
OPEN
Compute Project
OCP U.S. SUMMIT 2017
Santa Clara, CA

ENG. WORKSHOP: PRES.

Applying Cloud Principles to Hardware Drivers.

Tom Anschutz, DMTS, AT&T

AT&T

Tom Anschutz

Architecture & Planning
1057 Lenox Park Blvd.
Room A473
Atlanta, GA 30319

+1.404.499.7003
tom.anschutz@att.com

OPEN HARDWARE. OPEN SOFTWARE. OPEN FUTURE.

OPEN Compute Project

Agenda

Set a Baseline for What Exists

A Distributed Driver Architecture

New System Architectures

Work in Progress

AT&T

Legacy Approach – PNF

Hardware + Software

Bundle of Hardware, Software, EMS, and Support from a vertical supplier. Highly resistive to disaggregation or component re-use

Hardware choices are made to support many, but not customized to any one customer

Firmware, features, and EMS are updated by supplier based on multi-customer business case. Often ignore single customers features or operational time frames.

Redundancy through 1-1 duplication, so dedicated backup resources.

The operator must develop operational “tooling” to cater to the operations of the box, software, and also the needs of the business.

Test tools, data collection, and planning for growth and coverage become operations and IT responsibility.

When the Internet Routing table does its “thing” we often need to buy more memory or a new CPU Card

AT&T

Current Software Stacks

Are we still thinking in the box?

Software to operate a network element is typically architected like a single server or PC-centric application.

Software shares fate with the box.

The box CPU is often selected based on initial or simple requirements

- Not always a no-regrets approach

Complex control planes or large state working sets favor larger boxes that can distribute the compute cost across many line cards

- How many 1RU Internet Routers have you seen?

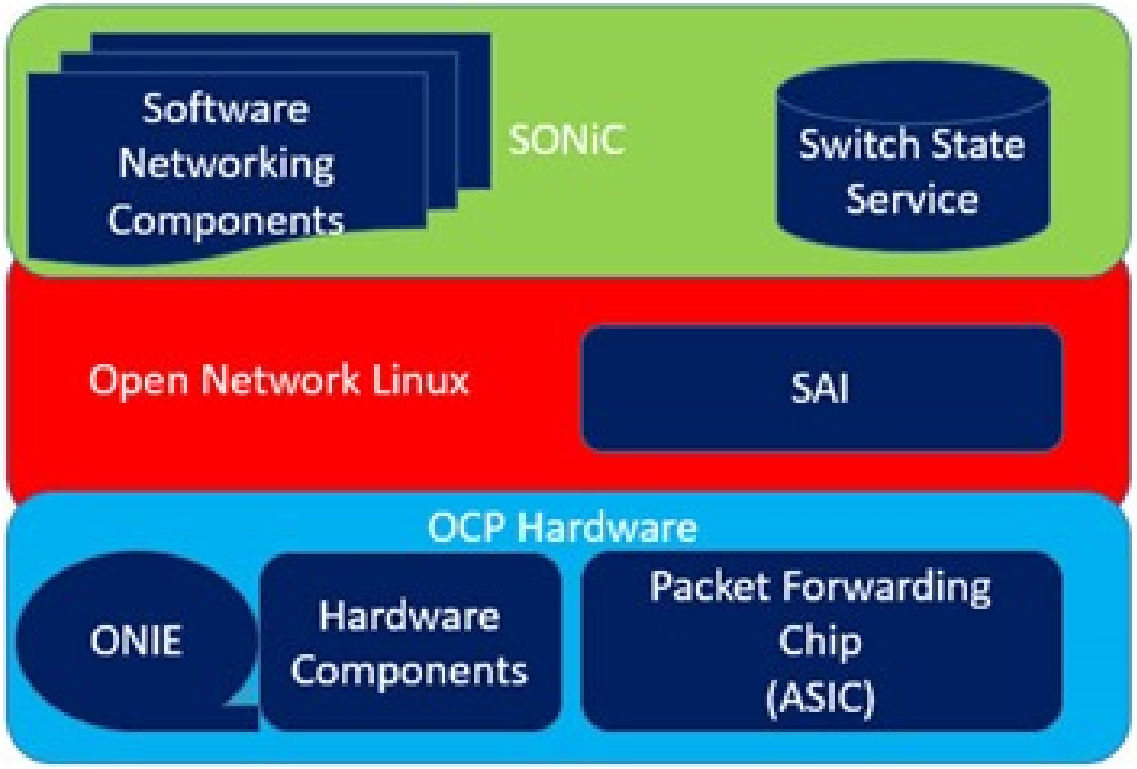


Figure used from Microsoft SONiC presentation

AT&T

SDN Controller Architecture

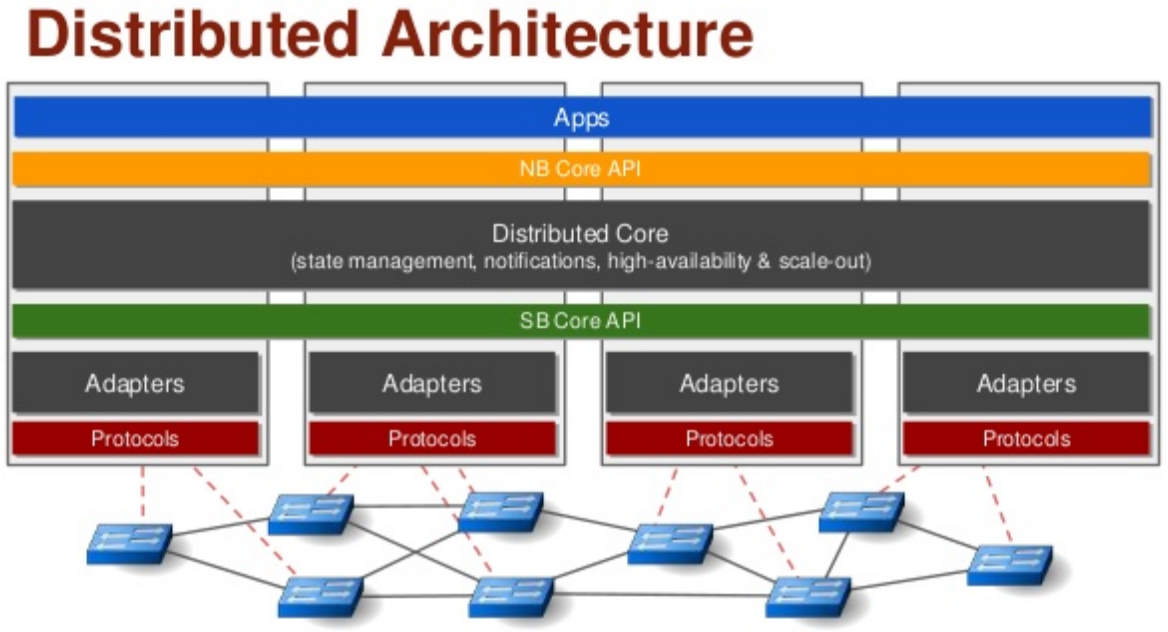
A Pattern from the SDN Controller Storybook

Core state is sharded across a distributed database

- Think Titan / Cassandra / Memcachd
- Can provide strong or eventual consistency as needed
- Shows new availability and scalability
- Small systems can be used in tandem to make big systems.

A small box, with a small CPU can be controlled to behave like a large, complex network element

The SDN Controller can remove some of the complexity in a typical network box…



AT&T

SDN Intended Focus

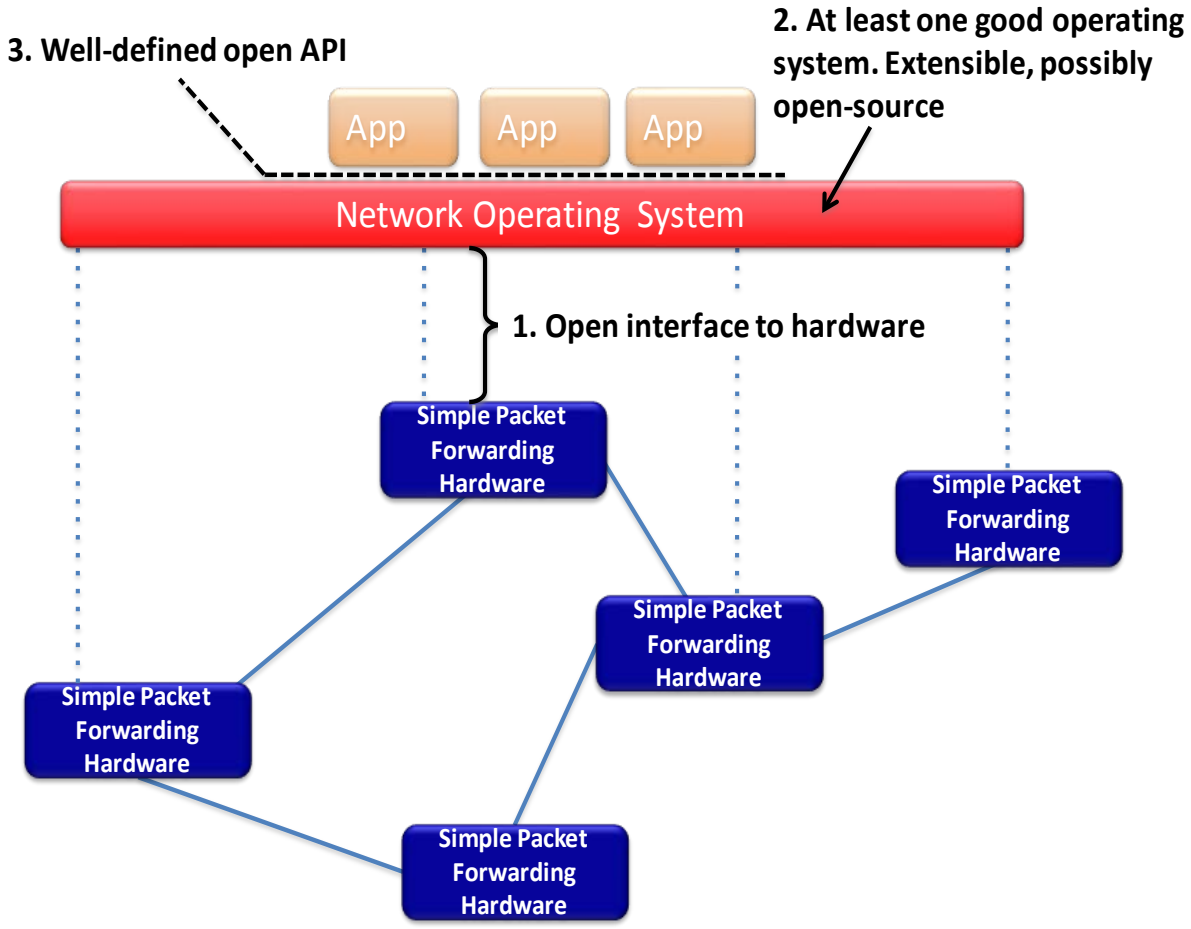
The SDN Pattern is “pretty good,” but with some drawbacks

The protocol is typically OpenFlow

- It’s pretty good for basic networking
- Doesn’t cover a lot of esoteric encaps
- Not native to many chips
- Doesn’t cover DPI, OAM, and other desirable capabilities that don’t fall into the match-action abstraction
- So may not be the best choice every time

But what if we re-focused on a lower level protocol, or allowed more protocol variations

- Is P4 a good candidate?
- Networked SAI?
- Others?



AT&T

Silicon APIs are Protocols

What does driving Silicon really require?

Typically silicon is managed through a communications path

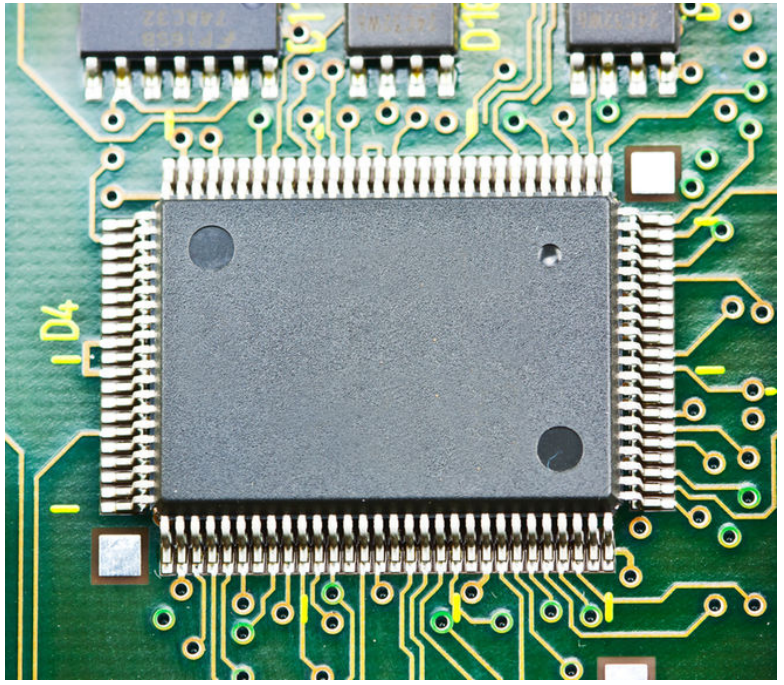
The path is usually low level, sometimes proprietary

In many sophisticated systems, the silicon is a SoC that has its own firmware or image that needs to be run.

– For these systems, configuration and control is typically an inter-process communications task.

For the silicon we've worked with for various OLT, G.Fast, and some switching devices there are typically several communications options, and Ethernet is almost always one of them.

Others include PCIe and I^2C, but Ethernet is networkable, so that's where we've focused.



AT&T

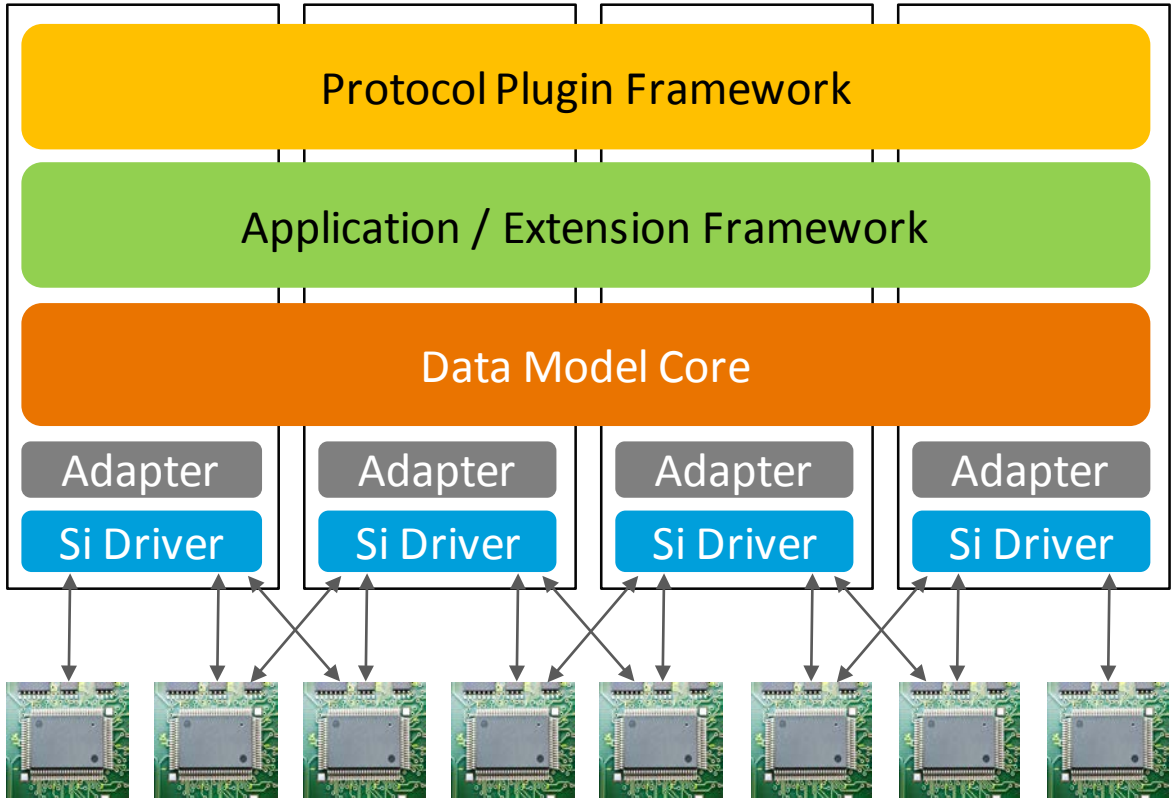
Distributed Driver Architecture

Let's use the SDN controller pattern for silicon drivers

Core state is sharded across a distributed database

- Think Titan / Cassandra / Memcachd
- Can provide strong or eventual consistency as needed
- Shows new availability and scalability
- Small systems can be used in tandem to improve availability.

The SDN Controller can remove some of the complexity in a typical network box…

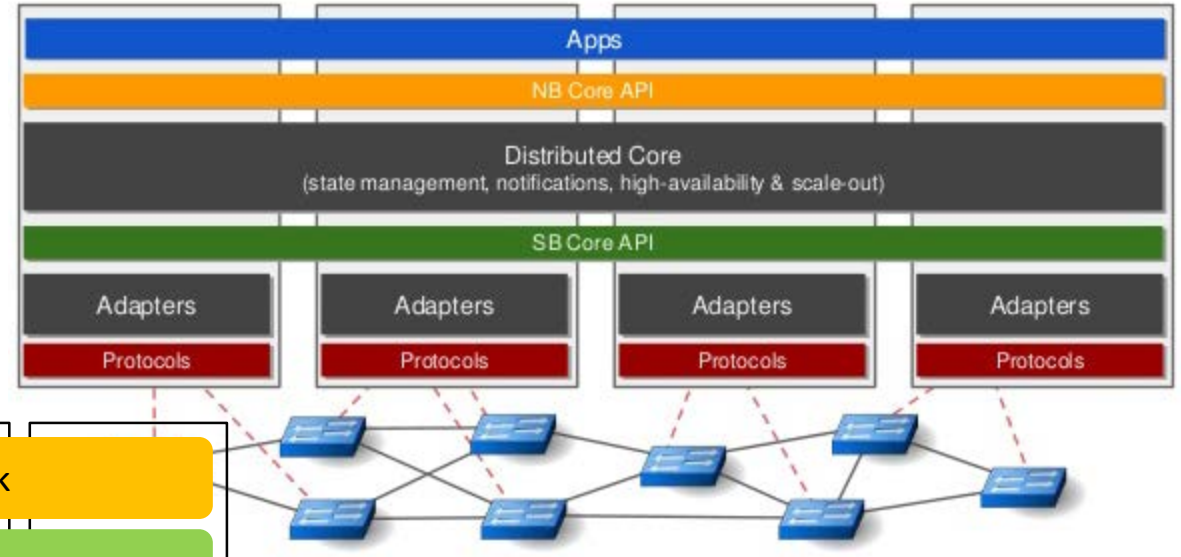


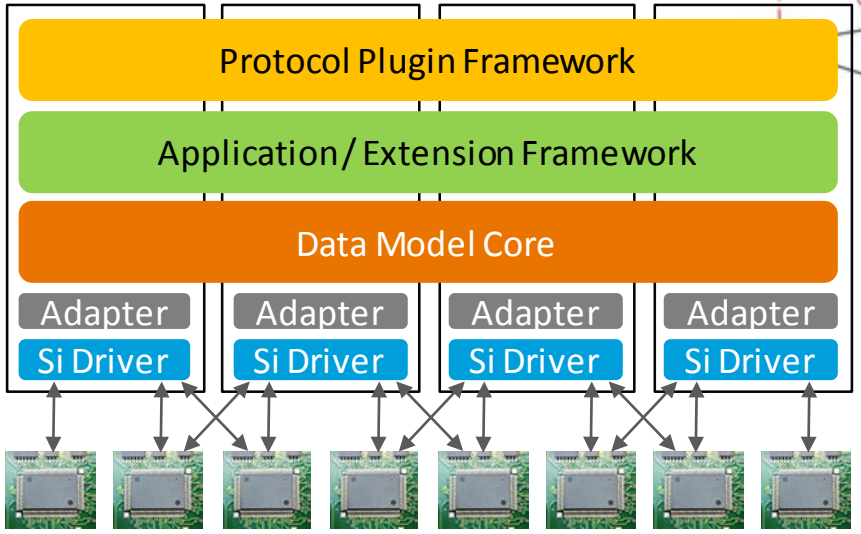
AT&T

Hybrid SDN with functional distribution options

Cloud drive can work with SDN controller to give more options for distribution

- Centralized Controllers perform macro control plane tasks
- Large scale routers, gateways, and network compositions
- Both share core technologies, but differ at layer of abstraction and scale

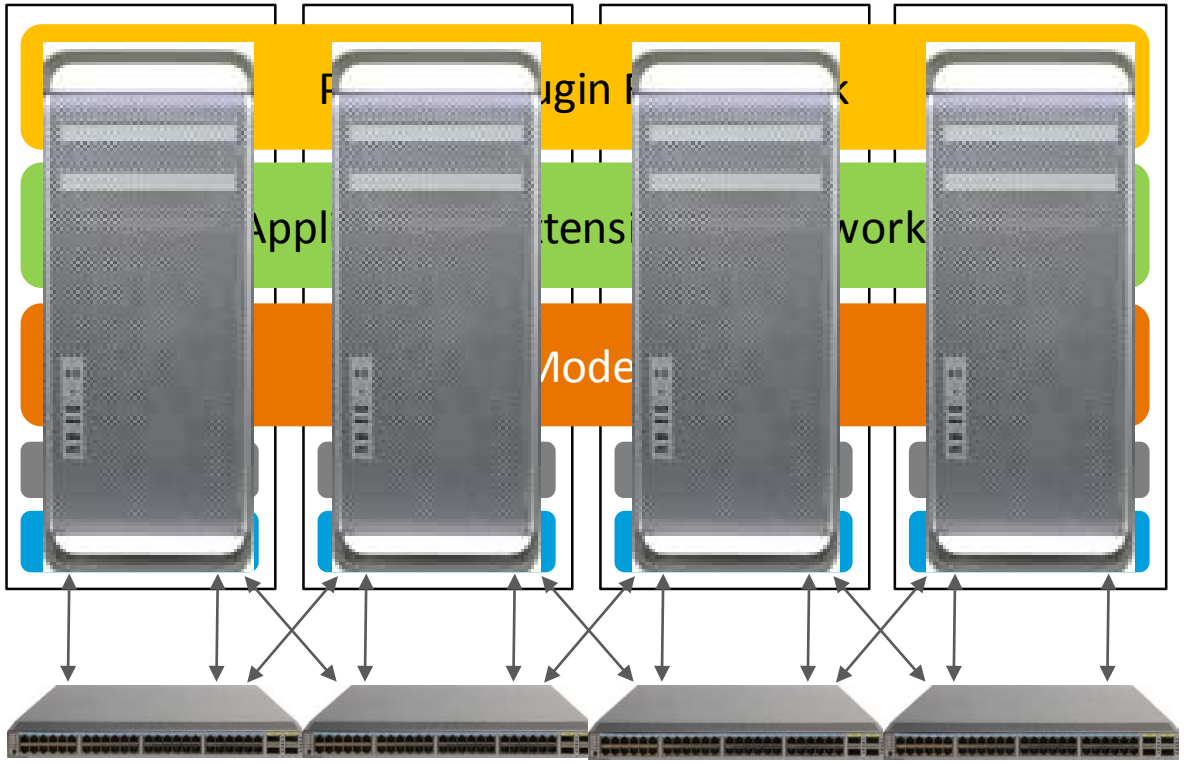


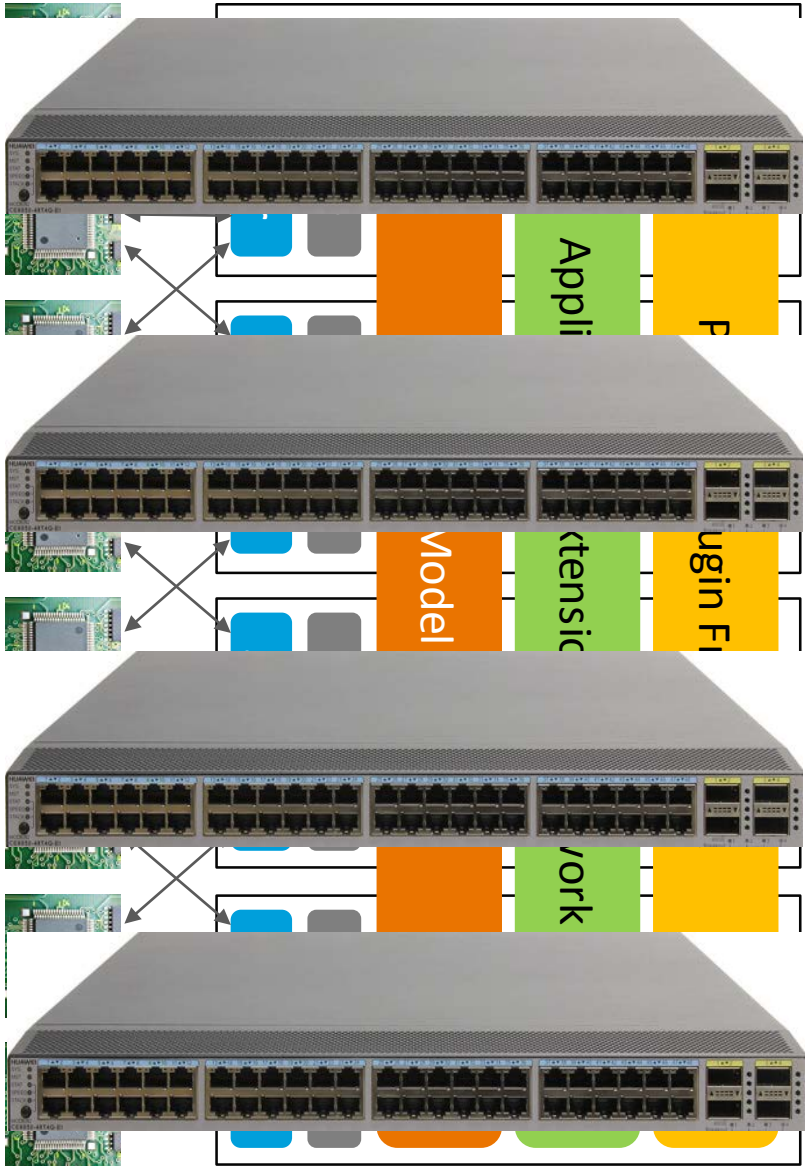


- Open Drivers perform micro control plane tasks
- OAM, link state monitoring, RAN scheduling
- Both share core technologies, but differ at layer of abstraction and scale.

AT&T

Functional Distribution Models



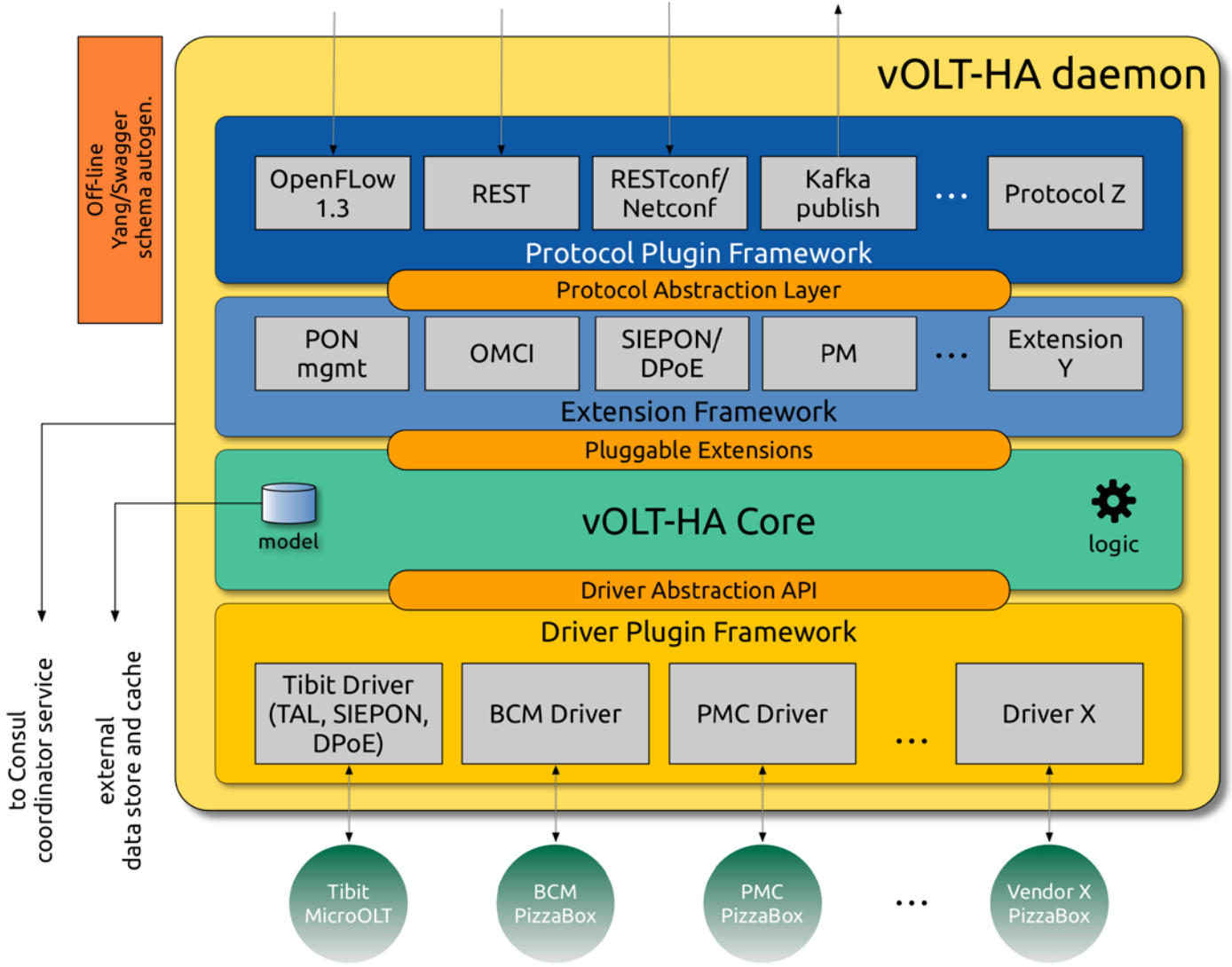


AT&T

Initial, Getting Started Work

VOLTHA

- Part of the CORD project
- Seeks to abstract and support multiple Si APIs
- Creates a common, model-driven core for state and mgt.
- Allows extending the core to apply to different access types
- Talks to the bigger system with many protocols
- Protocols are auto-generated



AT&T

In Summary

- Today network systems are self-contained
- SDN helps simplify, and shows how sharded microservices can improve scalability and availability
- Modern networking silicon requires very little local software

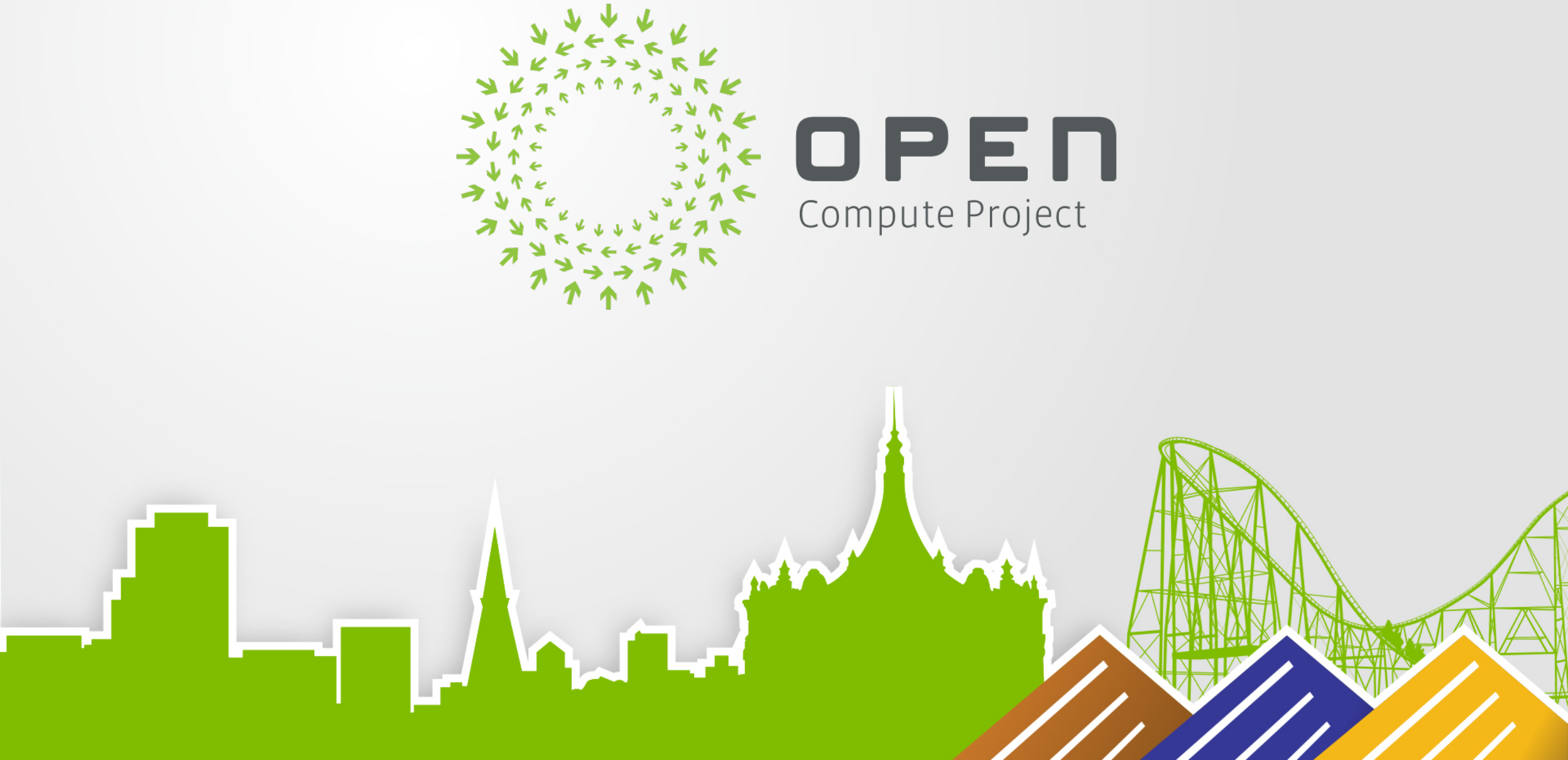
So

- Drivers can adopt SDN controller architecture to gain many of the same benefits
- This can make exterior protocols more flexible
- Disaggregated Network Functions gain distribution options

Work is going on in CORD

Thank You!

AT&T


OPEN
Compute Project