CORD Roadmap
– Release Management –
When is the reference implementation released?

- Four-month cadence: January / May / September
- Mid-cycle support branches (e.g., 3.0.1)
- Mid-cycle services branches (e.g., 4.1)

What’s officially included in each release?

- Platform: XOS, Build System, ONOS (Fabric, VTN), OpenStack
- Set of Service Profiles: R-CORD (only solution included through 4.0)

Moving towards a “CORD Distribution” Model

- 5.0 will be the 5th release of the distribution
- Components will be versioned independently
Short-Term Release Plan

M-CORD
E-CORD
R-CORD
VOLTHA

1.0
2.0
3.0
4.0
4.1 (Dec-2017)
5.0

MAE

R-CORD
Priorities Going Forward

CORD 4.0 stabilized service developer interfaces

Next → Build out CORD’s service portfolio

• Upgrade all current R/E/M-CORD services to the 4.0 API
• Integrate latest access peripherals – VOLTHA and xRAN
• Expand service portfolio to include micro-services – Kubernetes
• On-Board other VNFs into CORD – ONAP
• Streamline service on-boarding based on experience
CORD 4.0 refactored build system to improve developer workflow

Next → Exploit flexibility to improve operator workflow

• Automate the build-and-install process for physical PODs, including discovery and configuration of the POD switching fabric

• Make it easy to specify (and change) service profiles independent from configuring the underlying platform

• Improve lifecycle management capabilities to include in-service-software-upgrade of the CORD control plane

• Demonstrate how CORD can leverage available infrastructure rather than require that a POD be build on top of bare metal
Multi-Access Edge Cloud

- R/E/M-CORD Services running on the same platform
- Includes VOLTHA and xRAN access peripherals

Managed White-Box OLT

- Includes VOLTHA
- Light-and-Right R-CORD (Kubernetes-based)
- Includes OSAM (Open Source Access Manager)

Integrate CORD and ONAP

- Use Case 1: OSAM
- Use Case 2: E-CORD / MSO
- Use Case 3: A-CORD / DCAE
From PoC to Trial

Proofs-of-Concept

Field Trials
From PoC to Trial

Proofs-of-Concept / Domain-Specific (e.g., M-CORD...)
  – About Possibilities –

Integrate
  – Curated by TST
  – Integrated
  – Supported

CORD Reference Implementation

Customize

Field Trials / Operator-Specific (e.g., NTT, DT,...)
  – About Reality –
From PoC to Trial

Proofs-of-Concept / Domain-Specific (e.g., M-CORD...)
  – About Possibilities –

Integrate

CORD Reference Implementation
  – Curated by TST
  – Integrated
  – Supported

Customize

Field Trials / Operator-Specific (e.g., NTT, DT,...)
  – About Reality –
From PoC to Trial

Proofs-of-Concept / Domain-Specific (e.g., M-CORD...)

– About Possibilities –

Integrate

– Curated by TST
– Integrated
– Supported

Customize

Field Trials / Operator-Specific (e.g., NTT, DT,...)

– About Reality –

CORD Reference Implementation
Build System Roadmap

Improved Development Workflow

• Easier debugging ➔ ElasticStack & structured logging
• Easier to develop use cases ➔ Decouple profiles from platform

Improved Operator Workflow

• Fast, foolproof install process
  • Install all containers from Docker Hub onto K8S
  • Install a generic CORD system, use dynamic service onboarding
• Modular, flexible CORD
  • Use existing infrastructure (e.g., OpenStack, provisioned nodes)
  • Exchange pieces of the system (e.g., K8S for OpenStack)
• Discover and configure the fabric
  • PoC script to bootstrap the fabric at install time (QA)
  • Fabric configuration based on XOS models
• Support DB migration
Deploy CORD platform components using k8s
  • OpenStack
  • XOS
  • ONOS/ONOS Apps

Support Container-based VNFs using k8s
  • Common overlay network between OpenStack VMs and Docker Containers
  • Hybrid VM-Container service platform

Demonstrate Light-and-Right CORD configuration
  • Run a configuration with k8s but no OpenStack
**XOS Roadmap**

**Improved Support for Developers**

- Static Checker for services and manifests
- Simple Synchronizer template, with most code auto-generated
- Unit test framework for new Model Policies & Sync Steps
- Dynamic service on-boarding
- Improved Synchronizer performance
- Improved performance from XOS core API
- Auto-generated test coverage to include end-to-end tests
Improved Support for Deployment Engineers

- Better contextual tracing and debugging support
- Better upgrade support, including automatic DB migration
- Dynamic service on-boarding
- Model Policy framework that is free of race conditions
- Enforce Interface models between Service Instances
- Cleaner visualization of the service graph
  - Hide platform level service
  - Filter by subscriber
  - Display synchronization status
Trellis Roadmap

Dual-homing (released 1.11.1) fixes for known issues
Pseudo-wire support
Initial QoS support (supporting network slicing)
IPv6 additional features
  • DHCPv6 relay additional features (contributed by Nokia)
  • IPv6 Multicast (contributed by Nokia)
  • V6 Router Advertisement app (contributed by Infosys)
Support for New ASICs & Bare-metal switches
  • Broadcom Qumran (QMX)
  • Cavium Xpiiant
  • Quanta switches (QCT LY8)
  • Barefoot Tofino using P4 (not in 1.12 release)
Expand test coverage to include M-CORD and E-CORD

Extend automated tests to more fully exercise the platform

• Functional regression tests – Black box tests to make sure base components have not regressed

• End to end CI/CD tests – To make sure a system can be built from scratch, deployed, and can pass a baseline of tests for both control and for traffic.

• Performance tests – So can track performance over time
  • Build out performance automation framework
  • Populate framework with a few baseline performance tests
Integrate VOLTHA into R-CORD
Continue to improve white-box EdgeCore OLT adapter functionality in VOLTHA
Explore offloading of some VNFs into hardware
  • Fast-path where traffic remains in the hardware, only go out to VNF services when you need them
  • Leverage P4-compatible hardware that is now available
  • E.g. QoS done in switches rather than vSG
Work towards multi-access edge: combined deployments of R-CORD and M-CORD
E-CORD Roadmap

Services, Services, Services
  • E.g., firewall, WAN accelerator, encryption, ...
  • Open and closed source versions

ONAP Integration
  • Replace Global XOS with ONAP

Multi-Access CORD
  • \{R,E,M\}-CORD service chains co-existing in the same pod

Device Integration
  • CPE
  • Ethernet Edge
Integrate xRAN enhancements into the CORD Platform
Expand open source availability of NGIC
Continued SDN’ization and Disaggregation of 3GPP
A-CORD Roadmap

Upgrade all current A-CORD services to the CORD 4.1 release

Integrate P4 INT enabled Trellis to A-CORD

Integrate A-CORD and ONAP DCAE including VNF onboarding alignments

Move from unstructured to structured telemetry data combining data models from OpenConfig, VES and ETSI

Create an hierarchical multi-collector architecture with dynamic collector and probe instantiations

Create a programmable SD Collector architecture along with SD-Collector development guidelines and SD Collector SDKs

Create vProbe development guidelines and vProbe SDKs