μONOS Project Overview

Thomas Vachuska - Chief Architect
Ray Milkey - Member of Technical Staff
Open Networking Foundation
What is µONOS?

- µONOS is the next generation architecture of ONOS
- Complimentary to the existing ONOS platform
- Aims to provide a comprehensive platform for operations
  - configuration, control, monitoring, verification, live update, diagnostics
- Aims to provide first-class support for 5G RAN edge
- Based on µ-services, gRPC interfaces, next-gen SDN interfaces
  - e.g. gNMI, gNOI, P4Runtime, gRIBI, etc.
- Cloud-native (Kubernetes) and aimed at edge-cloud deployment
μONOS Architectural Tenets

• Must be modular, easy to scale down, scale out and to adapt to operators’ needs
• Use state of the art toolchain for development and deployment
• Provide native support for standard interfaces and standard models
  • P4, OpenConfig
• Continue high-performance, high-availability and scalability legacy of ONOS
  • built on next-generation of Atomix
μONOS Deployment

Device Roles (gRPC)
- gNMI/gNOI/P4Runtime

Intents (gRPC)
- gNMI//P4Runtime

P4Runtime
- gNOI
- ops
- config
- control

Device Roles (gRPC)
- gNMI/gNOI/P4Runtime

Intents (gRPC)
- gNMI//P4Runtime

SD-RAN
- gRNI

Discovered
- discovery
- various protocols

Network

k8s

k/v

app X

app Y

...
µONOS Tool Chain

- **Golang** using Go modules
- **Angular 7** for GUI components
- **Protoc** to generate Go and JavaScript from models
- **Docker** for container management
- **Kubernetes** for deployment
μONOS GitHub Repositories

- Multiple repos that reflect the component architecture
- Components built/packaged independently
- Current repos include:
  - onos-config, onos-topo, onos-control,
  - onos-gui, onos-cli, onos-test
- More refactoring to follow
- Everything hosted under https://github.com/onosproject
µONOS CI Tooling

- Cloud and GitHub based strategy - nothing self hosted
- GitHub for source code control, code review, documentation
- golangci-lint for built in linting of Go source
- Docker Hub to host container images
- Travis CI for continuous integration
- Coveralls.io for test coverage analysis
μONOS Status Update

- **onos-config**
  - implemented gNMI n/b and s/b APIs
  - multi-device configuration transactions
  - model driven, multi version support
  - rollback to previous points in time
  - device updates through subscription
  - flat storage of configuration data in k/v store
  - configuration validation against YANG models

- **onos-gui**
  - interactive configuration views
  - uses same framework as GUI2 in ONOS 2.2

- **onit**
  - integration tests suite and deployment

- **onos-topo**
  - device inventory and topology APIs
  - API design still work-in-progress

- **onos-ztp**
  - basic role-based configuration via gNMI to onos-config
  - basic table pipeline setup of devices via REST API to ONOS 2.2

- **atomix-go**
  - Go APIs for distributed primitives
µONOS Project Plans

- Continue expanding features in `onos-config`
- Begin implementation of `onos-control` to allow direct creation of flows
- Continue implementation of 5G RAN support
- Continue to track `Stratum` development
Get engaged with µONOS

• Join #micro-onos channel on onosproject.slack.com
• Attend ONOS TST meetings
  • bi-weekly Wednesdays at 9:00 PST/PDT
• Fork and send pull-requests to https://github.com/onosproject repositories
• Participate in onos-dev@onosproject.org mailing list
Thank You