NG SDN Controller

rationale & tenets

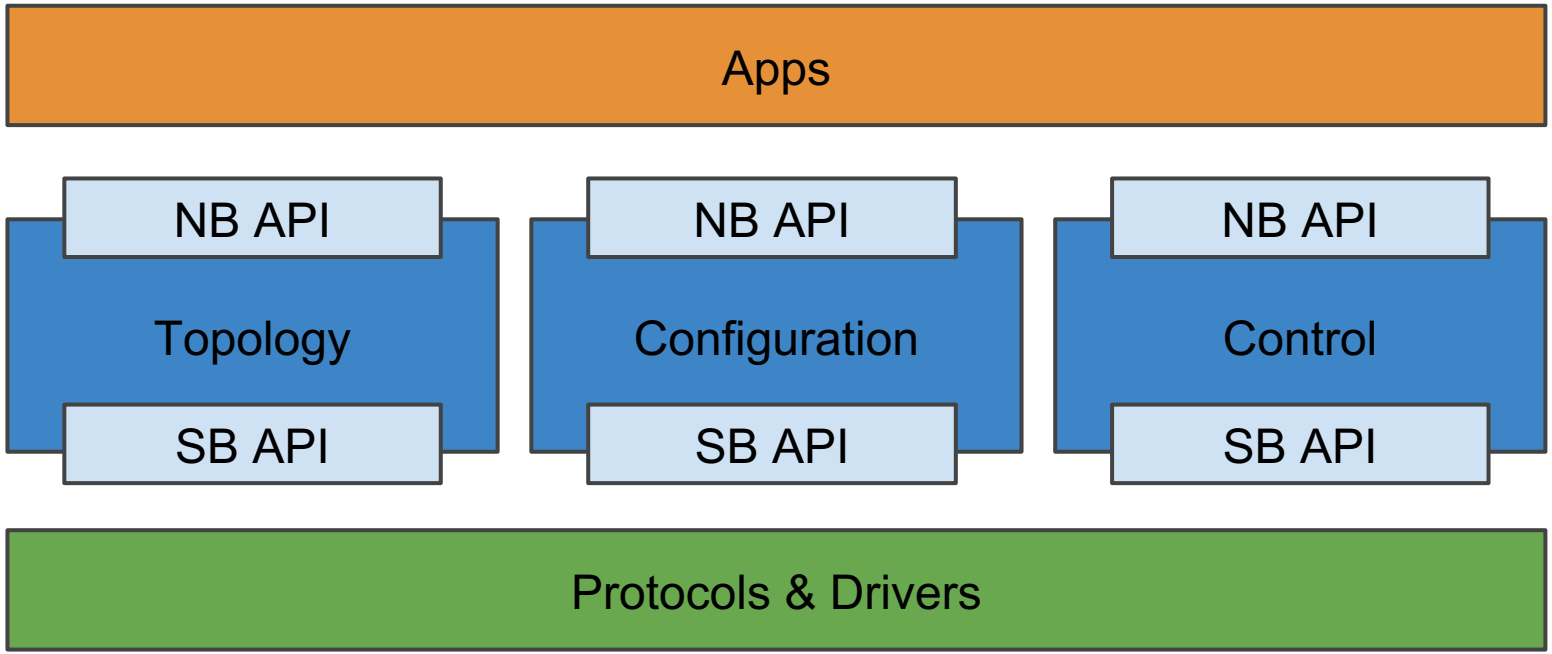
Looking Ahead

- With ONOS 2.0 being a stable platform for some time to come, now is the time to consider next generation architecture
- With Stratum starting to materialize as UPAN data plane, now is the time to consider UPAN control plane
- Goal is to establish the next generation SDN controller architecture
  - kicked off collaboration at start of 2019
  - completely in the open and with the help of the community at large
  - project name to be established by end of April
- Continue to curate ONOS 1.x & 2.x maintenance and releases
  - core team to do LTS bug fixes, code reviews and release engineering
  - community to continue new feature & applications development

NG SDN Controller Tenets

- Use gRPC-centric interfaces, standard YANG Models, P4
  - gNMI, gNOI, P4Runtime, gRIBI, OpenConfig, etc.
- Follow microservices principles
  - modularity, horizontal scaling of services, tenant apps, etc.
- Rely on existing solution orchestration platforms
  - e.g. Kubernetes, Helm charts
- Reuse code as appropriate
  - e.g. Atomix, GUI, protocol libraries
- Focus on features required for production deployments
  - live update, diagnostics, monitoring, integrations with orchestrators
- Allow components written in different languages
  - Go, Python, Java, C++, etc.

NG SDN Controller Architecture



Topology Subsystem

- Use gNMI and RFC 8345 IETF Network Topology model to exchange topology data and changes to topology state
  - allows for hierarchy of networks, i.e. virtual networks
  - can be augmented for different types of networks, e.g. optical, wireless
  - client-side libraries can be used to present language native abstractions
- Explore use of Google's Unified Network Model
  - pending information request Jeff Mogul (@Google)
  - UNM was part of his presentation at Stanford last year
- Custom gRPC API for streaming graph structure and changes
  - fallback alternative

Configuration Subsystem

- Designed as a separate entity
  - will support multi-device configuration transactions
  - will support rollback to previous points in time
  - will be integrated with existing ONOS 2.x as a transitional step
- SB API for the subsystem will be gNMI and gNOI
  - well-defined, low-profile interfaces with support for transactions
  - allows direct use with Stratum-compliant switches
  - adapters can be created for devices that do not support gNMI/gNOI
- NB API for the subsystem will be gNMI and gNOI
  - well-defined, low-profile interfaces with support for transactions
  - YANG models exposed as part of the NB API are TBD

YANG Toolchain - ygot

- Open-source, authored by Google and actively maintained
- Compiles YANG models into protobuf models
  - these can be used to generate language-specific bindings
- Use of gNMI both on the SB and NB API allows the same toolchain to be used

Control Subsystem

- New flow control abstractions
  - based on P4 concepts
  - protocol independent, but pipeline aware
  - details are yet to be determined
- SB API for the subsystem will be P4Runtime
  - well-defined, low-profile interfaces with support for transactions
  - allows direct use with Stratum-compliant switches
  - adapters can be created for devices that do not support P4Runtime
- NB API is yet to be determined

NG SDN Controller Architecture