Introduction to CORD project

(Central Office Re-architected as a Datacenter)

Sangyun Han
Mobile Convergence Lab, Dept. Computer Engineering
Kyung Hee University

한상윤 석박 통합 과정
경희대학교 컴퓨터공학과 모바일 컨버전스 연구실
Email : sangyun0628@khu.ac.kr

CORD (Central Office Re-architecture as a DC)

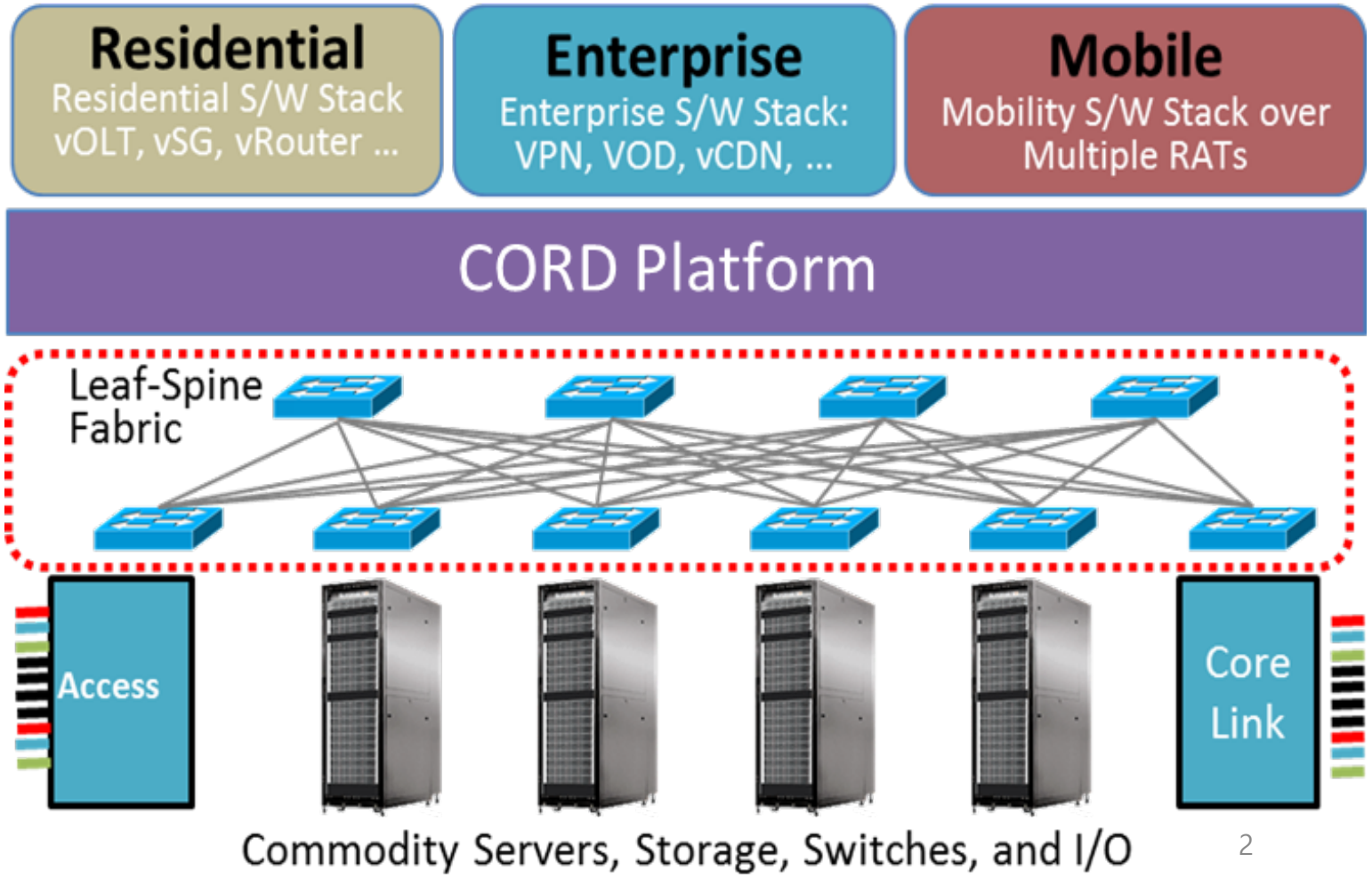
❖ **CORD is a platform that combines SDN, NFV and Cloud to deliver to Service Providers.**

▪ Economies of a datacenter

- Infrastructure built with commodity buil ding blocks using open source software and white boxes

▪ Agility of a cloud provider

- Software platforms that enable rapid cre ation of new services



Five Requirements of CORD

- **Economies of Commodity Hardware**
- **Enable Innovative Services**
- **Extensible and Controllable**
- **Multi-Domain Security**
- **Operational Robustness**

Two step of CORD

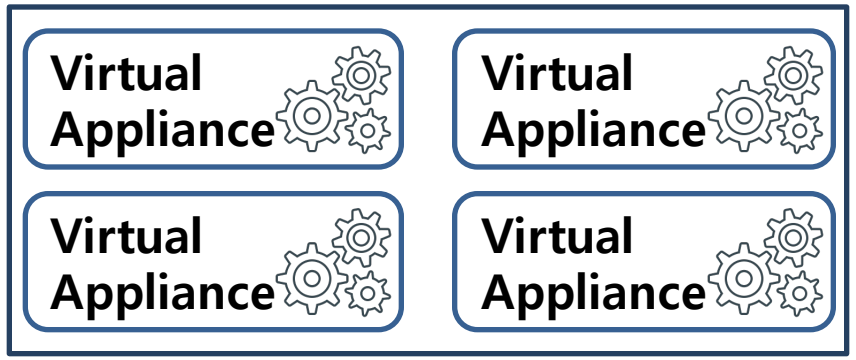
To disaggregate and virtualize the devices, turn each purpose-built hardware device into its software counterpart running on commodity hardware.

To provide a framework into which the resulting disaggregated elements can be plugged

Unifying abstraction that forge this collection of hardware and software elements into a scalable and agile system

Changing Infrastructure

Network Function Software



High volume standard processor

High volume standard storage & switching

Type of CORD

❖ M-CORD(Mobile CORD)

- ➢ A New Future in Networking with Mobile Edge Mashing up SDN & NFV

❖ E-CORD(Enterprise CORD)

- ➢ Enterprise WAN connectivity and innovative carrier grade services

❖ R-CORD(Residential CORD)

- ➢ Add applications and equipment that supports: Mobility, Metro Ethernet, Transport

Domain Services

❖ R-CORD

- ➢ vOLT, vSG, vRouter…

❖ E-CORD

- ➢ vCE, vOAM, vFirewall…

❖ M-CORD

- ➢ vBBU, vSGW, vPGW…

M-CORD

Disaggregated /Virtualized RAN

BBU, RRU front haul fabric

- Programmable Remote Radio Heads
- vBBU on commodity servers

Disaggregated /Virtualized EPC

MME, SGW, PGW

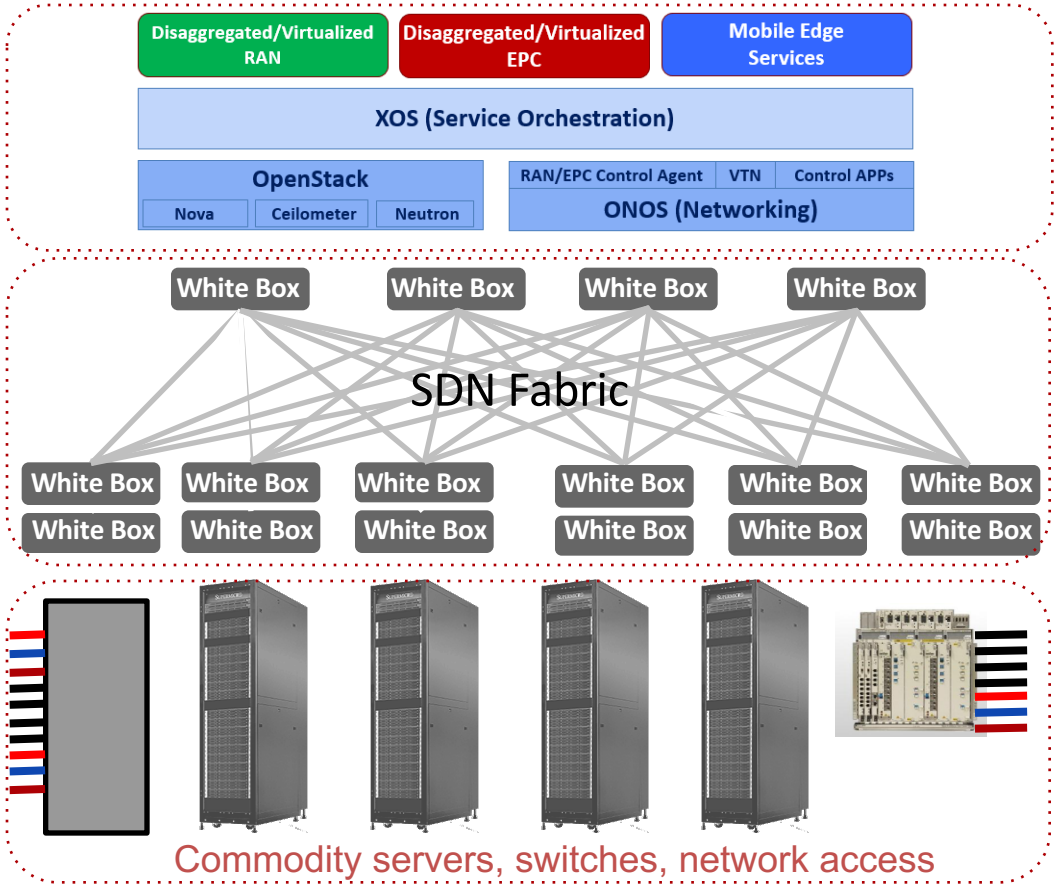
- Data plane management by ONOS
- VNF as a Service

Mobile Edge Services

Caching, SON, Billing

- Caching and other services from the edge
- Customized for enterprises and apps

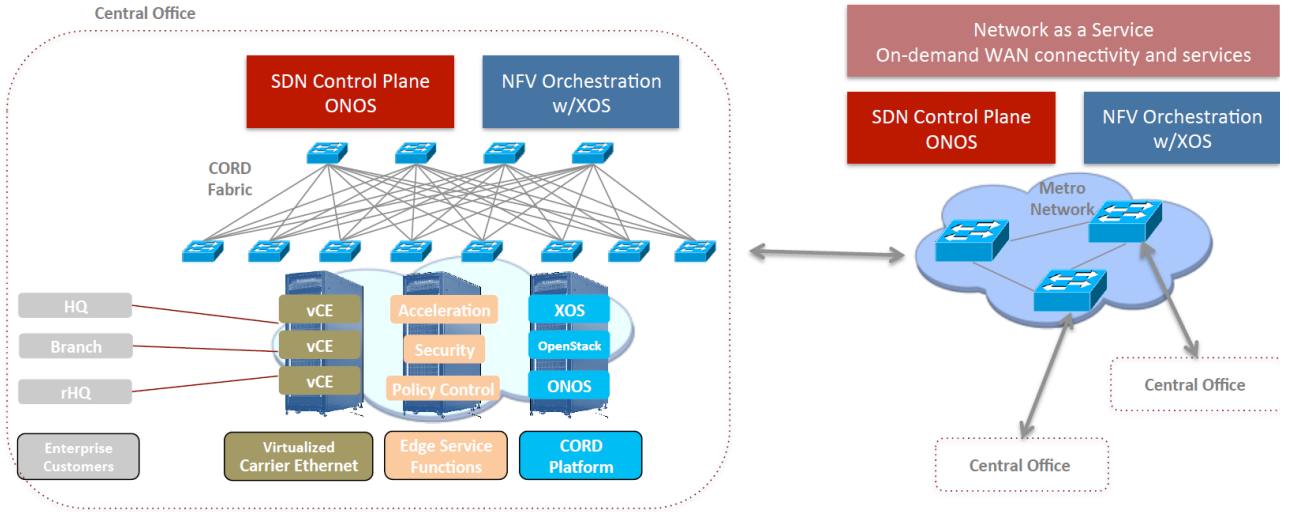
+



- Cloud-Agile Service Customization
- Dynamic radio resource optimization
- Open Control Interfaces
- Network Slicing
- Programmable Data Plane
- Deep Observability

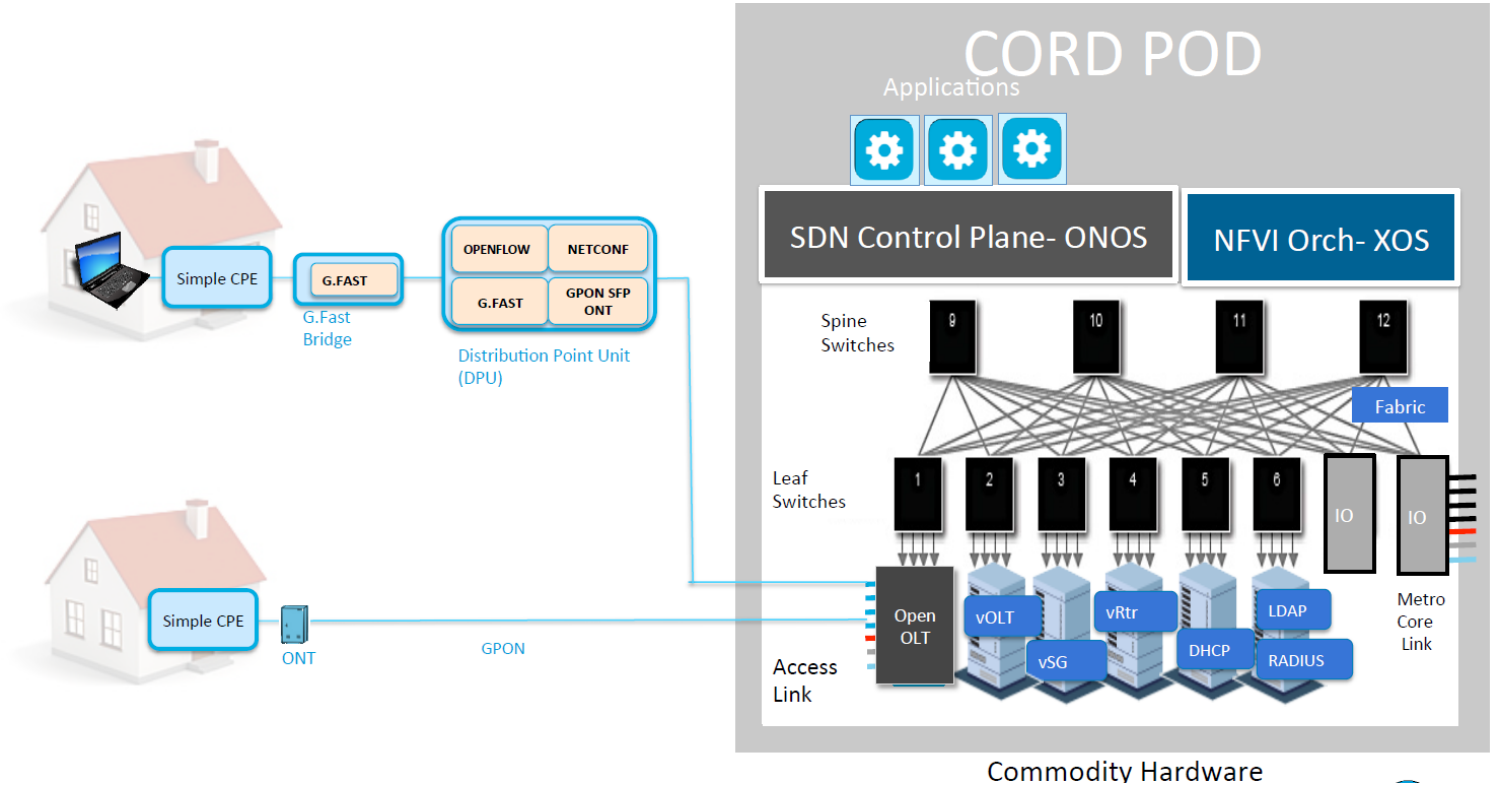
E-CORD

- Enterprise connectivity services over metro and wide area networks
- Built on commodity HW and open source software
- SDN/NFV-based elasticity of commodity clouds to bring datacenter economics and cloud agility to the Telco Central Office.
- Customized “network on demand” service
  - For different apps or user groups
  - With bandwidth on demand
  - Secure & isolated from other networks
- Software defined to observe, control, and adapt
  - With own portal and programmatic interface



R-CORD

- Services that leverage wireline access technologies like GPON, G.Fast, and 10GPON.
- Disaggregated and virtualized OLT, Subscriber Gateway, and router

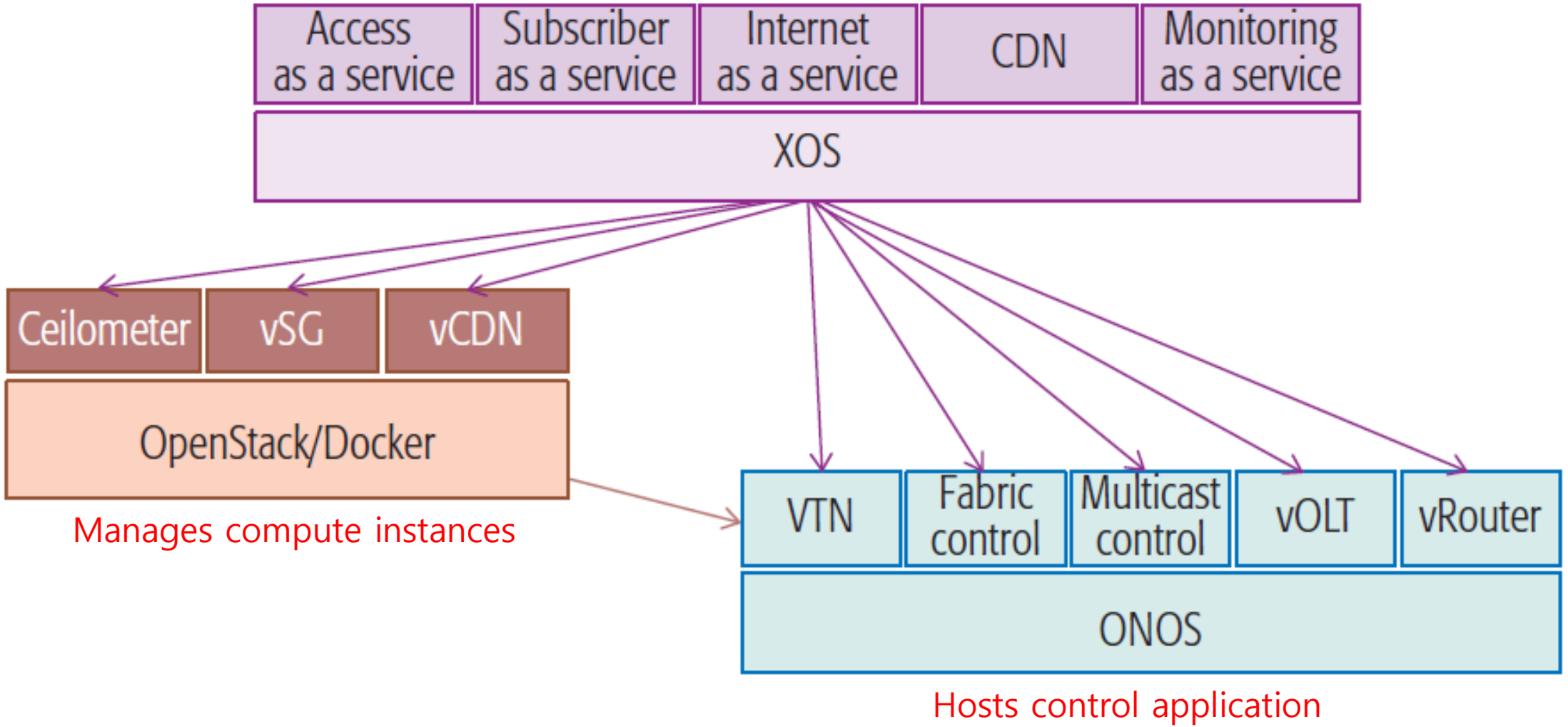


Inside CORD

Core Components

- OpenStack
- ONOS
- XOS
- OCP

Open Source Components in CORD



Software Building Block

❖ OpenStack

- ➢ Cluster management suite that provides the core Internet as a service(IaaS) and is responsible for creating and provisioning virtual machines and virtual networks

❖ ONOS

- ➢ It hosts a collection of control applications and manages both software switches and the physical switching fabric.

❖ XOS

- ➢ Framework for assembling and composing services. It unifies infrastructure services(provided by OpenStack), control plane services(provided by ONOS), and any data plane or cloud services(running in VMs or containers)

❖ Docker

- ➢ It is used to deploy and interconnect services. It also plays a role in deploying CORD itself.(e.g., the other management elements are instantiated in Docker container)

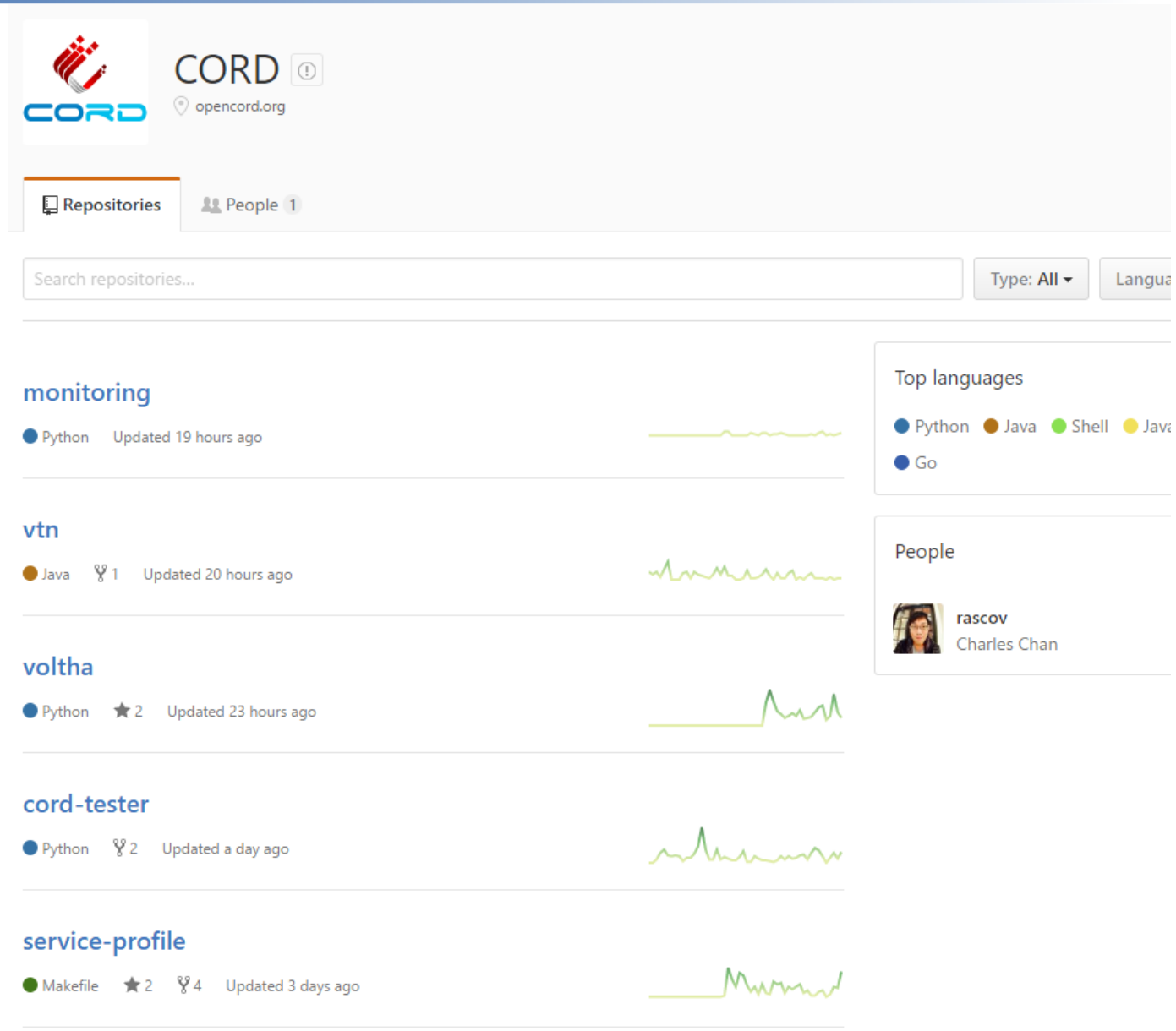
Two roles of ONOS in CORD

❖ **Interconnects VMs**

implementing virtual networks and managing flows across the switching fabric

❖ **Provides a platform for hosting control programs that implement CORD services.**

CORD project on Github



a list of OpenCORD Github

- **platform-install / service-profile**

Configurations, Profiles, Deployment, Installation

- **XOS**
- **MaaS**
- **AAA / Mcast / IGMP / OLT / VTN / CordConfig**
- **OpenStack / vRouter / vSG / vMME / vBBU / onos-service [only xos interface]**

OLT : Optical Line Terminal
IGMP(Internet Group Message Protocol)
OLT access management

a list of OpenCORD Github

- **platform-install / service-profile**
- **XOS**

XOS : service controller

- **MaaS**
- **AAA / Mcast / IGMP / OLT / VTN / CordConfig**
- **OpenStack / vRouter / vSG / vMME / vBBU / onos-service [only xos interface]**

OLT : Optical Line Terminal
IGMP(Internet Group Message Protocol)
OLT access management

a list of OpenCORD Github

- **platform-install / service-profile / CORD**
- **XOS**
- **MaaS**

Metal-as-a-Service is designed to help facilitate and automate the deployment and dynamic provisioning like PXE boot. It's canonical service

- **AAA / Mcast / IGMP / OLT / VTN / CordConfig**
- **OpenStack / vRouter / vSG / vMME / vBBU / onos-service [only xos interface]**

OLT : Optical Line Terminal
IGMP(Internet Group Message Protocol)
OLT access management

a list of OpenCORD Github

- **platform-install / service-profile / CORD**
- **XOS**
- **MaaS**
- **AAA / Mcast / IGMP / OLT / VTN / CordConfig**
- **OpenStack / vRouter / vSG / vMME / vBBU / onos-service [only xos interface]**

ONOS application for CORD

OLT : Optical Line Terminal
IGMP(Internet Group Message Protocol)
OLT access management

platform-install

platform-install

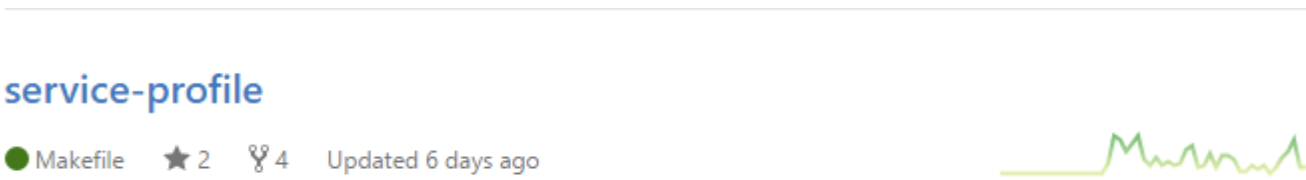
Shell ★ 3 9 Updated 4 hours ago

❖ **Contains Ansible playbooks for installing and configuring software components on a CORD POD : OpenStack, ONOS, and XOS.**

❖ **Ansible**

- Deployment and configuration automation tool
- Not Agent-based, it is based on SSH. A code is delivered by SSH and executed as a script.
- managed by ansible playbook
- This is a radically simple configuration management and deployment tool. It supports a wide variety of distributions, requires no software installed on managed machines, and users can get going in minutes. Extension modules can be written in any language.

service-profile

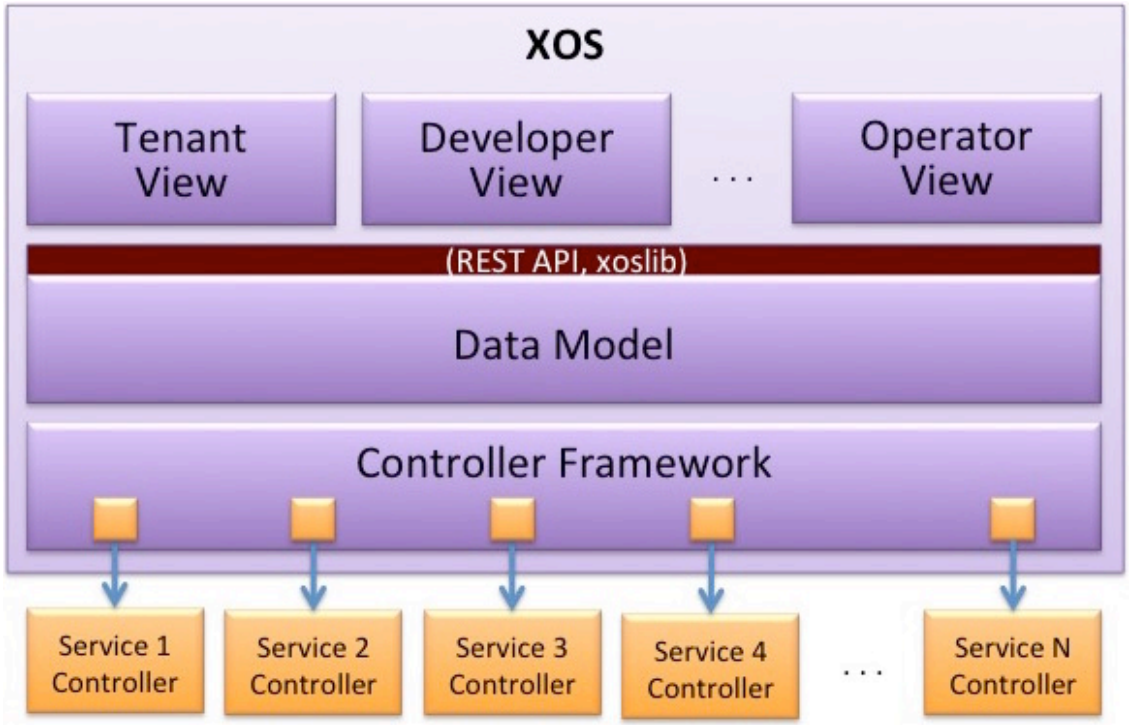


- Contains service profiles which configures XOS with a graph of services to be instantiated
- These configurations automate the creation of containers, loading things into the onboarding synchronizer, and starting XOS.

- configuration stuff like script & yaml file
- What is YAML
  - human-readable data serialization language.
  - lightweight markup language
  - nested list, hash, etc
  - Understandable, Readable

XOS

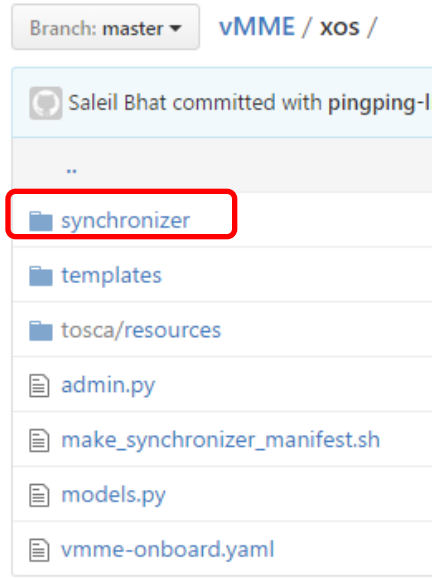
- Everything-as-a-Service
- Controller for CORD
- Make CORD both extensible and controllable.
- It is not an independent open source project,
  it is managed under CORD's project governance.



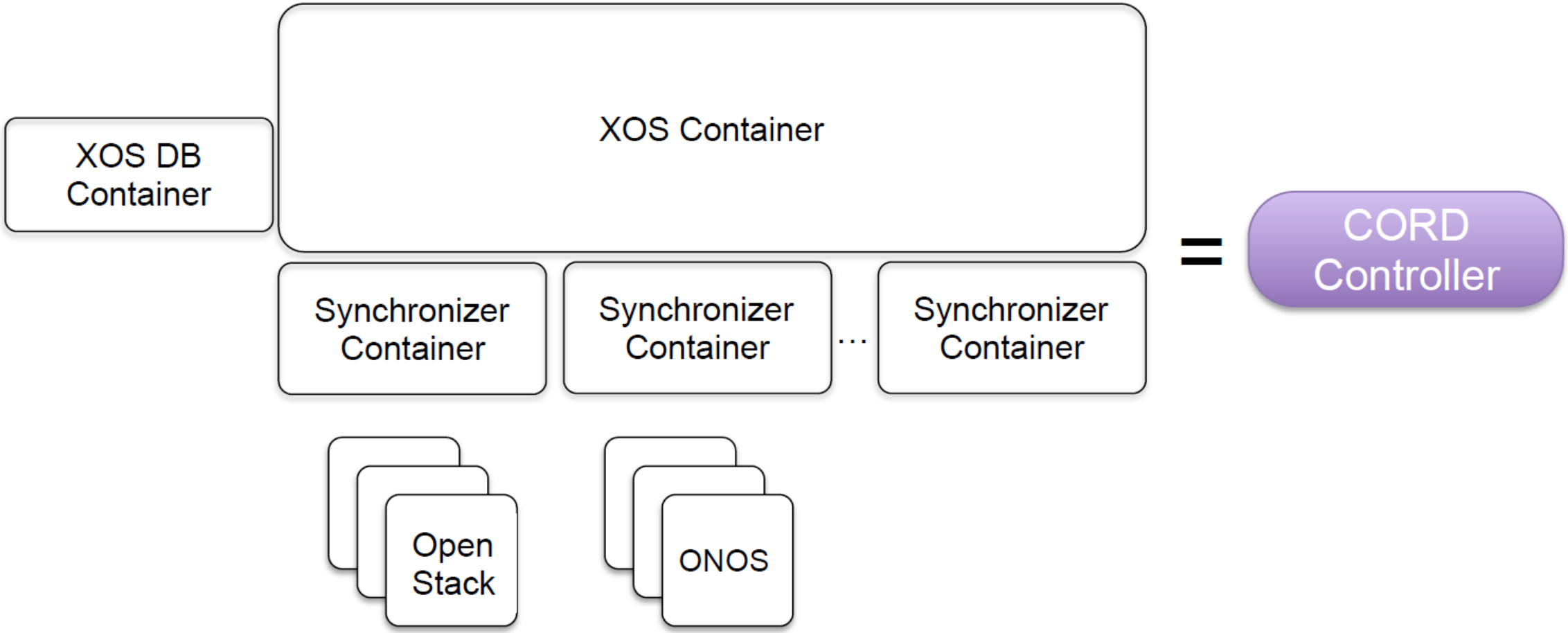
- 3 Core services (build using separate dockerfiles)
  - Database backend (postgres)
  - Webserver front end (Django)
  - Synchronizer daemon (interacting with other things)

- Note that earlier versions of XOS referred to the “Synchronizer” as the “Observer”.

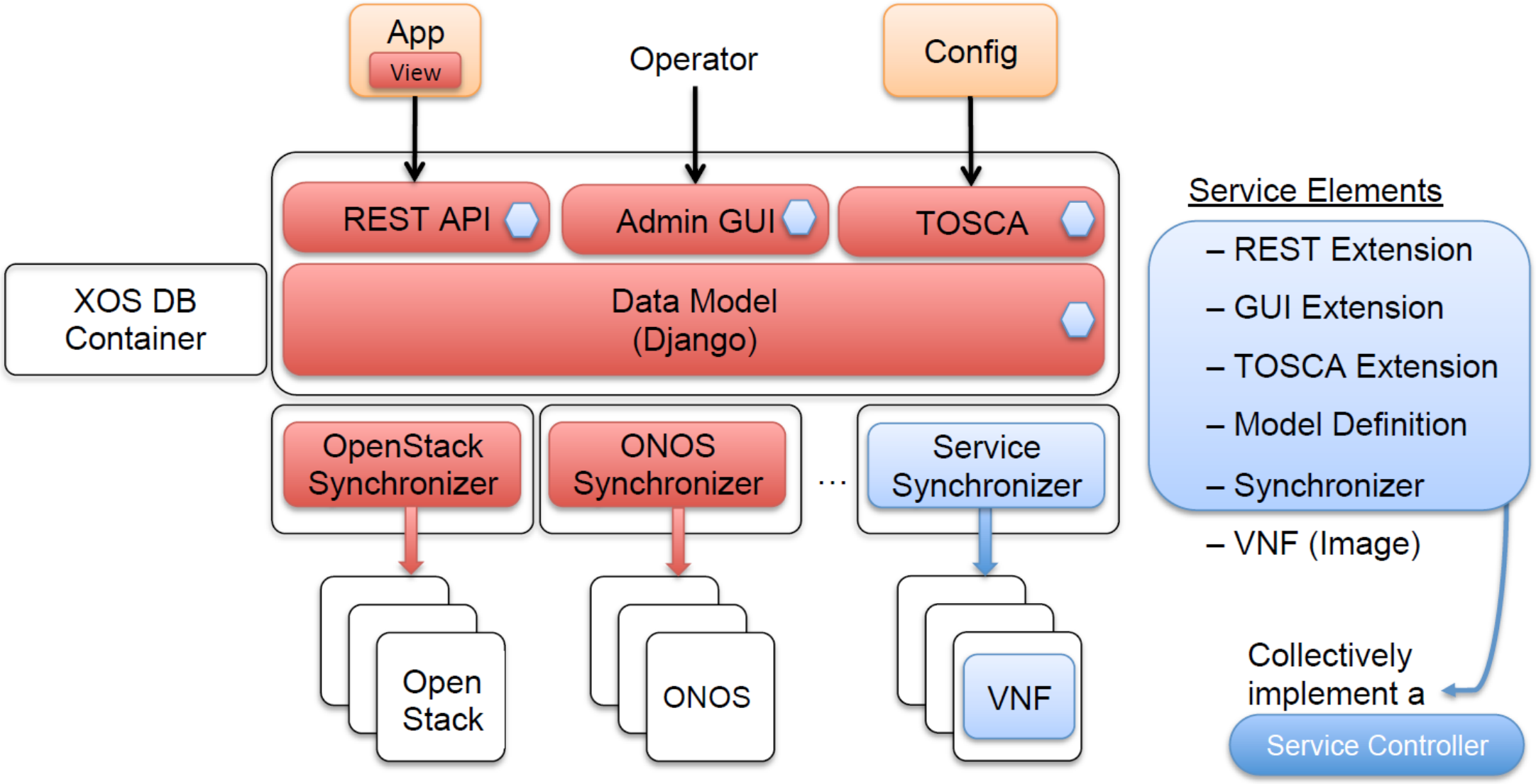
- Synchronizer is needed to set a state of data model



XOS : CORD controller

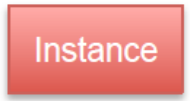


XOS Internals : assembling a service

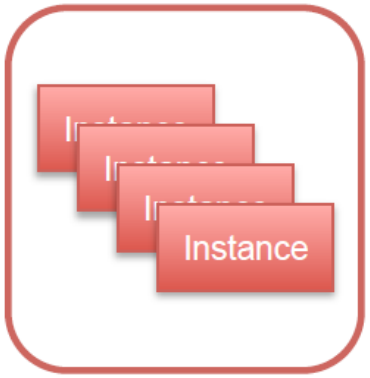


Topology and Orchestration Specification for Cloud Applications (TOSCA)

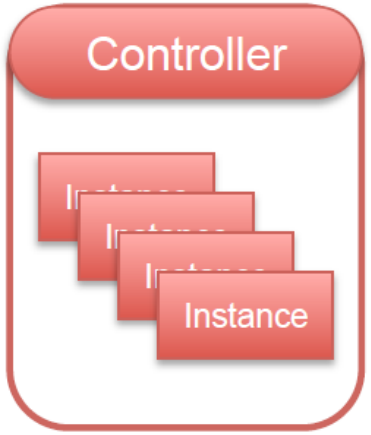
What is a service



Instance
(VM | Container | Container-in-VM)



Slice
(Instances[] + Networks[])

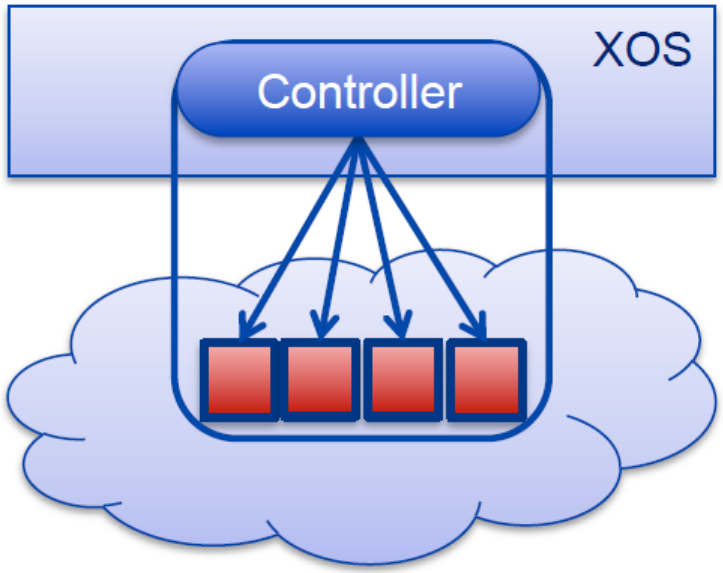


Service
("Controller" + Slices[])

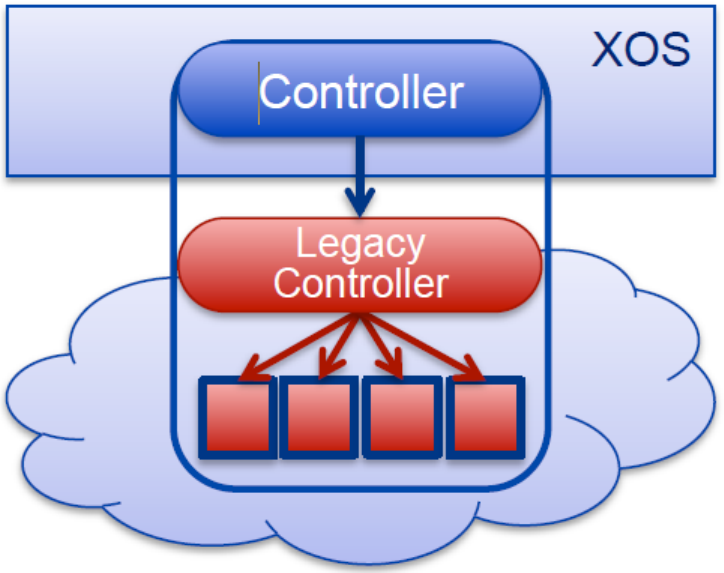
From native to legacy

Blue – XOS Defined/Managed

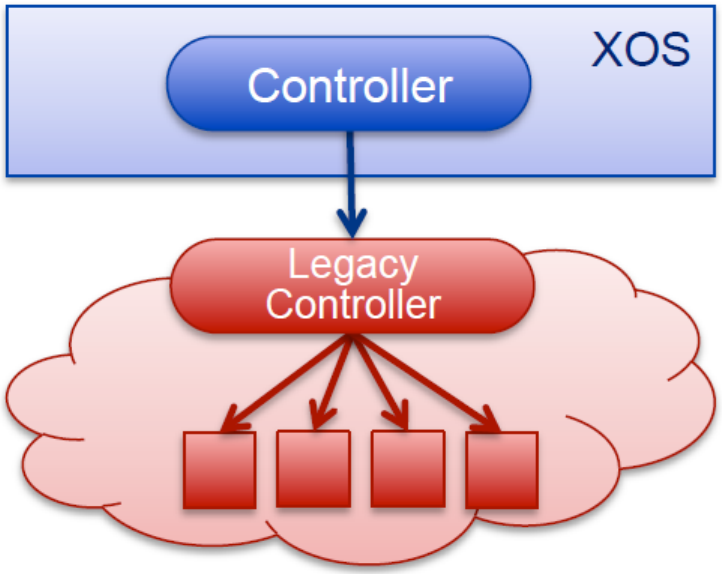
Red – External to XOS



e.g., vSG
(XOS includes tools to help construct a service)



e.g., vCDN
(XOS provides a means to coordinate VM acquisition & service init)



e.g., S3
(XOS provides a means to compose with an external service)

Thank you

Sangyun Han
Mobile Convergence Lab, Dept. Computer Engineering
Kyung Hee University

한상윤 석박 통합 과정
경희대학교 컴퓨터공학과 모바일 컨버전스 연구실
Email : sangyun0628@khu.ac.kr