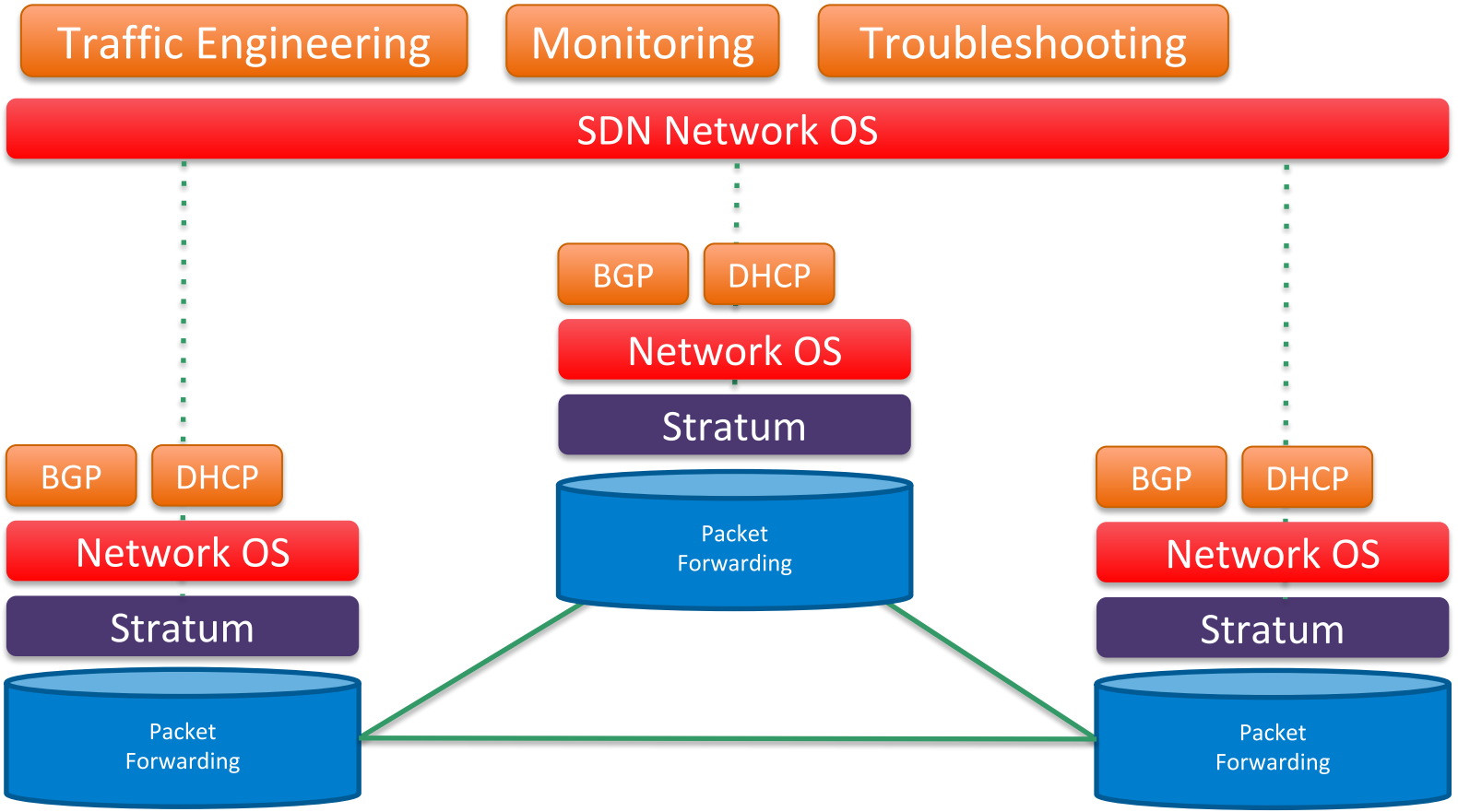
Slide adapted from Google

Stratum can support traditional Switch OSes

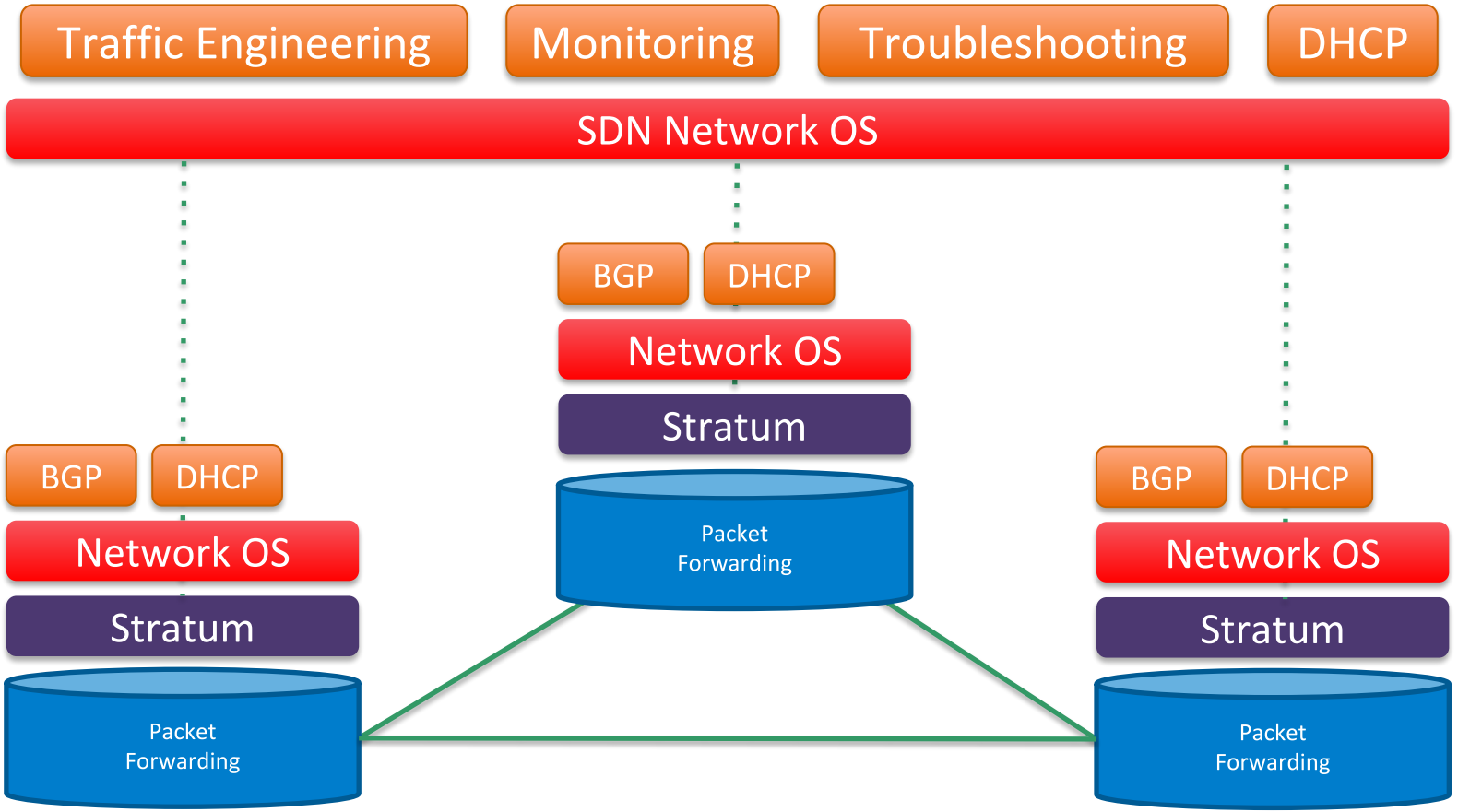
- Benefit: Clear pipeline definition and support for programmable chips
- For example, SoNIC using SAI or SAI Flex and P4Runtime/OpenConfig adapters



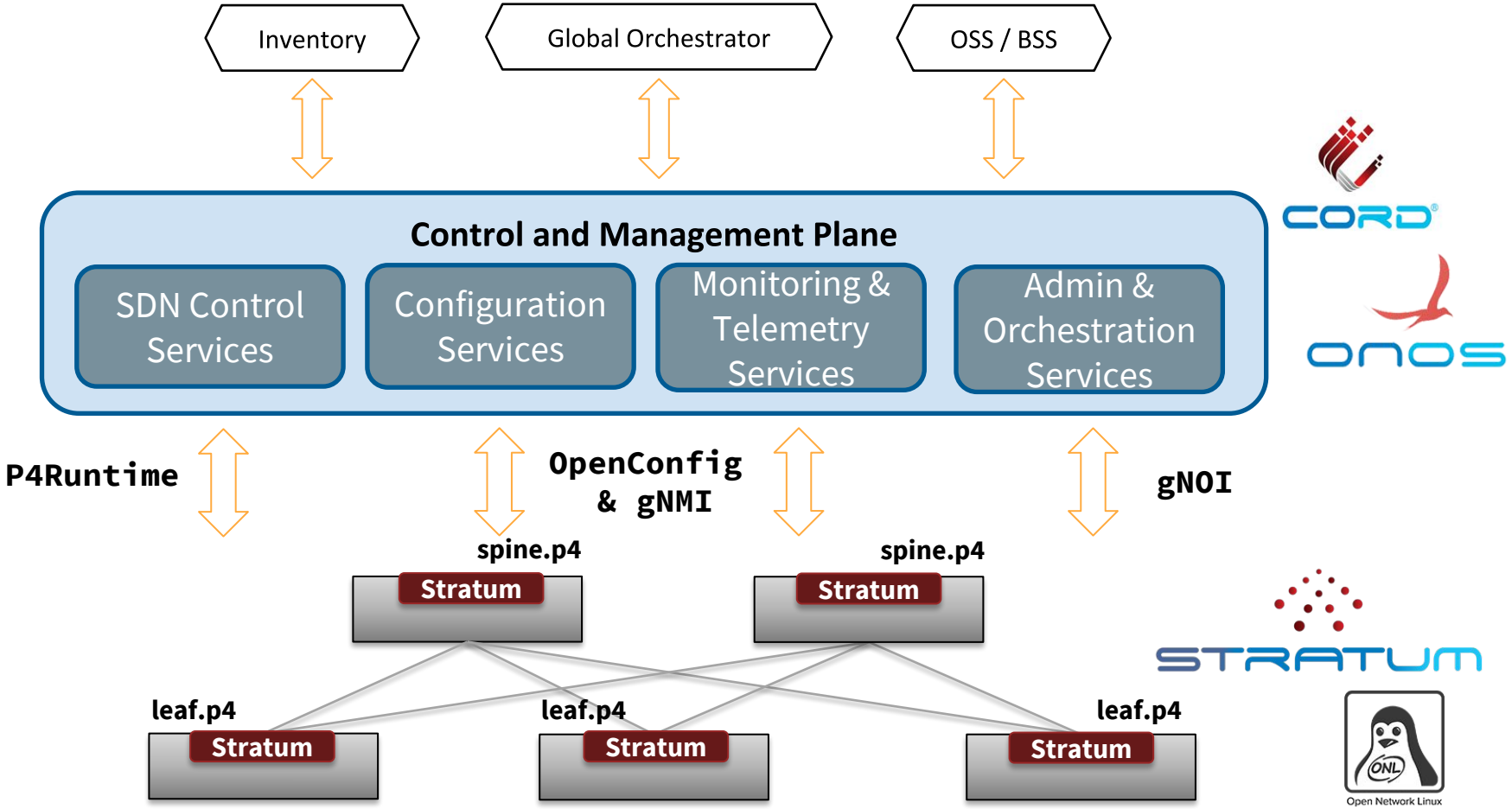
Stratum Enables Value-Add Hybrid Model



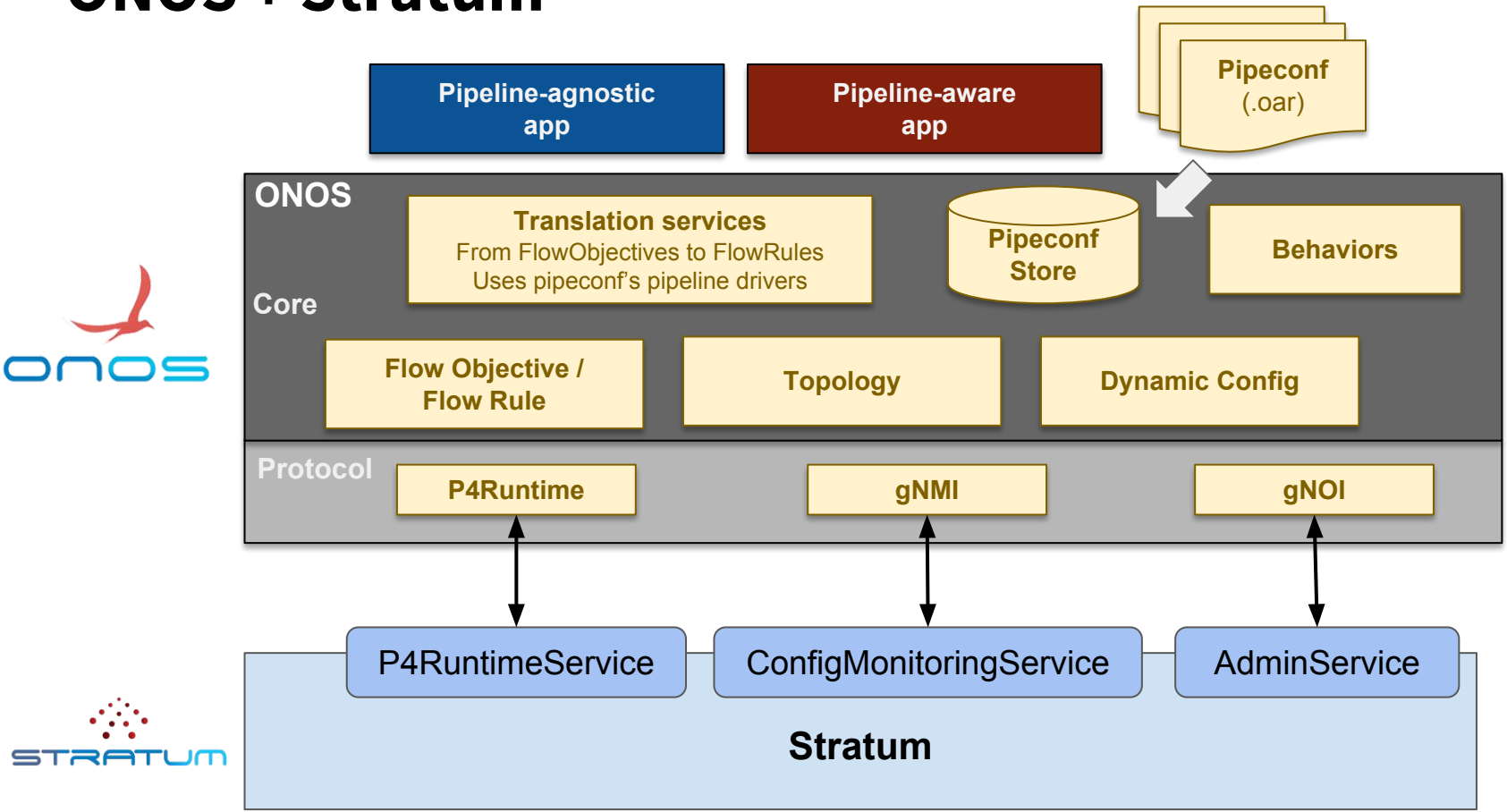
Facilitate Migration of Existing Services



The Next Generation SDN picture



ONOS + Stratum

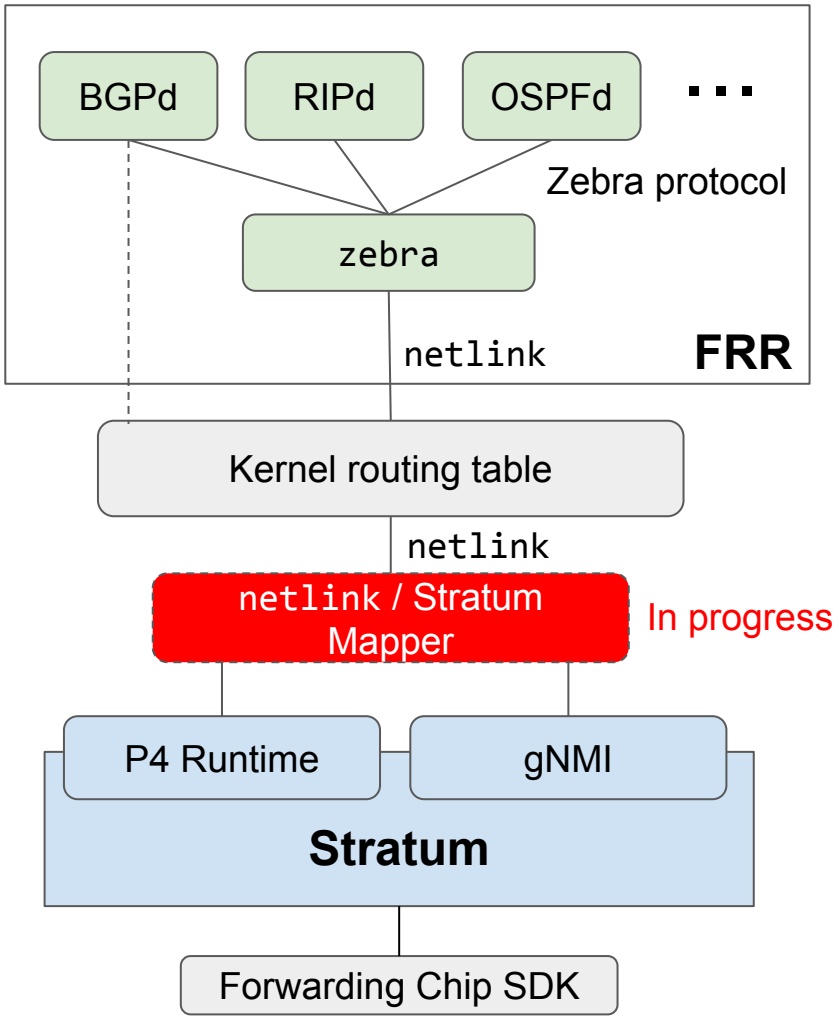


FRR + Stratum

FRR provides the IPv4 and IPv6 routing / control plane stack

Stratum provides the dataplane agent

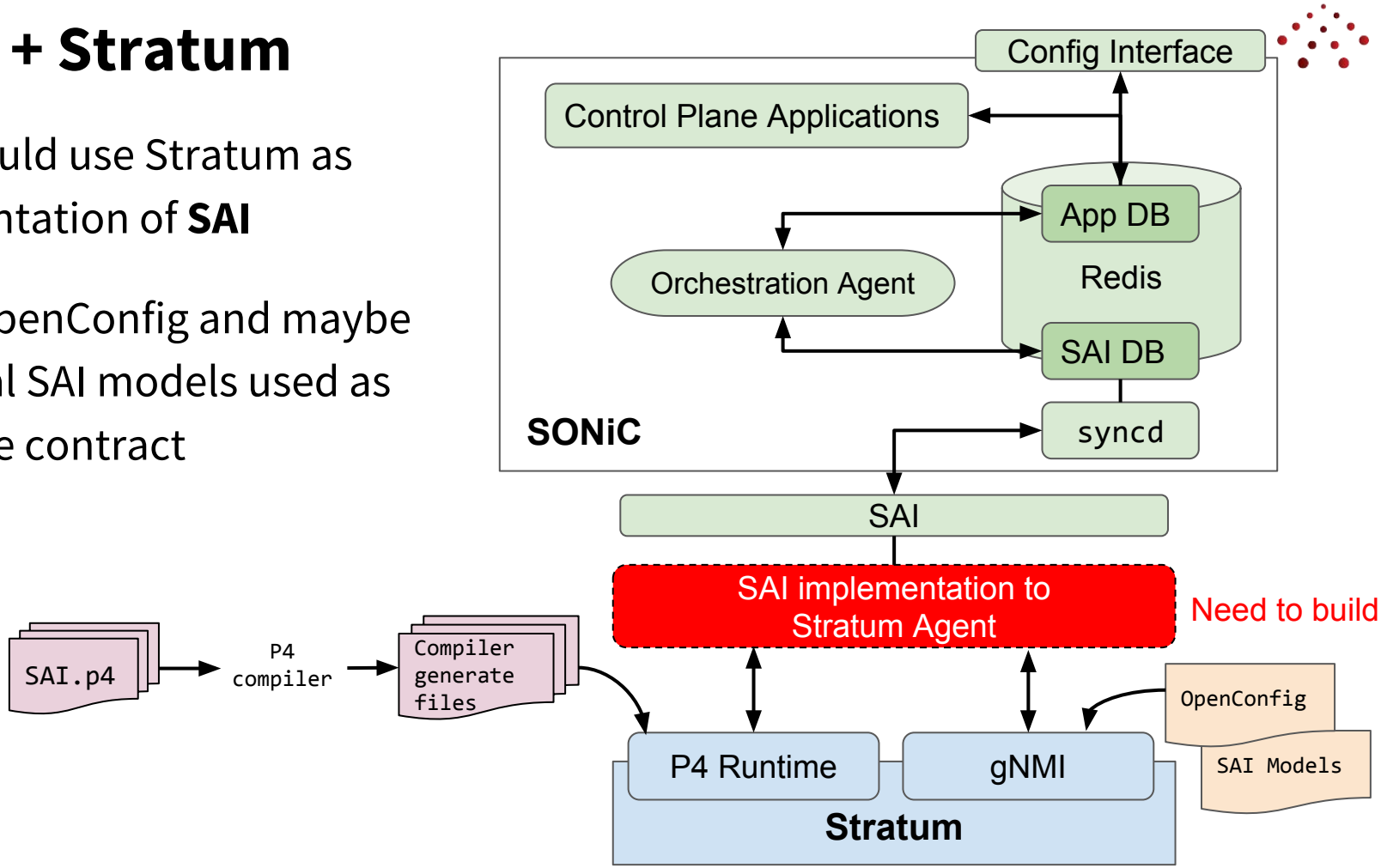
Community is currently working on **`netlink`** to Stratum mapper



SONiC + Stratum

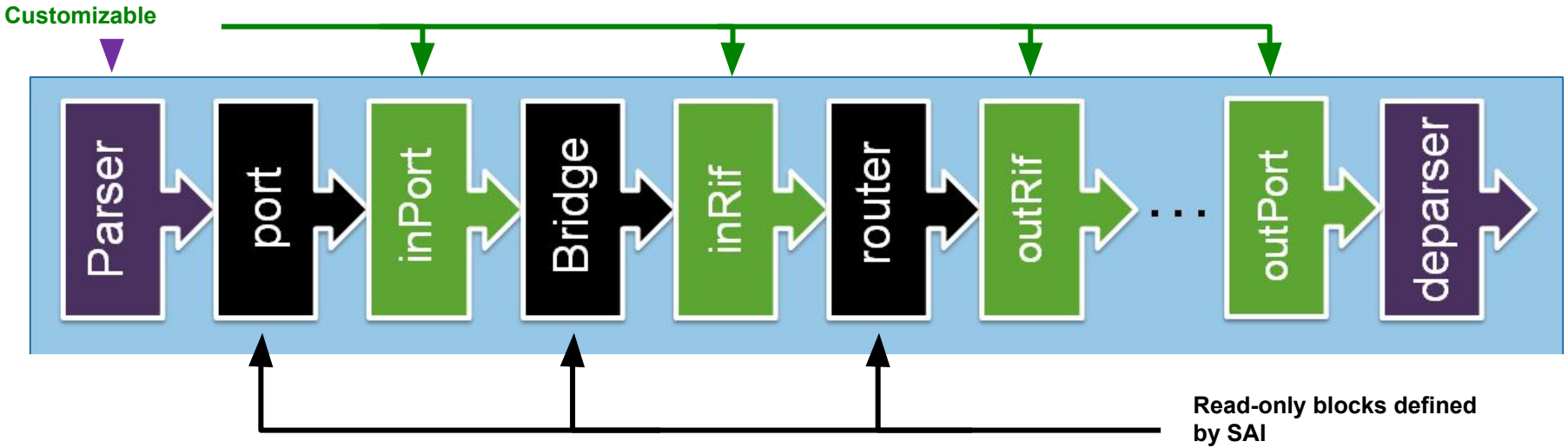
SONiC could use Stratum as implementation of **SAI**

SAI.p4, OpenConfig and maybe additional SAI models used as dataplane contract



SAI Flex

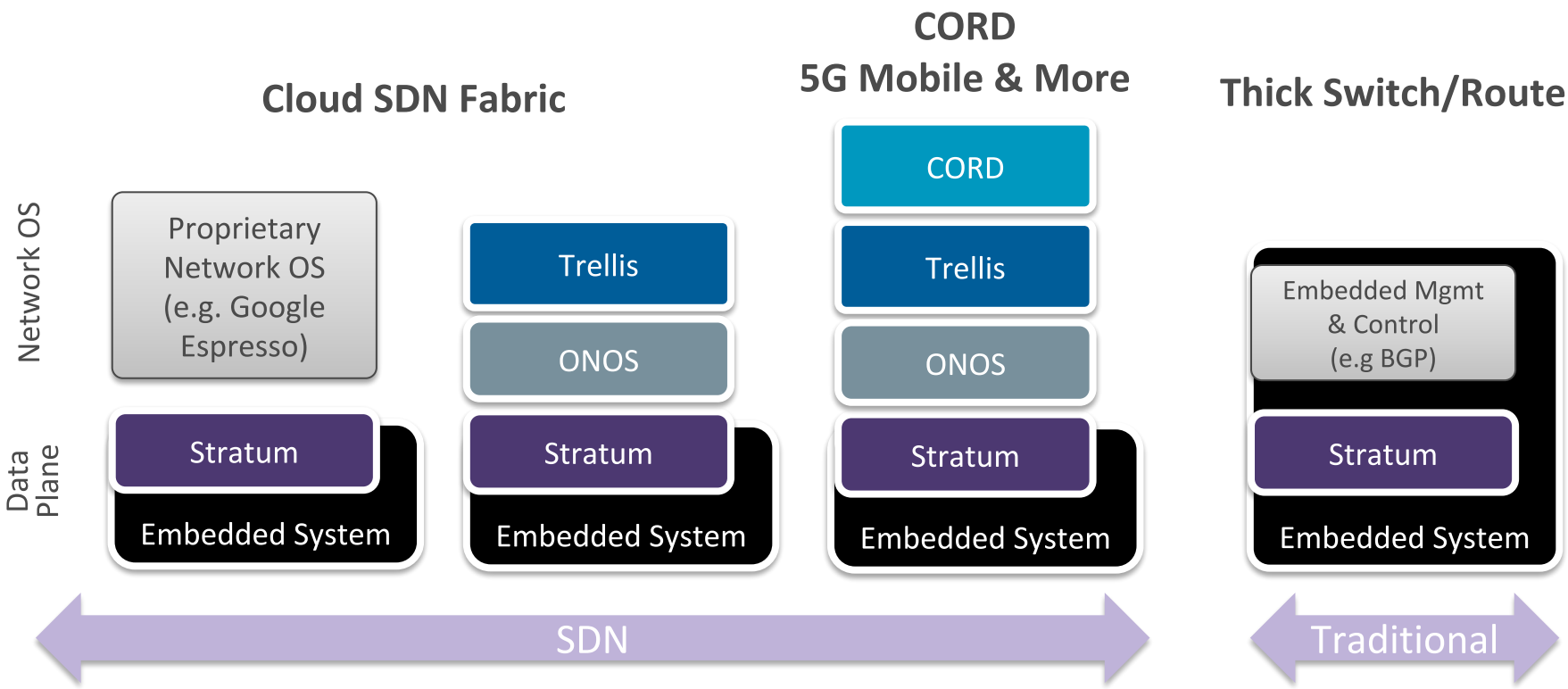
Customer-specific extensions can be expressed using P4 and exposed directly using P4 Runtime interface



For more details on SAI Flex:

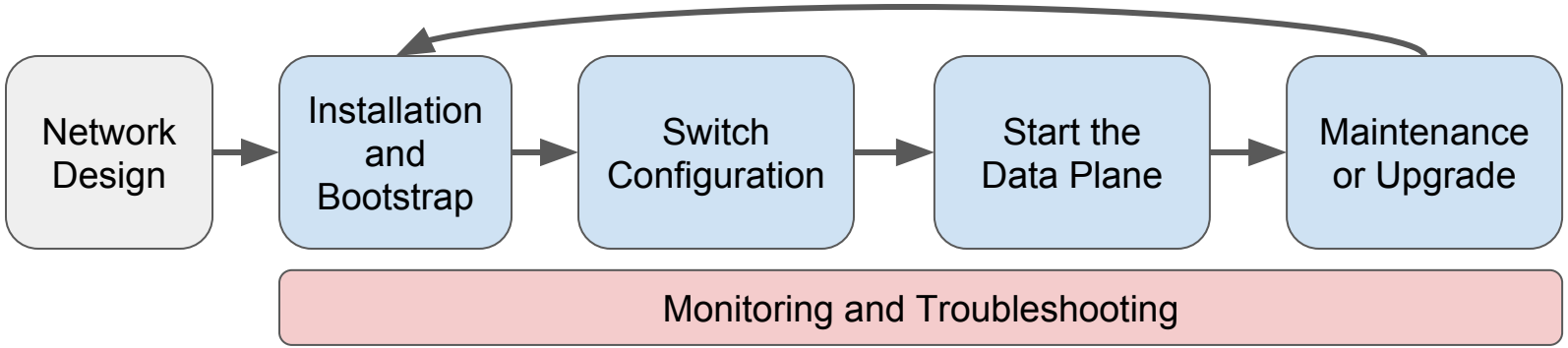
https://github.com/opencomputeproject/SAI/tree/master/flexsai/p4

Stratum Use Cases



Life of a Whitebox Switch: Day 0 to Day N

1. Design
2. Installation & Bootstrap
3. Switch Configuration
4. Start the Data Plane
5. Monitoring & Telemetry
6. Reboot
7. Upgrade

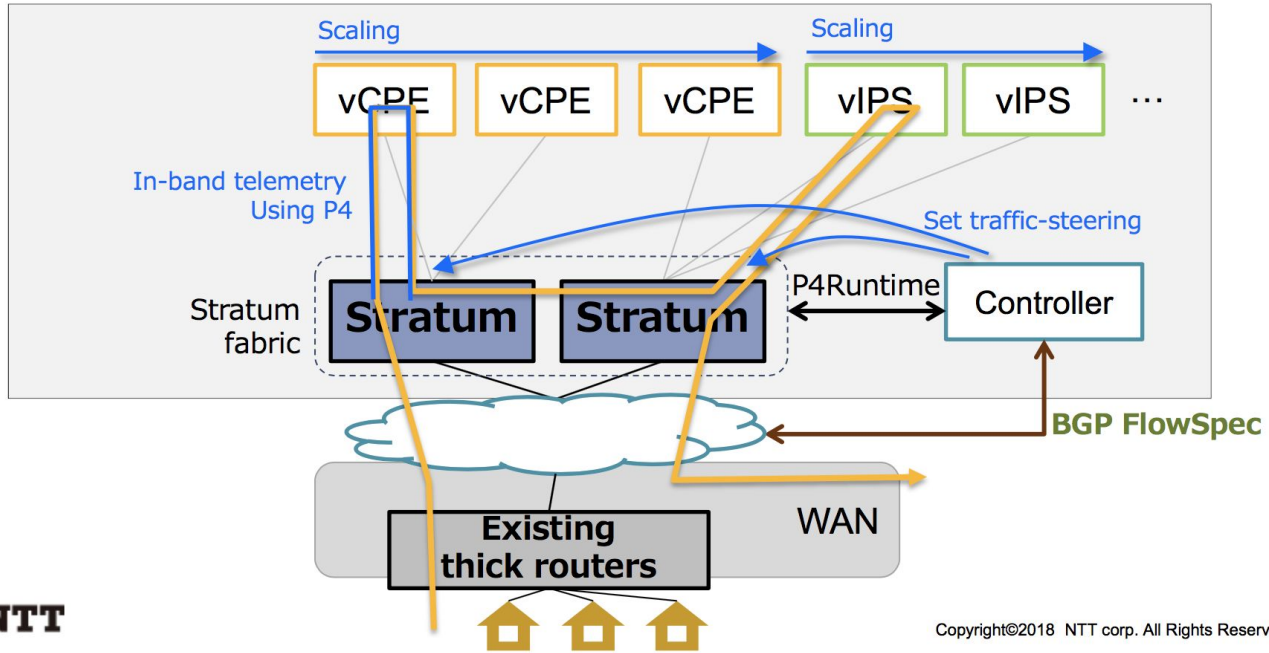


Slide from NTT

Use-case 1

Chaining and Scaling Edge Gateway

- Flexible traffic chaining with BGP FlowSpec
- Auto chaining/scaling
- In-band telemetry between VNFs

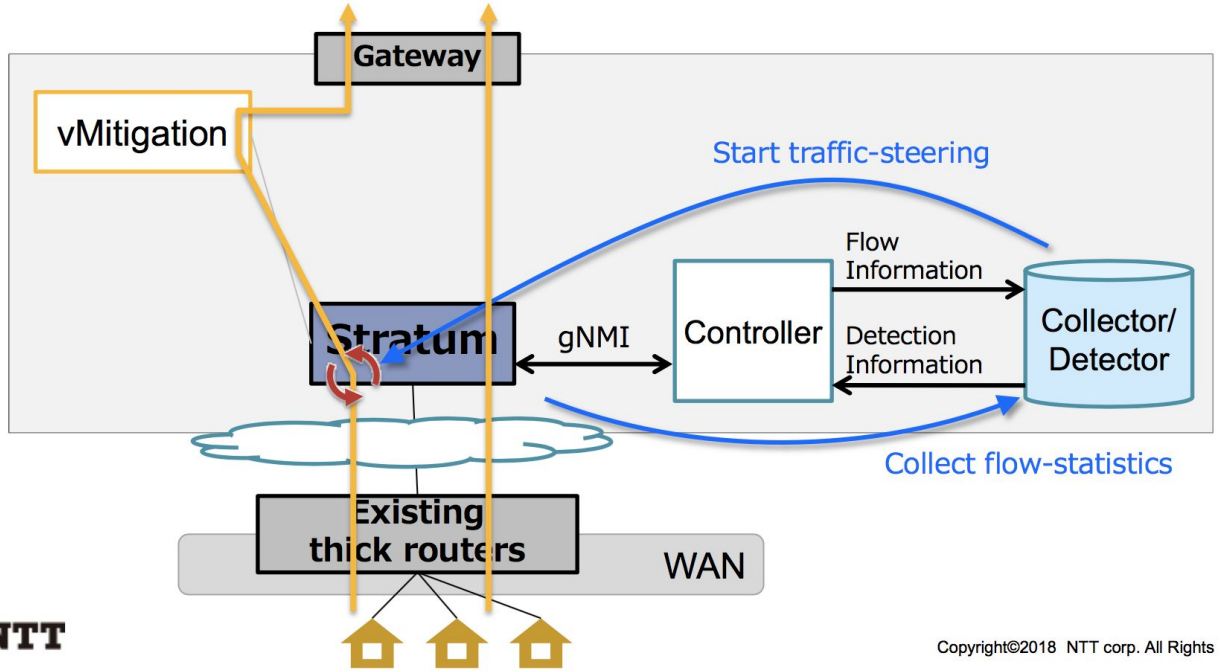


Slide from NTT

Use-case 2

DDoS Detection and Steering Function

- Collect flow-statistics from stratum switches
- Steering traffic to mitigation function when collector detects flow burst

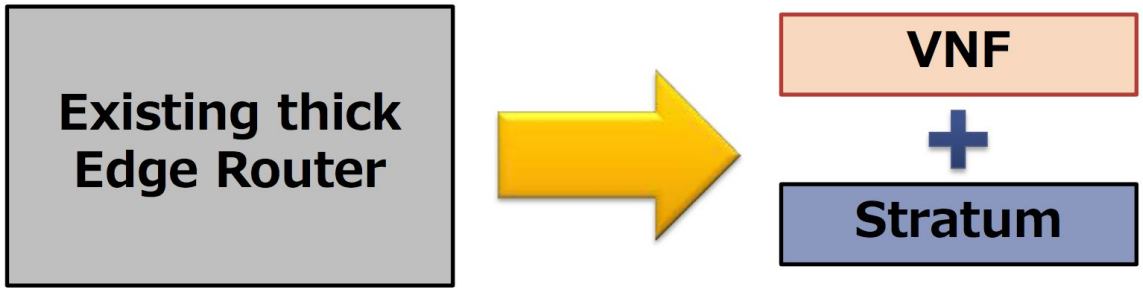


Slide from NTT

Use-case 3

Edge Router on Fixed Networks

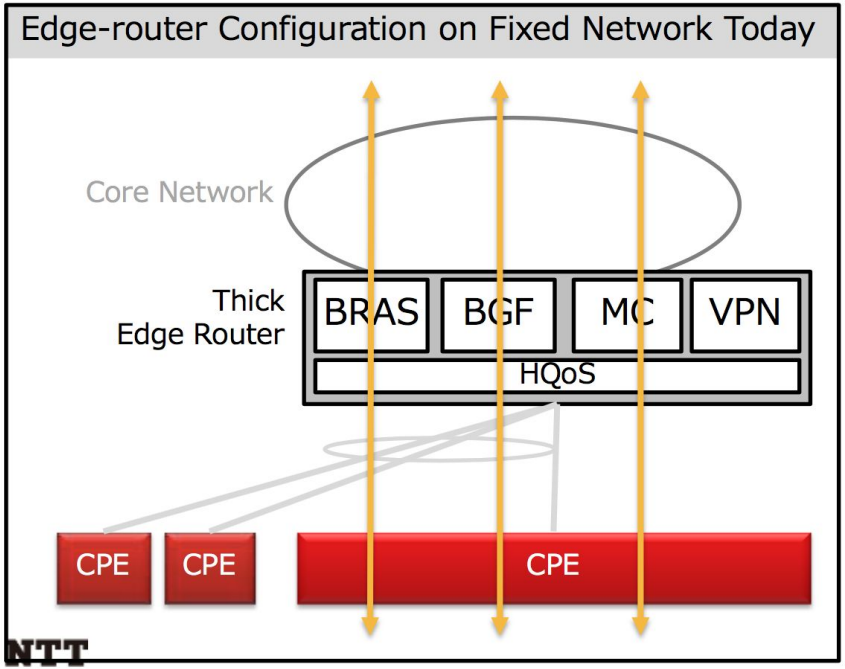
- There are thousands of NTT buildings that has the edge-router(s)
- Can edge-routers be replaced by Stratum?



Slide from NTT

Today's Service Edge Router

- **Edge-router contains service functions (BRAS/BGF/Video-Multicast/VPN-GW…) and Hierarchal QoS function**



Service Functions

BRAS:
- PPPoE termination
- AAA(Radius)

BGF:
- NAPT
- Flow-based shaping
- Diffserv

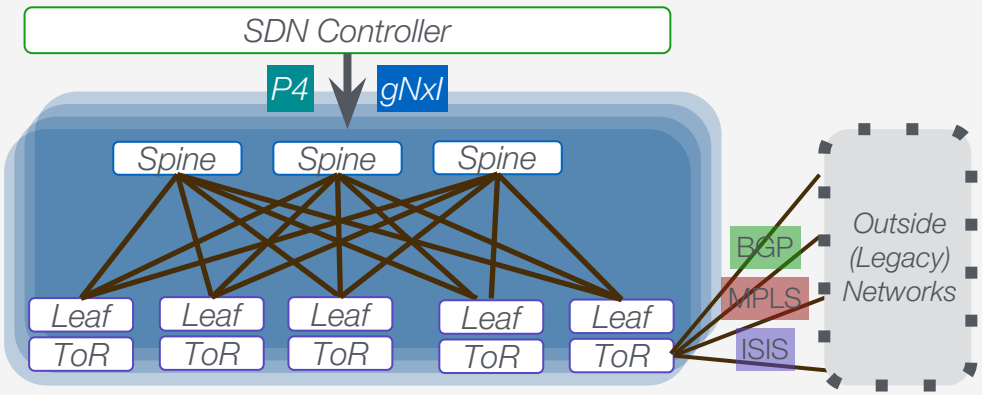
MC(Video Multicast):
- PIM/MLD
- IP Multicast

VPN-GW
- Tunnel termination
- Dynamic routing

Transforming Tencent's Network: One Datacenter at a Time

- Data center fabric as disaggregated modular switch

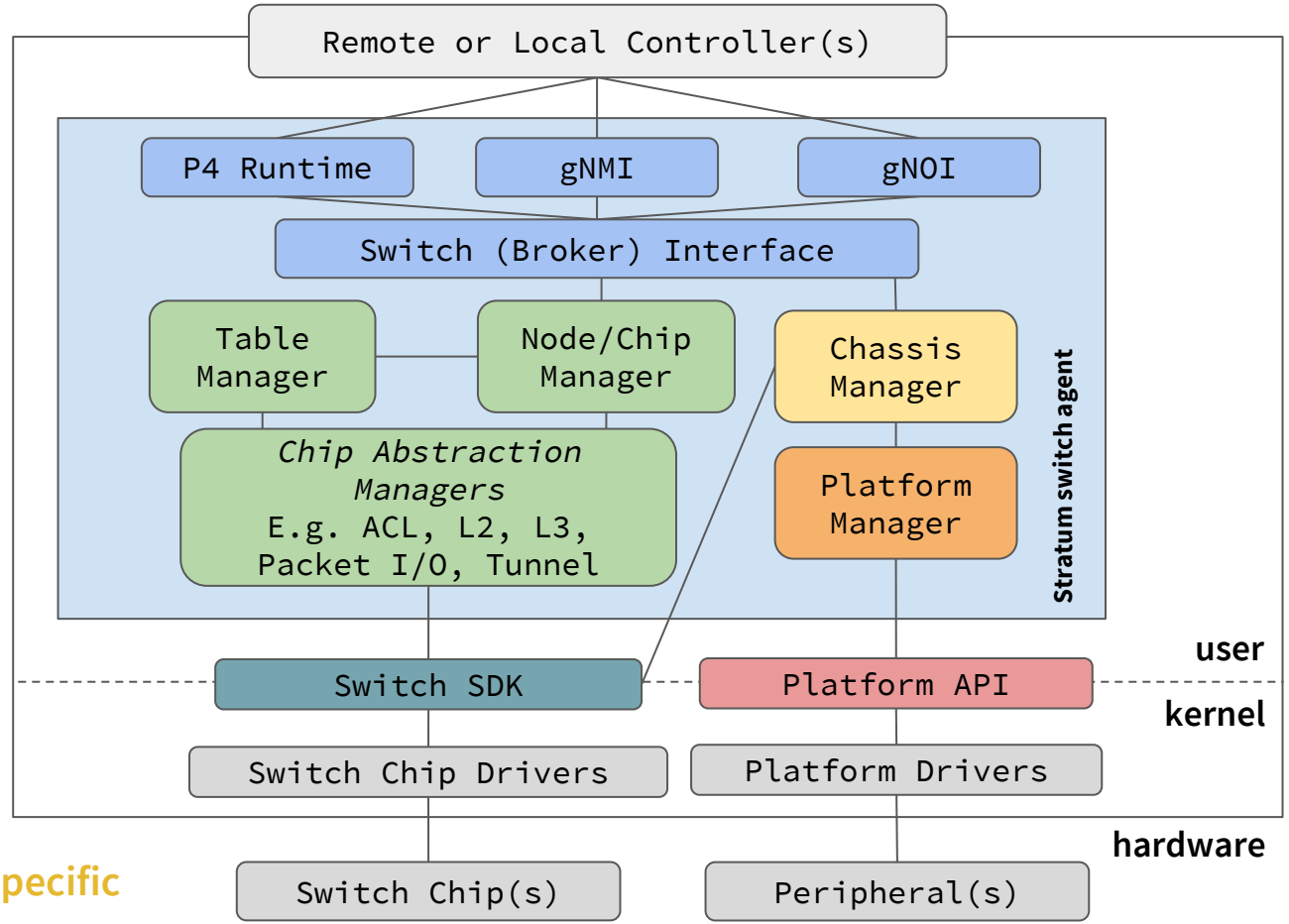
Switch OS	→	Data Center SDN Controller
Fabric Cards	→	Spine/Core Switches
Line Cards	→	Leaf/ToR Switches



Data Center Fabric
behaves like one network element

- Centralized control does not mean the entire network must have one controller.
- Rather we opt for a network of controllers, enabled by ONF CORD, Trellis and Stratum.
  - Freedom to use different protocols or RPC at outside controllers.
  - Facilitates integration with legacy networks.

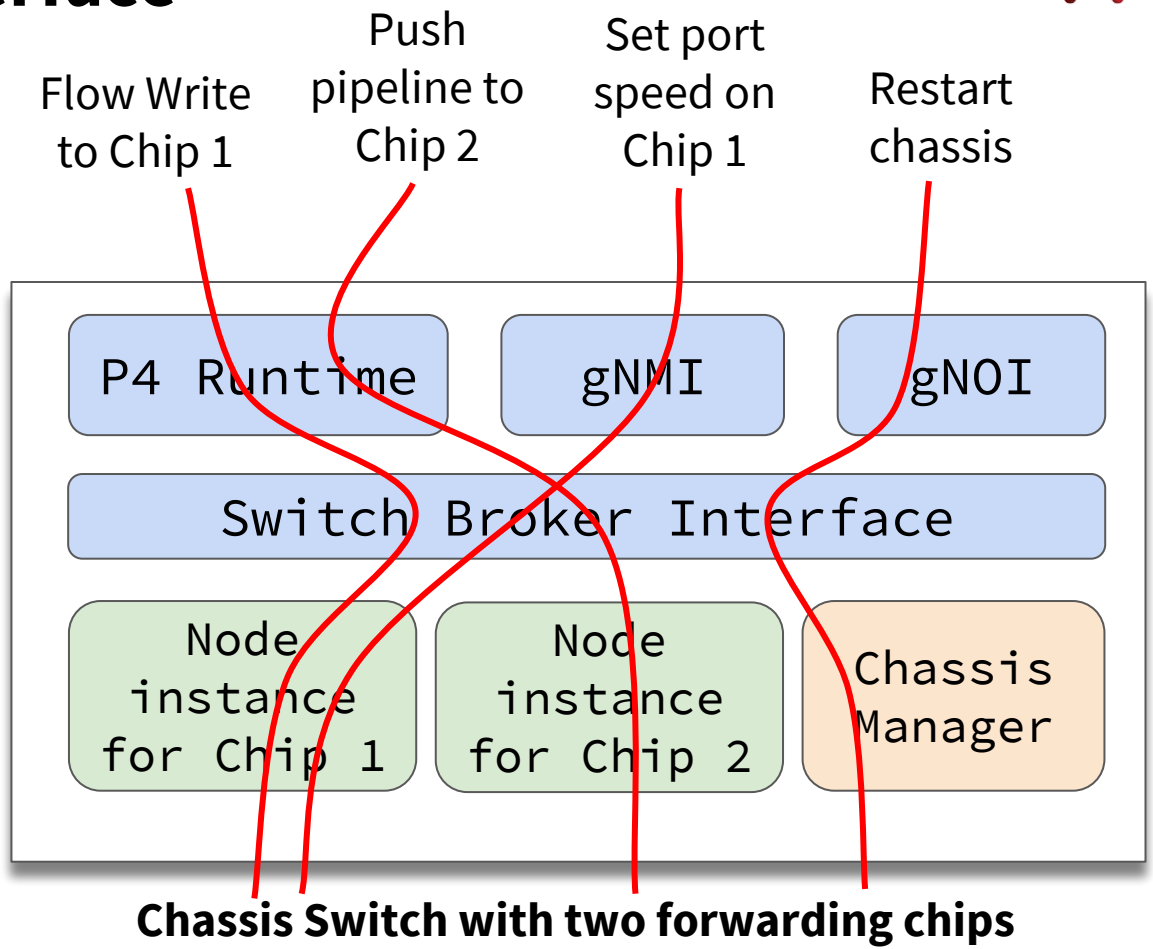
Switch Agent Architectural Components



Common
Chip specific
Platform specific
Chip and Platform specific

Switch (Broker) Interface

- This is NOT an abstraction like **SAI**
- Transparent broker interface between **P4Runtime** / **gNMI** / **gNOI** to vendor-specific managers



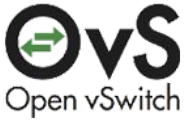
Stratum Implementation Details

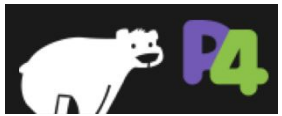
- Implements **P4Runtime**, **gNMI**, and **gNOI** services
- Controlled locally or remotely using **gRPC**
- Written in **C++11**
- Runs as a **Linux** process in user space
- Can be distributed with **ONL**
- Built using **Bazel**

Project Genesis

Stratum Community

Welcome





Stratum Development Timeline

Pioneer Phase
- Initial Reference Platform Support (HW & SW)
- Development Infrastructure (Build, CI, etc.)

Field Trials, Production Deployments
on cloud and telco networks

Stratum Member Preview
- Expanded platform support
- Feature development
- Hackathons

Open Source Launch
with forwarding chip and platform support for every vendor member

Community Development

Project Status

- Currently working toward Stratum Member Preview
- Reference platforms from 2 ASIC vendors and 3 platform vendors
  - ASIC Vendors: Barefoot, Broadcom
  - Platform Vendors: Delta, Edge-core, Inventec
- Expect support from another 2+ ASIC vendors and 4+ platform vendors in 2019
- Since April,
  - 6 new member companies
  - 2 code releases to Stratum pioneers (3rd by end of the year)

Code Releases

	Release 0.1 (May 2018)	Release 0.2 (Oct. 2018)	Release 0.3 (Dec. 2018)
P4Runtime	Support for pre-release	Support for 1.0.0-rc1	Support for 1.0 and minor fixes
gNMI	Basic framework	Stable support	Stable support and bug fixes
gNOI	-	Initial interfaces	4 service implementations (e.g. system, file)
Switch support	Google platforms; Partial Broadcom support	Barefoot Tofino on 3 vendors; BMv2 software sw.	Tofino platform integration; **`DummySwitch`** for testing
Platform abstraction	Basic interfaces	Support for platform mapping and DB	Add support for ONLP
Conformance Testing	-	Test framework definitions	Test framework definitions

Roadmap for 2019

1. Open Source Stratum Release
   - Build community and increase chipset/platform list
2. Production Deployments
   - Google
   - ONF's CORD with major operators
3. Synergy with open source Switch OSes and controller planes
   - e.g. ONOS, SoNIC, DANOS, OpenSwitch, FRR

Stratum Talks @ ONF Connect

Next-gen SDN track

Tuesday, December 4

2:00pm	Stratum - Overview, Interfaces, and Architecture
3:00pm	Leveraging Stratum to Achieve Fast P4 Program Swap on Tofino
3:30pm	Delivering the disaggregated Whitebox solution with Open Source
4:30pm	Trellis: A Production-ready Open-source Networking Fabric
5:30pm	Update of New IP Fabric Architecture and Expectation for P4

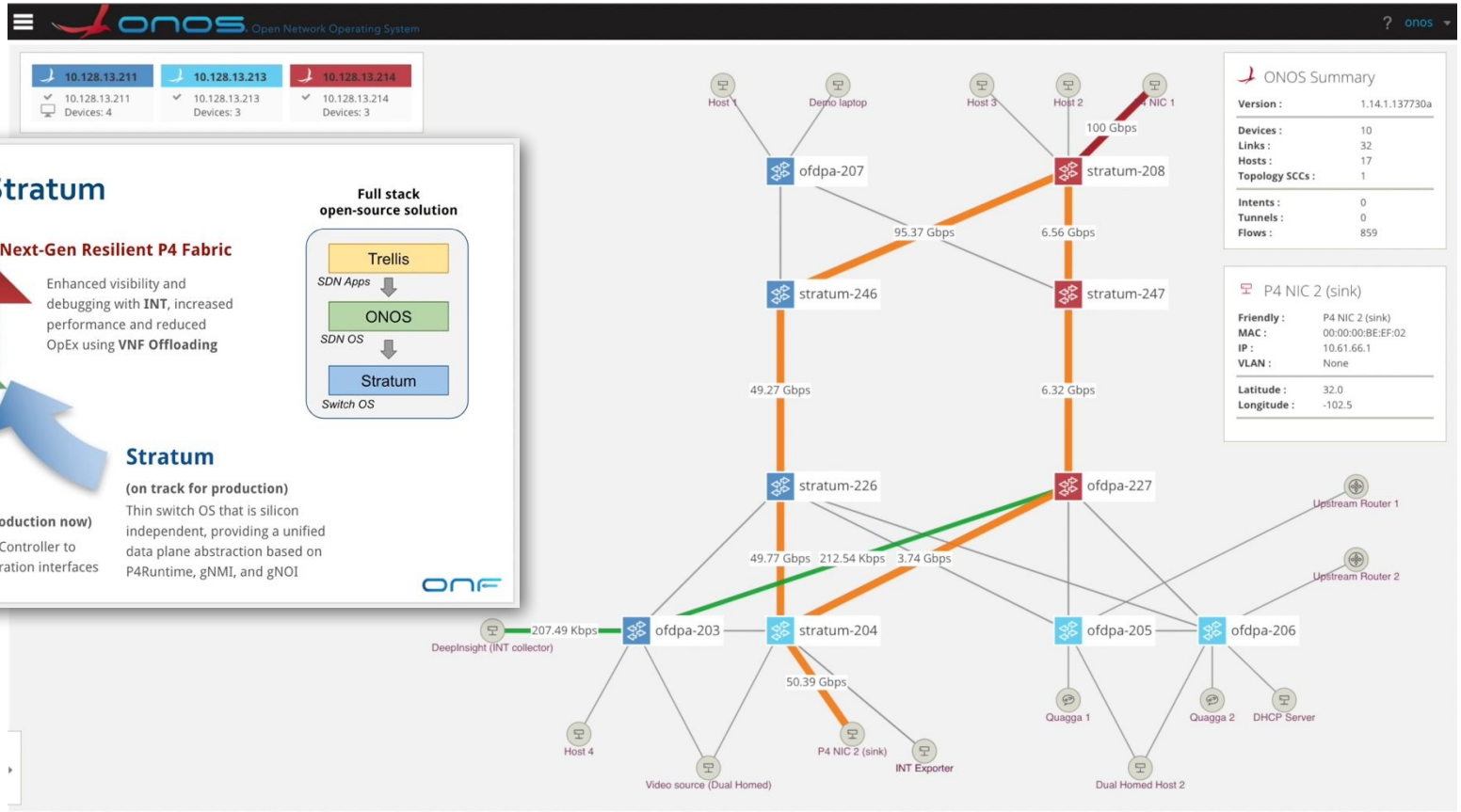
Wednesday, December 5

2:00pm	Flightplan: Dataplane Disaggregation and Coordination for In-network Computing
2:30pm	Black Box Testing of Stratum Enabled Switches
3:00pm	Implementing the Programmable Service Edge
3:30pm	P4 and Stratum Use Case for New Edge Cloud
4:30pm	Accelerating VNF Data Plane in FPGA Based P4-programmable Acceleration Card
5:00pm	Leverage Stratum and P4 to Achieve Granular Per-packet Telemetry and Visibility
5:30pm	Using INT to Feed the ML Monster

Thursday, December 6

2:00pm	Vendor Agnostic Pipeline for ONOS and Stratum
2:30pm	daPIPE: a DAta Plane Incremental Programming Environment
3:00pm	Strategy and Use Cases For Incremental Insertion Of Stratum/P4 In Field-Deployed Network Devices
3:30pm	TACOS-WR: The (Almost Completely) Open Source Whitebox Router
4:30pm	DANOS Architecture, Interfaces and APIs Overview
5:00pm	PANEL: Innovation + Open: Meeting Mobile Operator Expectations?

Come see our booth demonstration!



Getting Involved

Contribute to the Interfaces and reference P4 programs

- P4Runtime, gNMI, gNOI, and the OpenConfig models are already open source
- Fabric.p4, SAI Flex, etc.

Become a Stratum Member

1. Have a contribution plan
2. Sign the required documents

*We are still accepting hardware vendors and users (**including university students**)!*

Join the Public Mailing List

We will provide periodic updates on Stratum's progress.

For more details:

https://wiki.opennetworking.org/display/COM/Stratum+Wiki+Home+Page

https://stratumproject.org/

Bonus Slides

Which Models and Programs are supported?

P4 program

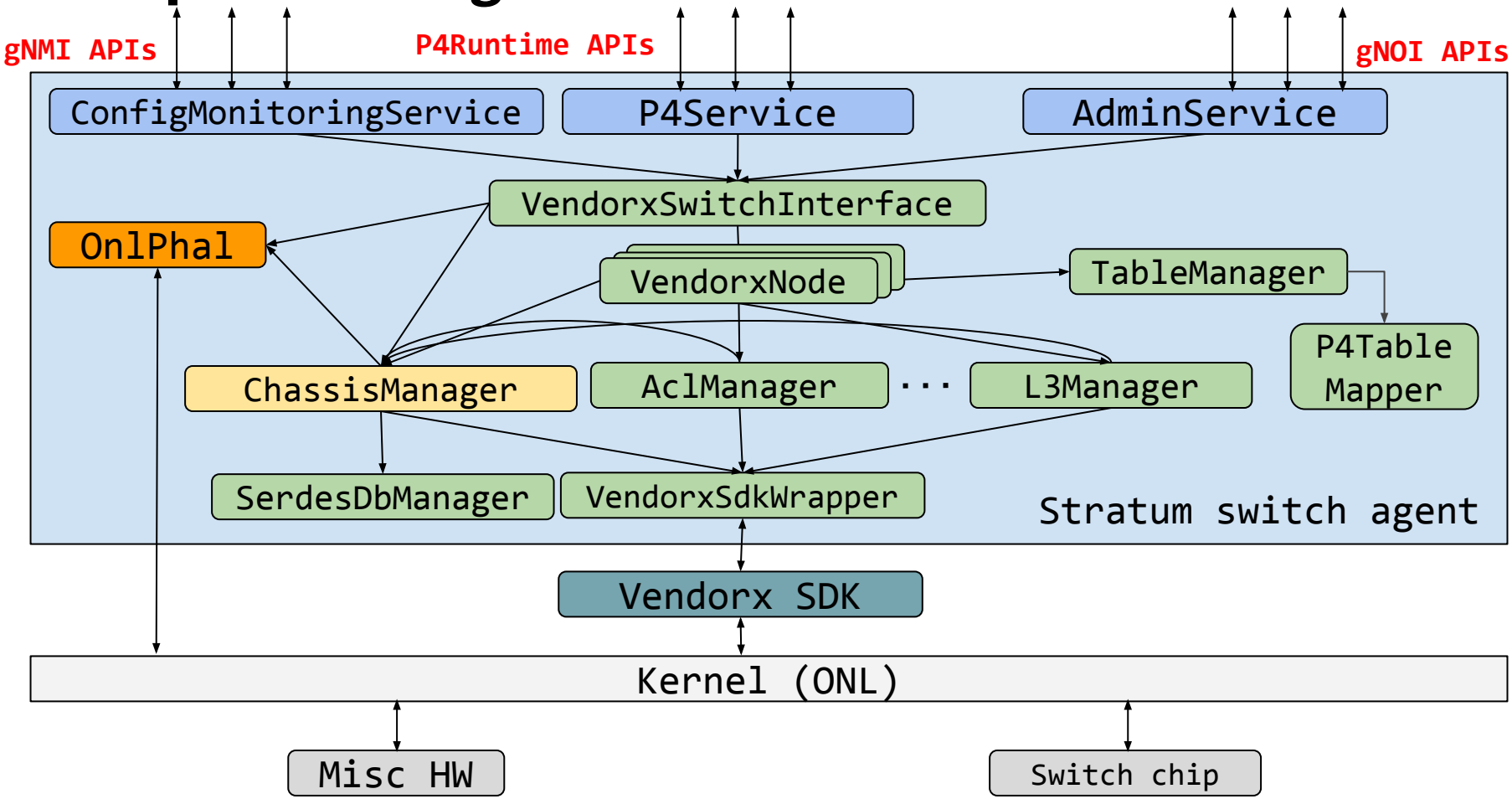
- Stratum is not tied to specific P4 programs
- Depends on what the target (and compiler) will support
- Examples:
  - Google's tor.p4 and spine.p4
  - ONOS fabric.p4
  - SAI.p4

YANG models

- Initial support for a subset of **OpenConfig**
  - interfaces, lacp, platform, qos, vlan, alarm
  - Along with some augmentations

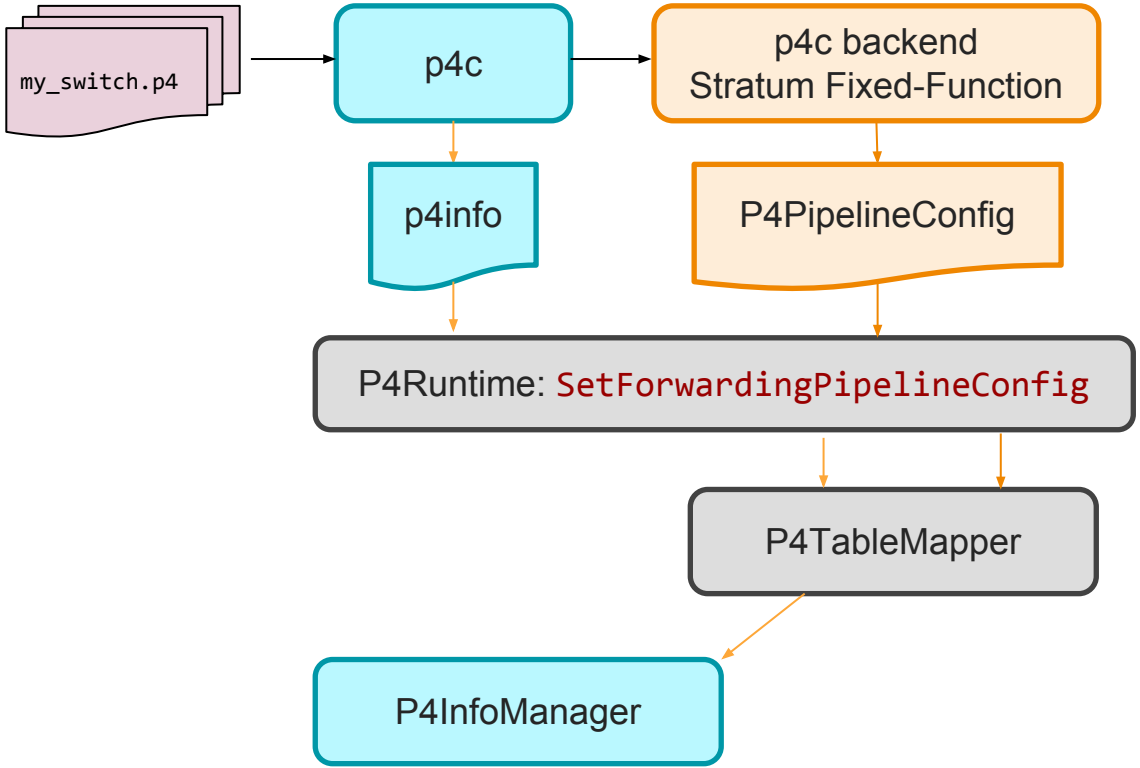
Operations (gNOI) - Initial support for *cert*, *file*, *diag*, *system*

Recap: Switch Agent Classes



Fixed Pipeline Mapping in Stratum

- Library for mapping P4 forwarding entries (e.g TableEntry, ActionProfileGroup/Member, etc) to a vendor agnostic proto format.
- Used for Broadcom implementation



Programmable Pipelines in Stratum

- Programmable devices that have PI implementations can use PI's device manager as the Node abstraction
- Basically just a shim
- Used by bmv2 and Tofino (maybe eventually Mellanox)
  - Tofino may eventually move away from PI and implement something more directly

Security -- Authentication & Authorization

- Authentication -- credential management
  - Rely on gRPC support for different ways of doing credential management
    i. gRPC allows loading different credential managers using **builder.AddListeningPort** w/o changing anything else -- so simple!
  - Vendors/companies can "potentially" have different credential manager classes
- Authorization -- per-service per-RPC authorization policy checking
  - A class called **AuthPolicyChecker** which handles reading auth policies (in form of a protobuf) from persistent storage and applies per-service per-RPC auth at the beginning of each single RPC
  - Auth policy is updated via gNOI (details are still WIP)