

## K-ONE 기술 문서 #16

# 멀티사이트 K-ONE Playground 활용을 위한 K-DevOps Tower의 구축 및 활용방법

Document No. K-ONE #16

Version 1.1

Date 2017-05-24

Author(s) 신준식

■ 문서의 연혁

버전	날짜	작성자	내용
0.1	2017.04.10	신준식	전체 문서 틀 정리
1.0	2017.04.27	신준식	초안 작성
1.1	2017.05.23	신준식	3.1 절 수정 및 Reference 정비

본 문서는 2016년도 정부(미래창조과학부)의 재원으로 정보통신  
기술진흥센터의 지원을 받아 수행된 연구임 (No. B0190-16-2012, 글로벌  
SDN/NFV 공개소프트웨어 핵심 모듈/기능 개발)

This work was supported by Institute for Information &  
communications Technology Promotion(IITP) grant funded by the  
Korea government(MSIP) (No. B0190-16-2012, Global SDN/NFV  
OpenSource Software Core Module/Function Development)

## 기술문서 요약

K-ONE 컨소시엄의 SDN/NFV/Cloud 오픈소스 프로젝트 공동 개발/실증 환경을 제공하기 위하여, Edge-Cloud 모델에 대응하는 클러스터형 테스트베드인 K-Cluster의 설계 및 K-Cluster 하드웨어 시제품 구축 과정을 설명한다. 그리고 단일 K-Cluster로는 실증이 어려운 멀티사이트/도메인 관련 실증을 통합적으로 제공하기 위하여 K-Cluster를 멀티사이트에 분산 배포하고 네트워크로 연동하여 구축하는 멀티사이트 환경의 K-ONE Playground를 설계한다. 그리고 K-ONE 컨소시엄 참여기관인 광주과학기술원, 고려대학교, 숭실대학교, 포항공과대학교, 한국과학기술원, 총 5개 사이트를 대상으로 K-Cluster 하드웨어 시제품을 배포하고 네트워크로 연동하여 K-ONE Playground를 구축하는 과정에 대해 상세히 설명한다. 그리고 K-ONE Playground를 활용한 다양한 SDN/NFV/Cloud 통합 실증방안 및 향후 활용계획에 대해 상세히 기술한다.

## Contents

### K-ONE #16. 멀티사이트 클라우드 실증환경에 대응하는 K-Cluster 중심의 K-ONE Playground 설계 및 활용방안

1. 서론 .....	6
1.1. 목적 .....	6
1.2. Edge-Cloud Computing .....	6
1.3. 소프트웨어 정의 인프라 .....	9
2. K-Cluster .....	11
2.1. K-Cluster 필요성 .....	11
2.2. K-Cluster 정의 및 요구사항 .....	11
2.3. K-Cluster 설계: 구성요소 .....	12
2.4. K-Cluster 설계: 네트워크 .....	13
3. K-Cluster 기반의 K-ONE Playground .....	16
3.1. K-ONE Playground 필요성 .....	16
3.2. 멀티사이트 실증을 위한 K-ONE Playground 구성안 .....	16
4. K-Cluster 및 K-ONE Playground 구축 현황 .....	19
4.1. K-Cluster 하드웨어 시제품 구성 .....	19
4.2. K-ONE Playground 구축 .....	20
5. K-ONE Playground 활용한 SDI 기술 실증 .....	22
5.1. K-Cluster 활용 실증: 인텔리전스 서비스 .....	22
5.2. K-Cluster 활용 실증: 박스 내/박스 간 Fast Inter-connections .....	23
5.3. K-ONE Playground 활용 실증: SDN/NFV/Cloud 통합 실증방안 .....	24
5.4. K-ONE Playground 활용 실증: SDN/NFV/Cloud 오픈소스 개발 및 검증 .....	26
6. 결론 .....	27

## 그림 목차

<그림 1: 소프트웨어 정의 인프라 구도> .....	9
<그림 2: K-Cluster 구성요소 설계> .....	12
<그림 3: 소프트웨어 정의 인프라 패러다임의 MultiX Challenges> .....	16
<그림 4: K-ONE Playground 구성 설계도> .....	18
<그림 5: K-Cluster 하드웨어 시제품 상세사양 및 구축모습> .....	19
<그림 6: K-ONE Playground의 구축 현황도> .....	20
<그림 7: 박스 내부와 다수의 박스 간 빠른 연결 이슈를 포괄하는 K-Cluster> ...	23
<그림 8: K-ONE Playground를 활용한 SDN/NFV/Cloud 통합실증방안> .....	25
<그림 9: K-ONE Playground를 활용한 오픈소스 소프트웨어 개발계획> .....	26

## 표 목차

## K-ONE #16. 멀티사이트 클라우드 실증환경에 대응하는 K-Cluster 중심의 K-ONE Playground 설계 및 활용방안

## 1. 서론

### 1.1. 목적

- o 본 기술문서는 K-ONE 컨소시엄 참여기관인 광주과학기술원, 고려대학교, 숭실대학교, 포항공과대학교, 한국과학기술원, 총 5개 기관에서 각자 목표로 상정한 다양한 오픈소스 소프트웨어의 개발 및 검증환경을 공통된 하드웨어 묶음 형태로 제공하기 위하여 Edge-Cloud 패러다임에 대응하는 클러스터형 테스트베드인 K-Cluster의 설계 및 구성 방안을 제시한다. 또한 오늘날 대규모의 실제 인프라 환경에서 발생하는 다양한 이슈를 반영하는 SDN/NFV/Cloud 기술의 연구개발을 위해 분산 배포된 다수의 K-Cluster를 네트워크로 연동하여 구축한 K-ONE Playground(또는 공용개발환경)에 대해 설명하고, 이를 활용한 다양한 SDN/NFV/Cloud 통합 실증방안에 대해 기술한다.
- o K-Cluster는 하드웨어 구성, 스펙, 모델이 고정적으로 정의된 물리적인 구성을 지칭하는 것이 아니며, Edge-Cloud 모델에 대응하도록 설계된 논리적인 요소로 구성된 논리적인 테스트베드 모델을 의미한다. 따라서 K-Cluster의 실제 물리적인 구성은 구축하는 사람에 따라 상이하며, 제공하는 자원/서비스 규모 또한 실제 물리 하드웨어 스펙에 의존한다. 본 기술문서에서는 K-Cluster의 각 논리 구성요소에 대응하는 물리 박스들을 정의한 K-Cluster 시제품을 제시한다.
- o K-ONE Playground는 단일 K-Cluster로는 실증이 어려운 멀티사이트/도메인 관련 실증을 통합적으로 제공하기 위하여 K-Cluster 시제품들을 K-ONE 컨소시엄의 참여 사이트에 각각 배포하고 이를 KOREN/KREONET 연구망으로 연결함으로써 구축한 멀티사이트 실증 테스트베드를 지칭한다. 본 기술문서에서는 K-ONE Playground의 설계 및 구축 과정에 대해 상세히 기술하고, Playground를 활용한 SDN/NFV/Cloud 실증 방안 및 향후 활용계획에 대해 설명한다.

### 1.2. Edge-Cloud Computing

- o ICT 인프라 분야에서 클라우드 컴퓨팅 (Cloud Computing)이 사실상 핵심 기술이 됨과 동시에 사물인터넷 (IoT, Internet of Things) 관련 기술들이 잇달아 등장함에 따라 IoT/Cloud의 두 계층으로 구성된 ICT 환경이 일반화 되었다. 물리적 관점에서 IoT는 데이터를 수집하는 센서 Things와 외부와 상호작용하는 액추에이터 Things로 구분되며, 지리적으로 광범위한 범위에 분산되어 위치한다[1]. 반면에 Cloud는 데이터센터로 대표되는 중앙 집중형의 대규모 박스(i.e., 서버, 스위치, 라우터 등)들을 Cloud의 Compute/Networking/Storage 자원으로 Pooling 하며, 요청에 따라 필요한 만큼의 자원을 제공한다[2].



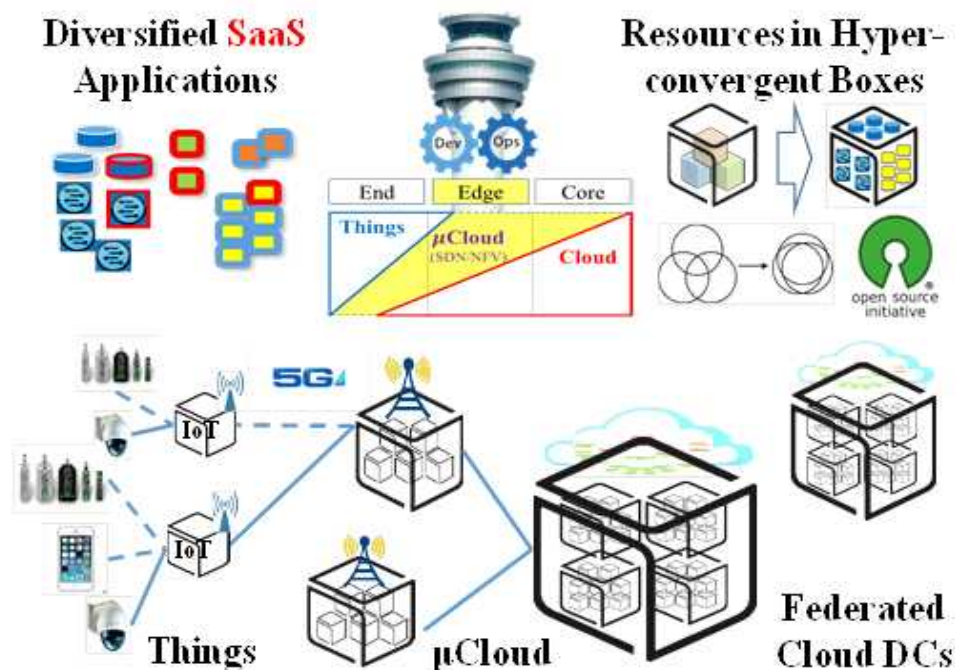
- o IoT Things 들로부터 생성된 센서 데이터들이 Cloud 데이터센터로 전송되며 Cloud의 자원을 활용해 데이터를 연산한 결과 값을 통해 센서/액추에이터의 동작을 제어한다. 나아가 IoT에서 수집된 빅데이터를 Cloud의 대규모 자원을 기반으로 Analytics Engine을 통해 연산함으로써 데이터를 인텔리전스(Intelligence)에 연결하려는 다양한 연구개발이 진행되고 있다.
- o 사물인터넷 기술의 발전과 대중화에 따라 기존의 사물인터넷 도메인에서 나아가 의료, 헬스, 공장 등의 특정 산업 도메인에 연결되는 산업 사물인터넷 (IIoT, Industrial IoT) 형태로 확장되고 있다. 사물인터넷의 확장으로 인해 IoT Things로부터 생성되는 전체 데이터양이 증가함은 물론이고, 각 사물인터넷 서비스별 사용자 체감 수준 (Quality of Experience)을 충족하기 위한 요구사항 또한 다양해지고 있다. 특히 산업 도메인에 적용된 의료, 헬스, 공장 등의 산업 사물인터넷 서비스의 경우 데이터 처리 시간에 따라 금전적/신체적인 손해를 입힐 수 있으므로 짧고 안정적인 처리 지연 시간을 요구한다[3].
- o 하지만 기존의 IoT/Cloud의 두 계층으로 구성된 ICT 환경에서 분산된 IoT Things로부터 발생한 모든 데이터는 연산을 위하여 반드시 중앙의 Cloud까지 전송이 되어야 한다. 이로 인해 대량의 데이터가 IoT/Cloud 사이의 네트워크 자원을 점유하고, 데이터가 반드시 WAN(Wide Area Network) 구간을 2번이나 거쳐 감에 따라 상당한 수준의 전송 지연이 발생한다. 이와 같은 문제를 해소하고자 IoT/Cloud 사이 계층에 위치하는 랙 규모(Rack-scale)의 소규모 Cloud 클러스터인 Edge-Cloud가 도입되어 IoT/Edge-Cloud/Core-Cloud의 형태로 ICT 인프라가 변화하는 추세이다. Edge-Cloud는 IoT Things 또는 ICT 서비스에 가능한 근접한 위치에 Compute/Networking/Storage 자원을 배치하는 것을 기본 개념으로 하며, IoT 환경에서 발생하는 데이터를 일차적으로 처리하는 전처리 연산(Front-end Computing)을 담당하는 동시에 IoT 환경과 Core Cloud를 데이터/네트워킹 측면에서 상호 연결하는 중계자 역할을 동시에 수행한다.
- o Edge-Cloud는 기존 IoT/Cloud 환경에서의 데이터 연산을 위한 자원 활용 측면의 유연성은 그대로 유지하면서 높은 수준의 데이터 응답 저지연을 보장할 수 있다[4]. 상기 예시로 든 IIoT 서비스와 같이 빠른 응답시간을 요구하는 서비스의 경우 지연에 민감한 소규모 데이터 연산은 Edge-Cloud 상에서 바로 연산 처리되어 그 결과를 반환하므로 빠른 응답이 가능하다. 하지만 Edge-Cloud의 소규모 자원으로 감당이 어려운 BigData 계열의 대규모 데이터 연산의 경우 Edge-Cloud가 중앙의 Core Cloud에게 데이터를 전달하고, 연산 결과를 다시 IoT 환경에 전달하는 중계자 역할을 수행한다.

- o IoT와 Core-Cloud의 자체 네트워크가 안정적이고 유연하게 구성되어 있다고 하더라도 Edge-Cloud가 IoT와 Core-Cloud 간을 상호 연결하지 못한다면, ICT 인프라 전체가 정상적으로 동작하지 못한다. 따라서 Edge-Cloud는 인프라 전체의 원활한 데이터 전송을 위한 네트워킹을 제공하는 허리의 역할도 담당하며, Edge-Cloud는 안정적이고 유연하며 고성능의 네트워킹을 제공해야 한다. 이러한 인프라 구조에 대응하기 위하여 오늘날 많은 네트워킹 기술들이 개발되고 있으며, 그 중 특히 소프트웨어 정의 인프라 패러다임의 핵심 기술인 Software-defined Networking (SDN), Network Function Virtualization (NFV) 기술이 Edge-Cloud 환경을 대상으로 개발되고 있다.
- o 세부적으로는 Edge-Cloud는 IoT Things와 직접적으로 연결되어 있어, Things들이 갖고 있는 3G/4G/Wifi/LoRaWAN 등의 다양한 무선 연결을 Edge-Cloud에서 제공하기 위한 Multi-Access 관련 연구개발이 진행되고 있다. 한편 포함하는 Edge-Cloud/Core-Cloud 간 또는 Edge-Cloud들 간의 WAN 연결성을 위해 Software-defined Networking을 WAN으로 확장하는 SD-WAN 기술과 멀티사이트 네트워킹 기술 또한 Edge-Cloud를 중심으로 이루어지고 있다.
- o 세계적으로도 융합형 자원박스 중심의 소규모 클러스터를 기반으로 하는 에지 클라우드를 구축 및 운영하면서, DevOps 방법론으로 상기한 SDN/NFV를 적용하려는 다각적인 접근이 일어나고 있다. 우선 유럽의 표준 단체인 ETSI에서는 IoT 환경과 코어 클라우드 간을 중계하는 에지 클라우드 기술의 중요성을 인식하고, 이를 표준단체의 관점에서 정리하려는 노력을 수행하고 있다. 또한 학계에서도 에지 클라우드 컴퓨팅에 대응하기 위한 Cloudlet[4]과 OpenBox[5]와 같은 연구 개발을 활발히 진행하고 있다. 그리고 실제 SDN/NFV 인프라 서비스를 제공해야 하는 통신사업자들도 클러스터링 된 융합형 박스 중심의 에지 클라우드에 주목하고 있으며, 이러한 동향을 바탕으로 ON.Lab 중심의 CORD[6] 오픈소스 프로젝트가 많은 통신 사업자들의 주목을 받고 있는 상황이다.
- o 이 중에서도 융합형 자원을 중심으로 SDN/NFV를 소프트웨어-정의 관점에서 담으려는 OpenBox 프레임워크 및 CORD 오픈소스 프로젝트에 주목할 필요가 있다. OpenBox 프레임워크는 기존 미들박스의 제어평면(Control Plane) 및 데이터 평면(Data Plane)을 분리하고, 제어평면을 중앙 집중화하면서 데이터 평면 하드웨어로 기존 미들박스 벤더의 특수 하드웨어가 아닌, 상용 화이트박스를 그대로 활용하려는 방향성을 갖고 있다. 또한 소프트웨어-정의 접근법의 특징점을 미들박스 환경에 반영하기 위하여, 미들박스 단위가 아닌 미들박스의 세부 모듈단위로 배포하고, 이를 체이닝하는 접근법을 취함으로써 미들박스의 혁신성, 유연성을 향상시켰다. 미들박스 환경을 소프트웨어-정의 관점으로 변화시키려는 OpenBox와 달

리, CORD (Central Office Re-architected as a Data Center) 프로젝트는 그 이름에서 볼 수 있듯 기존 통신 사업자의 전화국사를 SDN/NFV/Cloud 통합 기반의 SDI 형태로 변화시키기 위한 보다 큰 규모의 오픈소스 하드웨어/소프트웨어 프로젝트에 해당된다. CORD에서는 기존 전화국사를 SDI 형태로 변화시키기 위하여 기존의 특수 장비들이 아니라 상용의 융합형 자원박스로 구성된 랙(Rack) 규모의 클러스터를 활용한다. 그리고 클러스터 상에 OpenStack 클라우드 운영체제를 적용하여 VNF (Virtual Network Function)가 동작할 기본 환경을 구성하고, ONOS SDN 제어기를 통해 중앙에서 전체 트래픽 플로우를 스티어링(Steering)한다. 그리고 XOS라는 오케스트레이션 운영체제가 OpenStack 및 ONOS를 오케스트레이션하여 중앙에서 유연하게 관제하도록 한다. CORD는 주요 활용 케이스로 M-CORD (Mobile-CORD), R-CORD (Residential-CORD), A-CORD (Analytics-CORD), E-CORD (Enterprise-CORD)의 네 가지 경우를 정의하여 개발을 진행하고 있다.

- o 따라서 오늘날 IoT, 데이터 연산, 네트워킹 등 실제 인프라에 내제된 난제들을 반영하는 실증적인 ICT 관련기술의 연구개발을 위해서는 Edge-Cloud를 중심으로 풀어나가는 것이 중요하다.

### 1.3. 소프트웨어 정의 인프라



<그림 1: 소프트웨어 정의 인프라 구조>

- o SDN/NFV/Cloud가 통합된 융합형 SDI 개념에 따라 구성된 인프라는 그림 1과

같이 코어(Core), 에지(Edge), 그리고 종단(End) 영역들로 구분된다. 코어 영역에는 클라우드 데이터센터들이 연합(Federation) 되어 중심에 자리잡고 있으며, 종단 영역에는 메쉬 형태로 서로 엮어진 다양한 사물들이 넓게 분포하게 된다. 이를 세부적으로 살펴보면, SDN이 제공하는 유연한 네트워킹의 적용 범위를 액세스-WAN-Cloud 에 걸치는 종단간(end-to-end)으로 확장하는 모습도 보인다. 즉 유/무선으로 연결된 스마트한 사물에서 시작한 가상화 된 네트워킹이 SDN의 플로우 기반 유연성을 활용하면서 서비스 제공을 위한 핵심 자원이 위치한 클라우드로 연결되는 추세도 관찰된다. 또한 융합형 박스 형태의 자원집합의 입장에서는 코어 영역에서 멀어짐에 따라 계산/저장/네트워킹 자원의 배합이 달라지면서 박스의 규모가 점차 축소되는 소위 S-Box, M-Box, 그리고  $\mu$ -Box 의 구성이 나타나게 된다. 즉 필요로 하는 자원의 유연성/경제성에 대한 요구가 증대됨에 따라 목적에 맞춰 선택적으로 대응하는 각종 융합형 박스들로 만들어내는 자원집합의 부상이 예상되고 있다.

- o 자원(resource)과 서비스(service)의 효율적인 연계 구도에 따라 사용자에게 할당된 융합형 자원집합(resource pool)을 편리하게 연동하여 필요한 서비스들을 적절하게 창출하면서 활용할 것으로 전망된다. 즉 세부적인 차원에서는 다양한 자원집합으로의 확대도 나타나겠지만 전반적인 흐름에서는 프로그래밍과 가상화가 가능한 융합형 자원들이 유연성이 있게 하나로 묶인 확장성이 있는 미래형 자원집합에 주목해야 한다.
- o 미래지향적인 ICT 인프라로의 혁신을 위해서 융합된(hyper-converged) 박스형 자원집합을 활용하면서 SDN, NFV, 그리고 Cloud 등을 하나로 통합하여 사용자들이 원하는 다양한 서비스를 담아낼 수 있어야 한다. 따라서 원하는 서비스에 따라 자유롭게 맞춰지는 조합형(Composable) 특성을 지니는 융합형 소프트웨어-정의 인프라(Software-Defined Infrastructure: SDI)로의 진화가 예상된다. 이를 위하여 SDN/NFV/Cloud가 통합된 융합형 SDI 환경이 제공하는 자원집합을 활용하여 다양한 SaaS 기반 서비스들을 증하는 모습이 그려진다. 살펴보면 종단에 위치한 스마트한 사물들을 유/무선 메쉬(mesh)로 연결하는 가상화 된 수준의 플로우 기반의 SDN제어기를 활용한 유연한 네트워킹과 에지 Cloud 개념에 따른 계산/저장 기능을 사물들과 이에 근접한 사용자들을 효과적으로 지원하고 있다. 또한 에지 Cloud 의 자원집합으로는 충분하지 못한 대규모의 사용자들을 지원할 수 있도록, 대규모 자원집합들을 필요로 하는 서비스들을 감당하는 메인 클라우드로 연결하는 것도 중요하다.

## 2. K-Cluster

### 2.1. K-Cluster 필요성

- o SDI 패러다임에서 자원집합의 유연성과 경제성에 대한 요구가 커짐에 따라 목적에 맞춰 선택적으로 대응하는 융합형 박스들로 만들어내는 유연하고 경제적인 자원집합의 실체화가 필요하다. 그러므로 융합된 박스형 자원집합의 입장에서는 위치한 영역에 맞춰서 계산/저장/네트워킹 자원의 배합을 바꾸고 이에 부합하도록 필요한 소프트웨어 기능들 (functions)을 다양하게 조합하도록 만들어 내는 숙제가 주어진다. 따라서 미들박스 (MiddleBox)에 대응하는 중간의 위치에서 SDN/NFV/Cloud 통합을 훈련해 볼 수 있는 자원집합이 있다면 매우 효과적일 것이다. 이러한 융합된 박스형 자원집합은 오픈소스 소프트웨어 기반으로 사용자들이 마음대로 변화시키는 화이트박스의 속성을 지녀야 한다. 또한 영역별로 분산된 융합형 박스 자원들 간의 연결(inter-connect)은 느슨한 결합에서 시작하여 시간이 흐르면서 DevOps (개발운영 병행체제) 도구에 의한 자동화를 통해서 점차적으로 강한 결합으로 변화해야 한다. 이를 정리하면 <그림 1>과 같은 융합된 화이트 박스들의 묶음으로 구성하는 에지에 위치한 분산된 소규모 클라우드인  $\mu$  Cloud를 실험적으로 구축하고 운영해 보는 K-Cluster 형태의 자원 구성이 요구된다.

### 2.2. K-Cluster 정의 및 요구사항

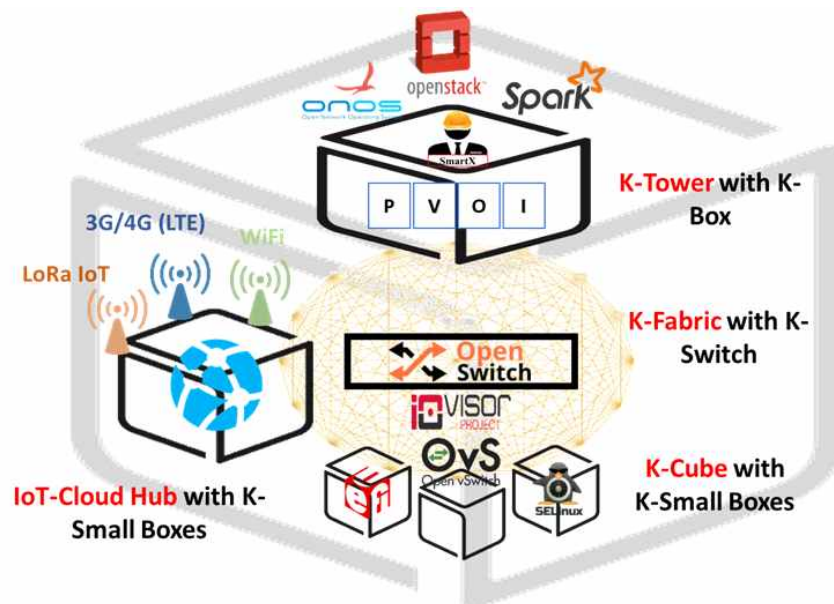
- o K-Cluster는 소프트웨어 정의 인프라 패러다임에 대응하는 다양한 SDN/NFV/Cloud 실증을 제공하기 위한, Edge-Cloud 모델에 대응하는 경제적인 소규모 클러스터 형태의 테스트베드로 정의한다.
- o K-Cluster는 하드웨어 구성, 스펙, 모델이 고정적으로 정의된 물리적인 구성을 지칭하는 것이 아니며, Edge-Cloud 모델에 대응하도록 설계된 논리적인 요소로 구성된 논리적인 테스트베드 모델을 의미한다. 따라서 K-Cluster의 실제 물리적인 구성은 구축하는 사람에 따라 상이하며, 제공하는 자원/서비스 규모 또한 실제 물리 하드웨어 스펙에 의존한다.
- o K-Cluster의 예상 활용처는 대규모의 ICT 서비스를 제공하는데 필요한 안정적이고 고성능의 인프라를 구성하는데 쓰이는 것이 아니라, Edge-Cloud와 연관된 다양한 SDI 실증을 및 연구개발을 위한 적정 규모의 테스트베드를 확보하기 어려운 연구자, 개발자를 위한 소규모 테스트베드 환경을 기본 활용처로 상정한다. 물론 K-Cluster도 안정적이고 고성능의 자원을 제공하는 것이 주요 요구사항 중 하나로 실제 Production 환경에 적용하는데 부족함이 없으나, 요구사항의 우선순위 측면에서 다양한 Edge-Cloud 기술을 하나의 테스트베드에 수용하기 위한 자원 측면의 유연성과 구축 경제성을 설계 시 가장 높은

우선순위로 한다.

- o 상기 소프트웨어 정의 인프라 패러다임에 대응하는 실증을 지원하기 위해 K-Cluster는 다음과 같은 요구사항을 만족해야 한다.
  - K-Cluster는 DevOps 자동화 도구를 활용하여 소프트웨어 기반으로 자동화된 관제 (Monitor & Control)가 되어야 함.
  - K-Cluster의 각 요소는 특정 벤더에 종속되지 않은 오픈 하드웨어 박스들로 구성할 수 있어야 함.
  - K-Cluster는 내부 구성요소 간 높은 수준의 안정성/유연성/대역폭을 보장하는 연결성을 제공하는 네트워킹을 제공해야 함.
  - 경제적으로 저렴하게 구축할 수 있어야 함.
  - 하나의 하드웨어 묶음으로 다양한 SDI 실증에 대응할 수 있어야 함.

### 2.3. K-Cluster 설계: 구성요소

- o K-Cluster는 오픈소스 하드웨어를 용이하게 적용할 수 있는 화이트박스 개념의 장비들로 <그림 2>와 같이 구성하여 경제성이 확보되어야 한다. 이때 K-Cluster가 K-Tower, K-Fabric, K-Cube, IoT-Cloud Hub등으로 나뉘지는 세부적인 구성을 효과적으로 감당하는 오픈소스 중심의 하드웨어와 소프트웨어를 적용하는 것도 필수적이다.



<그림 2: K-Cluster 구성요소 설계>

- o 먼저 K-Tower는 K-Cluster 전체를 DevOps(개발운영병행체제) 관점에서 관리/운영의 자동화를 위한 단일 또는 복수의 K-Box들로 구성된 관제(monitor & control)를 전담한다. K-Tower는 내부적으로 오픈소스 DevOps 도구를 활용하여

자동화를 지원하는 Provisioning 센터, Visibility 센터, Orchestration 센터, 그리고 Intelligence 센터를 위한 지원역할을 담당하는 소프트웨어 적인 기능들을 포함한다. 상기한 각종 센터들을 위한 기능들과 함께 K-Tower 상에는 K-Cluster의 대표 인터페이스 역할을 수행하는 SmartX 에이전트가 위치하여 K-Cluster와 중앙 DevOps 타워와의 상호작용을 위한 창구 역할을 수행한다. SmartX 에이전트는 K-Cluster 간 상호작용을 위한 창구 역할도 수행하여 다수의 K-Cluster들의 효과적인 연동도 지원한다. 그리고 K-Cluster의 하단부에 위치한 K-Cube는 다수의 동종 K-Small 박스 자원들로 구성된 K-Cluster 내부의 클러스터로써 계산/저장/네트워킹을 담당하는 공유된 자원집합을 지칭한다. 예를 들어 K-Small 융합형 박스를 클러스터링 한 K-Cube를 활용하면 앞서 설명한 SDN/NFV/Cloud 등의 다양한 에지 클라우드의 기능들을 수용할 수 있다. 그러면 K-Tower는 에지 클라우드에서 요구되는 다양한 기능들을 내부에 소재하여 소프트웨어 기반으로 자동화된 각종 센터 기능들을 통해 K-Cube가 감당하도록 설정하고 운영한다.

- o 그리고 K-Fabric은 K-Cluster의 중앙에 위치하여 물리/가상 네트워킹 기능들을 활용하여 패브릭 형태의 밀결합된 네트워킹을 구성한다. 이를 통해 K-Cluster 구성 요소들인 K-Tower, K-Cube, IoT-Data Hub 간의 빠르고(fast) 유연하며(flexible) 안정된(reliable) 연결성을 제공한다. 이를 위한 활용하는 K-Switch 스위치 자원은 K-Cluster에서 요구하는 연결성을 제공하기 위해 API를 통해 소프트웨어로 모든 부분이 제어 가능한 오픈소스 네트워킹 박스를 의미한다. K-Switch 자원은 Linux Foundation OpenSwitch, Big Switch ONL (Open Networking Linux) 등의 리눅스를 기반으로 개발되는 네트워킹 운영체제를 활용하면 효율적이다. 또한 K-Fabric를 K-Tower의 DevOps 도구를 통해 소프트웨어로 제어하는 것이 가능하며, 새로운 네트워킹 기술들을 손쉽게 적용하도록 지원함으로써 K-Cluster의 유연성을 향상시켜야 한다.
- o 한편 K-Cluster를 IoT 환경과 연계하기 위해서 IoT-Cloud Hub를 교두보로 활용한다. 구체적으로 IoT 환경의 다양한 센서 및 액츄에이터들과의 통신에 대응하기 위한 3G/4G, WiFi, LoRa IoT 등과 같은 무선 인터페이스를 통해 IoT를 K-Cluster에 물리적으로 연계할 수 있다. 이를 통해 IoT 환경에서 발생하는 다양한 소량의 데이터를 전달받아 K-Cluster로 전달하는 진입로의 역할을 수행한다. 이와 동시에 연결된 IoT 환경들을 Cloud 입장에서 관제하는 대장 역할을 수행하면서 매쉬 형태로 구성되는 IoT와 Cloud 간의 네트워킹을 지원해야 한다. 즉 IoT-Cloud Hub라는 중간자를 통해 IoT와 K-Cluster가 담당하는  $\mu$ Cloud 간의 이질적인 극복하면서 매끄럽게 연계하는 것이 중요하다.
- o 정리해보면 K-Tower, K-Fabric, K-Cube, IoT-Data Hub 등으로 구성된 K-Cluster

모델을 활용하면 에지 클라우드에서 요구하는 SDN/NFV/Cloud 통합과 FastData/BigData/HPC 기반의 다양한 서비스를 저렴한 상용 화이트박스들을 활용하여 유연하게 대응할 수 있다.

## 2.4. K-Cluster 설계: 네트워크

- o SDI 패러다임과 연계된 SDN, NFV, Cloud 기술과 근래 ICT의 핵심 기술로 떠오르고 있는 BigData, FastData, HPC와 같은 인텔리전스 서비스를 실증하는 공통 인프라로써 동작하기 위해서는 다양한 기술들을 유연하고, 신뢰성있게 수용할 수 있는 네트워크를 내부적으로 지원해야 한다. 이와 같은 맥락에서 K-Cluster는 상급 공급한 다양한 기술들이 공통적으로 생성하는 트래픽의 종류에 따라 안정적이고 분리된 네트워킹 연결성을 제공할 수 있도록 K-Cluster의 네트워크를 크게 P+M/C/D의 구도로 설계하였다.
- o P+M (Power + Management\_ 네트워크: 운영자가 K-Cluster 내의 박스, 서비스 어플리케이션 등의 모든 요소를 관리하기 위해 사용하는 네트워크이다. 구체적으로 서버, 스위치 등의 물리적인 인프라 관점에서 관리 네트워크는 박스의 상태를 확인하고 원하는 상태로 설정하기 위한 SSH, Telnet 등의 원격 박스 접속/관리 트래픽과, 박스에 인터넷으로부터 소프트웨어를 설치/설정하기 위한 인터넷 트래픽이 전송되는 네트워크이다. 또한 K-Cluster 상의 개발자 입장에서는 본인이 개발하는 어플리케이션 서비스를 인프라에 설치/설정하고, 동작 중인 서비스 어플리케이션을 원격에서 관리하는 용도로 활용되는 네트워크이다. 이와 더불어 박스의 관리 측면에서 다수의 분산된 박스들로 구성된 K-Cluster/K-ONE Playground의 인프라 관리 자동화를 위해서는 원격에 배포한 박스의 전원을 제어하는 기술이 필수적이다. 현재 판매 중인 서버의 경우 IPMI (Intelligence Platform Management Interface), HP iLO 등의 기술들이 사실상 기본적으로 탑재되어 있는데, 이들은 박스에 별도의 Out-of-Band 인터페이스를 확보하여 박스의 전원이 꺼진 상태에서도 원격에서 박스의 각종 센서 값을 확인하거나 전원을 제어할 수 있다. 따라서 인프라/서비스의 관리를 위한 Management 네트워크와 원격의 박스 전원을 관리하기 위한 Power 네트워크를 하나의 Power + Management (P+M) 네트워크로 정의한다. P+M 네트워크는 원격에서 인프라/서비스 운영자가 접속 가능해야 하며, 필요 소프트웨어를 인터넷으로부터 내려 받기 때문에 인터넷 접속이 가능한 Public IP 주소를 할당하는 것이 일반적이다.
- o C (Control) 네트워크: K-Cluster 내부에서 인프라/서비스의 정상동작을 위해 구성요소 간 제어를 위한 메시지들이 전송되는 네트워크이다. 인프라 관점에서는 K-Cluster 상에 SDN/NFV/Cloud 소프트웨어가 적용되어 운영될 것이며, 이 때 소프트웨어의 모듈들은 상호 간의 제어 목적의 메시지를 주고 받아야 한다. 예를

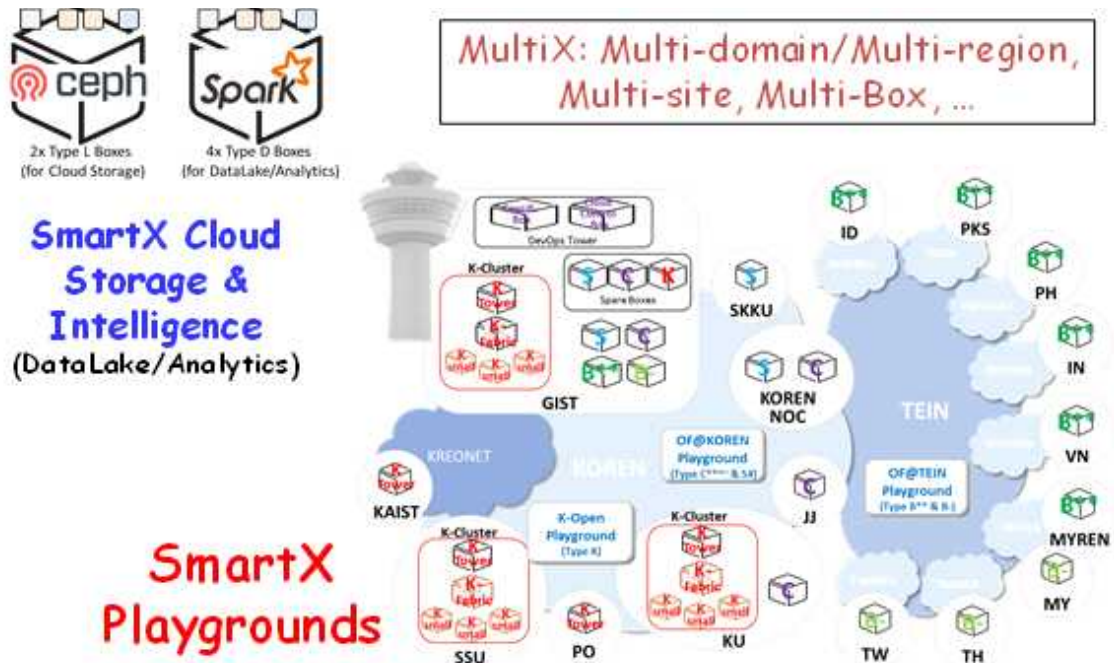


들어 SDN의 경우 중앙의 SDN 제어기는 인프라에 배포된 스위치들과 경로 및 박스 설정을 위한 오픈플로우, NetConf와 같은 제어 메시지를 Control 네트워크를 통해 주고 받는다. 오픈스택 Cloud의 경우 Control 노드, Compute 노드로 구성되며 Compute 노드들은 Control 노드와 제어 메시지를 주고받으며 VM 인스턴스를 생성하고 가상 네트워크를 설정한다. 이를 위한 제어 메시지가 결국 Control 네트워크를 통해 전송되는 것이다. 서비스 어플리케이션 관점에서 바라보면 근래 하나의 서비스를 독립적으로 동작하는 다수의 서비스들로 나누고 이를 엮음으로써 기능을 제공하는 마이크로 서비스가 일반화 되고 있다. 이러한 마이크로 서비스 구조에서 각 소규모 서비스들은 상호 간에 제어 메시지를 주고받음으로써 하나의 큰 서비스를 구성하게 된다. 이 때 마이크로 서비스가 K-Cluster에서 동작하는 경우 각 서비스 간의 주고받는 메시지들은 Control 네트워크를 통해 전송된다.

- o D (Data) 네트워크: K-Cluster 내부에서 서비스 간 실제 데이터들이 전송되는 네트워크이다. 일반적으로 인프라 자체 보다는 인프라 상에서 동작하는 서비스들이 연산을 위한 데이터를 상호 교환할 때 사용되는 네트워크이다. Service Function Chaining이 적용된 NFV 환경을 고려할 때 체인을 따라 전송되는 데이터 패킷이 결국 Data 네트워크를 통해 전송되는 것이다. Cloud의 경우 서비스 어플리케이션을 수용하고 있을 때 연산작업을 위해 VM 간의 East-West 트래픽이 발생하게 되는데, 이 트래픽이 Data 네트워크 상에서 전송된다.
- o 상기한 각각의 P+M/C/D 네트워크는 비록 트래픽 패턴에 기반한 논리적인 설계로 반드시 3개의 독립적인 네트워크로 분리하지 않아도 된다. 하지만 각각의 네트워크 별 트래픽을 분리하도록 물리적으로 구성하는 것을 추천한다. 네트워크를 나누는 방법에는 VLAN, 오버레이 네트워킹, L3 분할 등 여러 가지가 있을 수 있으나, 향후 구축 사례(4.1절)에서 볼 수 있듯 K-Cluster의 요소별 연결성을 제공하는 K-Switch 상에 각 네트워크 별 VLAN을 나누는 방식을 사용하였다.
- o K-Cluster내에 각각 독립적인 P+M/C/D 네트워크를 구성함에 따라 각각 다른 네트워크 구성을 요구하는 다양한 인프라 소프트웨어를 K-Cluster에 쉽게 대응하여 적용 가능하다. 이와 같은 네트워크 설계를 통해 하나의 고정된 하드웨어 묶음을 활용하여 SDN, NFV, Cloud, BigData 등의 다양한 서비스들을 유연하게 수용할 수 있다. 또한 여러 개의 독립적인 네트워크를 활용해 특정 네트워크에 장애가 발생한 경우 다른 네트워크를 활용한 고가용성을 향상시키는 연구로 확장도 가능하다.

### 3. K-Cluster 기반의 K-ONE Playground

#### 3.1. K-ONE Playground 필요성



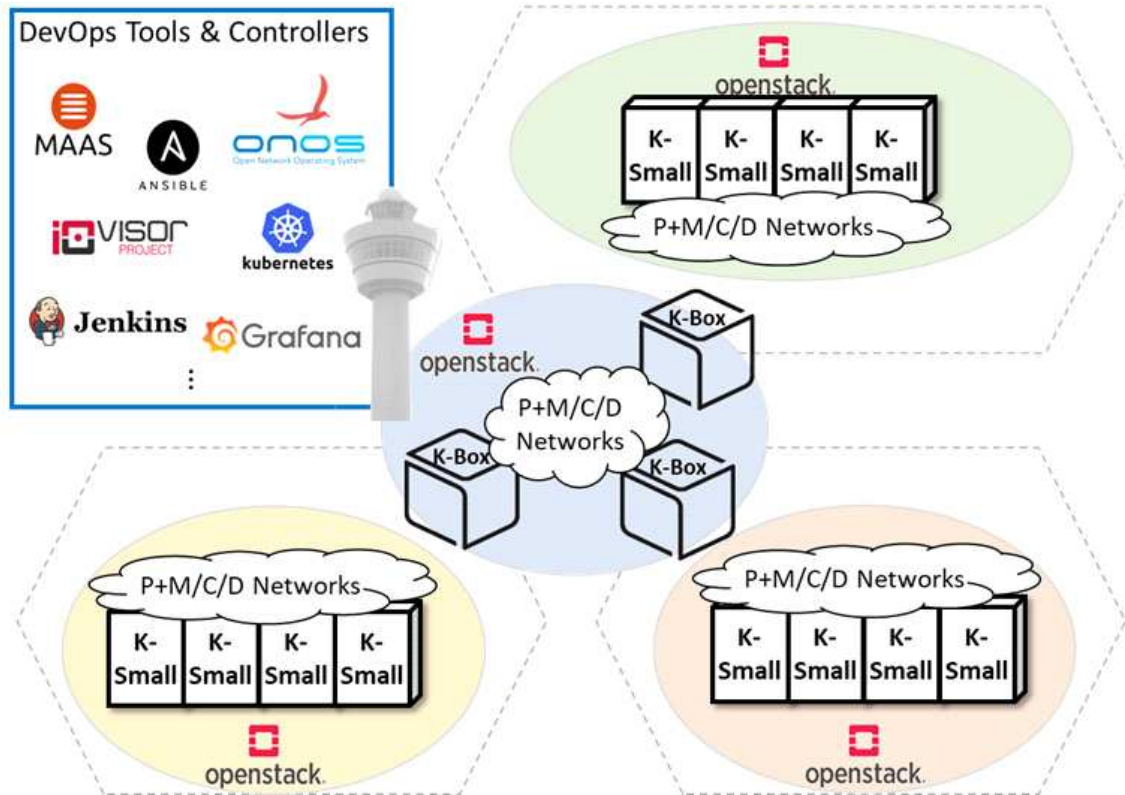
<그림 3: 소프트웨어 정의 인프라 패러다임의 MultiX Challenges>

- 고립된 하나의 박스, 하나의 클러스터, 하나의 사이트를 고도화 하는 전략은 기술적으로 이미 많이 성숙되어 자원의 용량 및 성능 측면에서 큰 향상을 기대하는 것은 어려우며, 근본적으로 자원 활용의 유연성/확장성 측면의 커다란 단점을 내포하고 있다. 게다가 다수의 멀티 자원요소들 간의 연결성을 제공하는 클러스터링, 네트워킹 기술이 고도화 됨에 따라 다수의 요소를 연결함으로써 발생하는 병목현상, 성능저하 보다 자원의 클러스터링을 통한 유연하고 확장성 있는 활용이 큰 가치를 갖게 되었다. 따라서 최근에는 <그림 3>의 우측 상단과 같이 멀티박스를 넘어서 멀티리전, 멀티사이트, 심지어 멀티도메인과 같이 다양한 멀티 이슈에 대응하는 MultiX가 ICT 인프라의 연구 키워드의 하나로 부상하고 있다.
- 소프트웨어 정의 인프라 패러다임의 핵심 기술인 SDN, NFV, Cloud의 경우에도 그동안의 연구개발을 통해 단일 클러스터, 단일 사이트를 대상으로 기술이 상당 수준 성숙되었다. 그리고 이를 시장의 대규모 인프라 상에 도입하기 시작하면서 상기 기술들이 다수의 사이트, 다수의 소유자들을 연결하는 멀티리전, 멀티사이트, 멀티도메인 기술로 근본적인 이슈 도메인을 변화해 나가고 있다. 따라서 오늘날의 SDN/NFV/Cloud 기술은 곧 MultiX 이슈와 연계한 연구 개발이 요구되며, 이를 위한 테스트베드가 필요하다.

- o 단일 K-Cluster를 활용하면 단일 클러스터/사이트를 범위로 한 SDN/NFV/Cloud의 통합적인 실증이 가능하나, 앞서 언급한 MultiX에 대응하는 실증을 지원하는 것이 어렵다. 따라서 K-Cluster들을 지리적, 네트워크 적으로 구분된 사이트들에 분산 배포하고 이를 유연하고 신뢰성 있는 네트워크로 연계함으로써 하나의 MultiX 테스트베드로 구성하는 것이 필요하다. 그리고 소프트웨어 도구들을 활용해 인프라의 구성을 자동화함으로써 다양한 SDN/NFV/Cloud 실증 환경을 쉽고, 빠르고, 유연하게 제공하는 테스트베드가 필요하다.
- o 따라서 상기 요구사항에 대응하여 각 K-Cluster를 다수의 사이트에 분산 배포하고 이를 연구망으로 엮어 구성한 멀티사이트 테스트베드인 K-ONE Playground가 필요하다.

### 3.2. 멀티사이트 실증을 위한 K-ONE Playground 구성안

- o 다양한 멀티사이트 SDN/NFV/Cloud 들은 기본적으로 지리적, 네트워크 적으로 분할된 다수의 사이트들에서 발생하는 다양한 문제를 다루는 것을 목표로 한다. 따라서 이러한 멀티사이트 실증을 지원하기 위해 K-ONE Playground를 하나의 L2 네트워크로 묶는 것 보다는, 각 사이트 별로 구분된 L3 네트워크 서브넷에 연결된 구성이어야 한다. 독립적인 네트워크 환경을 기반으로 K-ONE Playground의 각 사이트들은 독립적인 SDN/NFV/Cloud 인프라를 구성하며, 이러한 각 사이트를 하나의 인프라로 연결하는 실제 환경에서 기인하는 멀티사이트 이슈를 K-ONE Playground 상에서 개발/실증할 수 있도록 지원해야 한다.
- o 비록 각 사이트는 독립적인 인프라 환경으로 구성되어야 하지만, 멀티사이트 환경인 K-ONE Playground를 효율적으로 운영하기 위해서는 중앙에서 전체 사이트들을 다양한 오픈소스 DevOps 도구들을 활용하여 자동화된 관제를 담당하는 Tower가 필요하다. K-ONE Playground에서는 특별히 확보된 박스를 활용하는 것이 아니라 각 K-Cluster의 K-Box (K-Tower)를 묶어 하나의 논리적인 Tower로 구축하는 것을 기본 설계로 하며, 이 논리적인 Tower를 K-DevOps Tower로 정의한다. 이 때 각 사이트에 분산된 K-Box들을 하나의 논리적인 Tower로 클러스터링하기 위해 OpenStack 클라우드를 활용해 K-Box들을 하나의 자원 풀로 클러스터링 한다. K-DevOps Tower의 각 K-Box는 자신이 속한 K-Cluster만 전담으로 관리하게 된다. 즉, Site A의 K-Box 상에 생성된 관제 VM은 Site B의 K-Cluster의 관제를 수행할 수 없게 구성되어야 한다. 다수의 사이트에 배포된 K-Box들을 하나의 OpenStack 클라우드로 구성하기 위해서는 각 K-Box들은 L2 네트워크로 직접 연동되어야 한다. 이 때 K-DevOps Tower 또한 상기 제시한 P+M/C/D 네트워크 설계에 따라 구성되어야 한다.



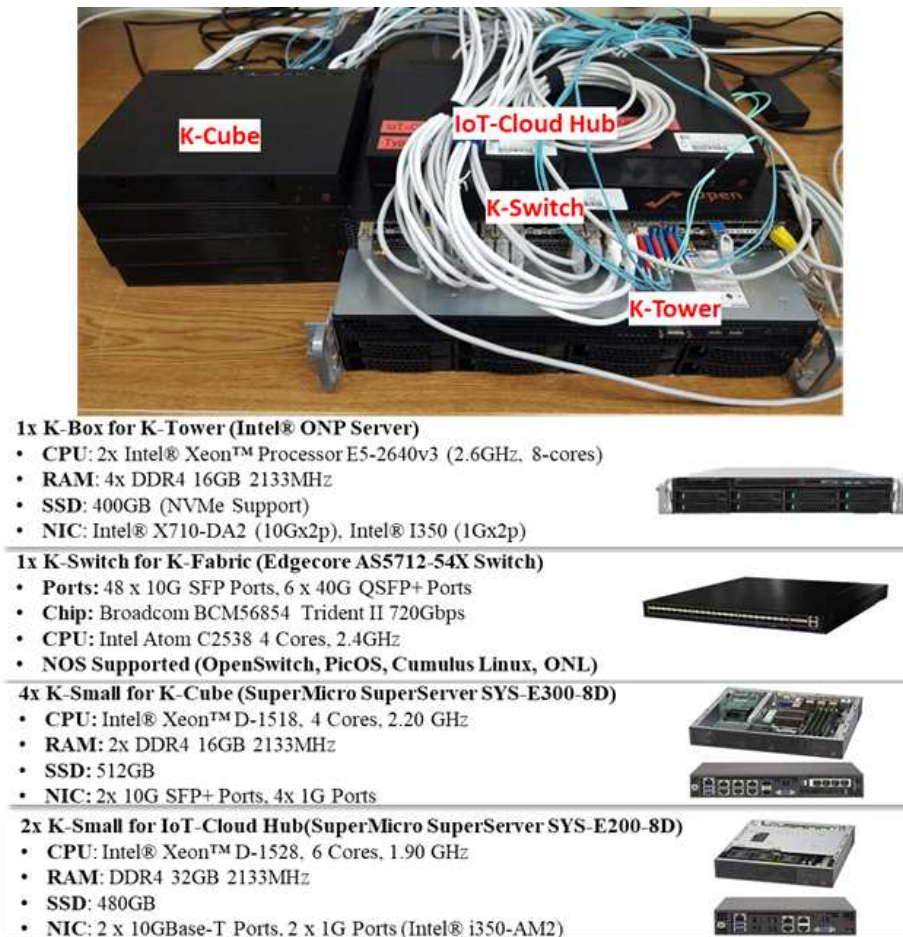
<그림 4: K-ONE Playground 구성 설계도>

- o 상기 내용을 바탕으로 K-ONE Playground의 설계를 <그림 4>와 같이 정리할 수 있다. 각 사이트는 독립적인 SDN/NFV/Cloud 인프라를 구축하고 이를 L3 네트워크로 연결 가능한 상황을 구성하여 멀티사이트 실증환경을 구축한다. 동시에 각 사이트의 K-Box들을 연결하여 하나의 논리적인 K-DevOps Tower로 구축한다. 이와 같은 설계에 따라 K-ONE Playground를 구성하면, K-DevOps Tower 상에 각종 DevOps 자동화 도구 및 SDN/NFV/Cloud 제어기를 배치함으로써 멀티사이트 Playground 상에 다양한 실증환경을 자동화 기반으로 쉽게 구성할 수 있으므로 효율적인 실증이 가능하다.

## 4. K-Cluster 및 K-ONE Playground 구축 현황

### 4.1. K-Cluster 하드웨어 시제품 구성

- o K-Cluster의 설계에 따라 K-Cluster 하드웨어 시제품을 아래 <그림 5>와 같이 구성하였다. 각 박스들은 요구사항에 기술하였듯 특정 벤더의 특수한 하드웨어를 포함하는 벤더 종속적인 모델이 아니며, 동일 수준의 다른 모델로 대체 가능한 화이트 박스들을 선택하여 구성하였다.



<그림 5: K-Cluster 하드웨어 시제품 상세사양 및 구축모습>

- o 상기 제시한 K-Cluster 하드웨어 시제품은 약 2000만원 내외의 구성으로, 이를 활용하여 다양한 SDI 실증을 유연하게 할 수 있음을 고려하였을 때, 상당한 경제성을 갖고 있음을 확인할 수 있다.
- o K-Cluster 하드웨어 시제품에서 구성요소인 K-Box, K-Small, K-Small 박스들은 K-Switch에 P+M/C/D 네트워크 설계에 따라 각각 4개의 UTP/Fiber 케이블로 연결되어 있으며, K-Switch 상에 P+M/C/D 네트워크 별로 VLAN ID를 할당하

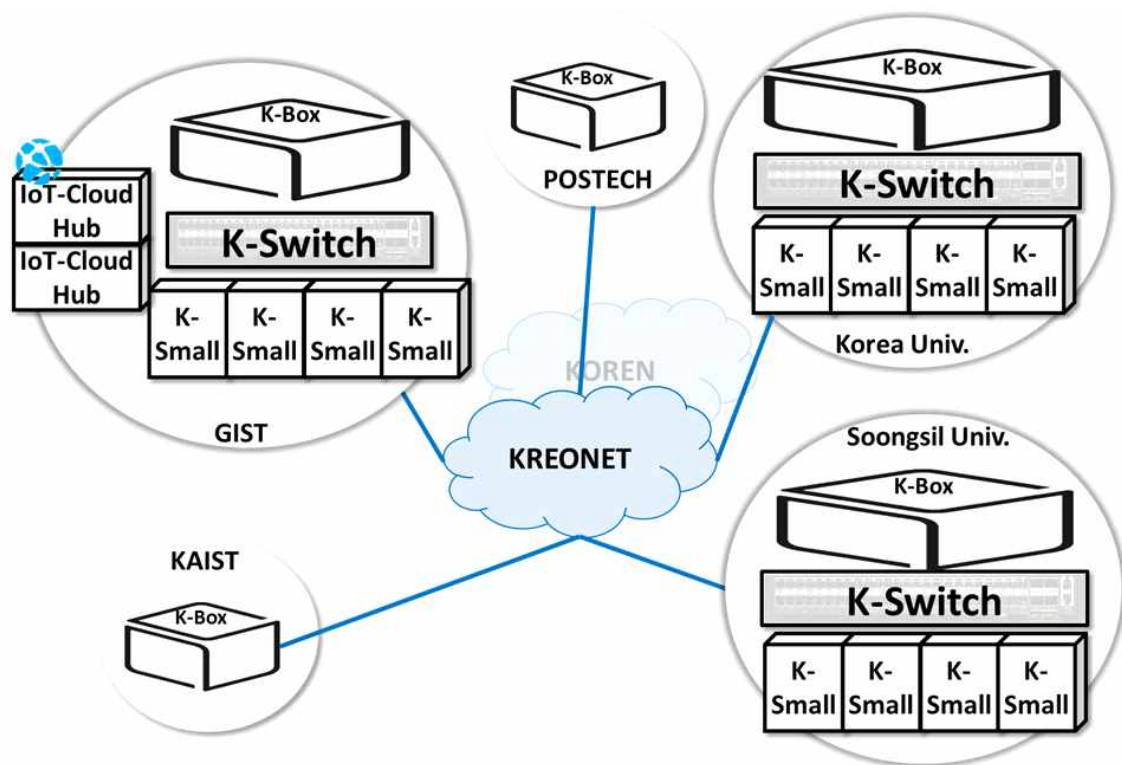


여 네트워크 간을 독립적으로 분리하였다.

- 제시한 K-Cluster 구성은 하드웨어 시제품 모델로써 시제품에서 활용한 박스들을 다른 모델로 대체하여도 K-Cluster의 구성 예시 중 하나에 해당한다. 즉, K-Cluster의 설계에 부합한다면 K-Cluster의 하드웨어 구현으로 본다.

#### 4.2. K-ONE Playground 구축

- K-ONE 컨소시엄 기관을 대상으로 멀티사이트 SDN/NFV/Cloud 통합 실증환경인 K-ONE Playground을 <그림 6>과 같이 구성하였으며, 이를 위해 아래와 같은 노력을 수행하였다.



<그림 6: K-ONE Playground의 구축 현황도>

- 우선 K-ONE Playground의 하드웨어 배치 측면에서는, 1차년도에 K-Box들을 전체 컨소시엄 기관 사이트들에 한 대씩 배포하였으며, 2차년도에는 NFV, Cloud를 연구주제로 수행하는 GIST, 숭실대학교, 고려대학교를 대상으로 K-Switch와 K-Small 들을 추가로 배포함으로써 앞서 제시한 K-Cluster 하드웨어 시제품을 세 사이트에 구성하였다.
- K-ONE 컨소시엄의 다섯 참여기관 사이트를 하나의 Playground로 구성하기 위하여 각 사이트의 여건에 맞추어 KREONET/KOREN 연구망을 들이는 작업을 진행

하였다. 결과적으로 1-2차년도에 걸친 노력으로 <그림 6>과 같이 각 사이트에 배포한 K-Box/K-Cluster는 KREONET/KOREN 연구망으로 모두 연동하였다. 특히 K-Cluster가 배포된 GIST, 숭실대학교, 고려대학교는 별도의 KREONET Public IP 대역을 각각 할당받아 사이트별로 독립적인 실증환경을 구축하였다.

- o K-DevOps Tower의 구축을 위해 P+M/C/D 네트워크에 대응하는 3개의 추가 L2 네트워크를 GIST, 숭실대학교, 고려대학교 사이트를 대상으로 KREONET의 협조를 받아 추가로 연동하였다. 확보된 세 개의 L2 네트워크에 K-Box를 연결하고 이를 하나의 OpenStack 클라우드로 구성하였다. K-DevOps Tower의 대시보드 GUI를 통해 세 사이트에 손쉽게 관리용 VM을 배포할 수 있도록 구축을 완료하였다. <그림 4 참고>

## 5. K-ONE Playground 활용한 SDI 기술 실증

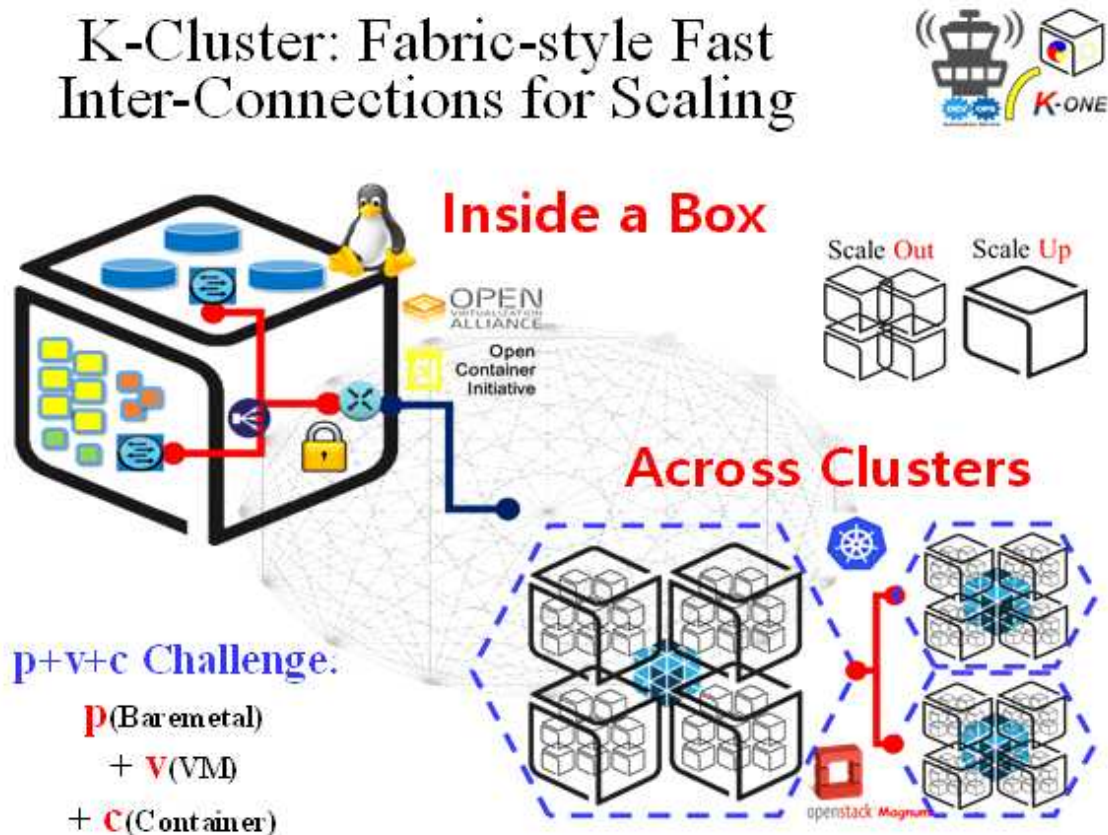
### 5.1. K-Cluster 활용 실증: 인텔리전스 서비스

- o 최근 다양한 기기들과 자연/사회 현상들로부터 발생하는 막대한 데이터를 HPC/BigData 기술과 접목하여 분석/가공하여 정보(information), 지식(knowledge), 지능(intelligence) 등으로 정리하고 이를 기반으로 사용자들을 위한 다양한 서비스들과 접목하여 고도화된 지능형 서비스들로 변모 시키는 것이 대세로 자리잡고 있다. 이에 따라 SDI 환경을 구성하는 융합형 박스들을 클러스터 형식으로 하나로 묶은 후에 계산 중심의 HPC 클러스터 기능과 데이터 중심의 BigData 클러스터 기능을 자유롭게 넘나들면서 지원하는 것이 필요하다. 또한 자유롭게 넘나드는 클러스터 부분을 급속도로 확산하는 클라우드 데이터 센터 차원으로 확대해서 살펴보면, 대용량의 데이터의 저장을 선도하는 아마존 AWS, 마이크로소프트 Azure 등의 클라우드 진영이 슈퍼컴퓨팅 영역에서 선도하던 고성능/대용량 계산을 위한 고급 자원집합을 온-디맨드(On-demand) 형태로 제공하는 것까지 확대하는 추세이다.
- o BigData/FastData 클러스터와 HPC 클러스터가 감당해야 하는 다소 이질적인 워크로드(workload)들을 하나의 융합형 박스 클러스터를 가지고 처리하도록 구성하는 것은 매우 어려운 작업이다. 즉 하이퍼바이저(hypervisor) 중심으로 가상화 기술이 발전하여 고도화됨에 따라서 계산/저장/네트워킹의 모든 측면에서 가상화가 가능하도록 하는 수준에 근접하고 있다. 이러한 추세에 부합하면서 융합형 박스 기반의 자원집합이 등장하고 확산하기 이전에는, 대용량/실시간 계산 또는 빅데이터 워크로드만을 위해 특별하게 제작된 전용 클러스터가 반드시 필요했다. 전용 클러스터는 고가의 하드웨어 자원들과 특화된 소프트웨어 자원이 요구된다. 그러나 계산/저장/네트워킹을 하나의 박스에 담고 박스 숫자를 늘리는 융합형 자원집합의 등장과 함께, HPC 및 BigData/FastData 클러스터가 요구하는 워크로드를 통합적으로 처리하는 것이 용이해 지고 있다.
- o 따라서 HPC 및 BigData/FastData 클러스터를 융합형 박스기반으로 통합적으로 제공하는 K-Cluster구성이 필수적이다. 다른 측면에서도 K-Cluster 는 SDI 참조모델의 에지 클라우드에 대응하는 개념으로써 좌측으로는 IoT를, 우측으로는 코어 영역의 클라우드의 두면서 양쪽 사이의 중계 역할도 수행해야 한다. 이를 위해 K-Cluster는 네트워킹 측면의 연결성 제공을 위한 SDN/NFV기능의 제공뿐만 아니라, IoT 사물들로부터 생성되는 막대한 데이터를 일차적으로 처리하는 역할과 함께 코어 클라우드에서 수행되는 막대한 양의 계산을 위한 보조 자원으로써 동작해야 한다. 그러므로 융합형 자원박스로 구성하고 오픈소스 소프트웨어를 적용한 K-Cluster 를 활용함으로써, IoT 데이터를 위한 FastData 기능과 함께 코어 클라우드의 보조를 위한 HPC/BigData 클러스터 기능을 지원하도록 준비해야 한다.



## 5.2. K-Cluster 활용 실증: 박스 내/박스 간 Fast Inter-connections

o K-Cluster를 중심으로 하는 SDI에서는 지속적으로 증가하는 자원의 요구에 대응하기 위한 다양한 규모의 확장(scale-out)이 요구된다. <그림 7>에 제시한 것과 같이, 가상화 및 박스 클러스터링 개념과 관련 기술들이 급격히 발전함에 따라 박스 자체를 키워서 내부 자원을 향상/보충하는 Scale-up 방식의 규모 확장과 더불어, 박스 숫자를 늘리고 이를 클러스터링 하여 하나의 논리적인 박스처럼 활용하는 병렬적인 규모 확장인 Scale-out 방식이 자리잡고 있다. 따라서 K-Cluster는 SDI 상의 다양한 서비스를 유연하게 수용하도록 상기한 Scale-up 및 Scale-out 자원집합의 확장에 통합적으로 대응해야 한다. 가장 기초적인 수준으로는 K-Cluster를 구성하는 K-Box, K-Switch, K-Small 박스의 내부 하드웨어 자원을 확충하는 박스 수준의 Scale-up이 가능하다. 그리고 K-Cluster의 박스 숫자를 늘리고 고속의 K-Fabric을 통해 박스들을 연동하는 것도 가능하다. 따라서 박스에 기반한 자원집합의 확충이 하나의 거대한 논리적 박스의 내부 자원을 확충한 것과 동일하게 만들어지므로 박스 수준의 Scale-out이자 클러스터 수준의 Scale-up 구도로 K-Cluster의 효과적인 규모 확장이 가능하다.



<그림 7: 박스 내부와 다수의 박스 간 빠른 연결 이슈를 포괄하는 K-Cluster>

o 또한 하나의 K-Cluster 내부만을 바라보는 국지적인 시야에서 벗어나 SDI 전체의

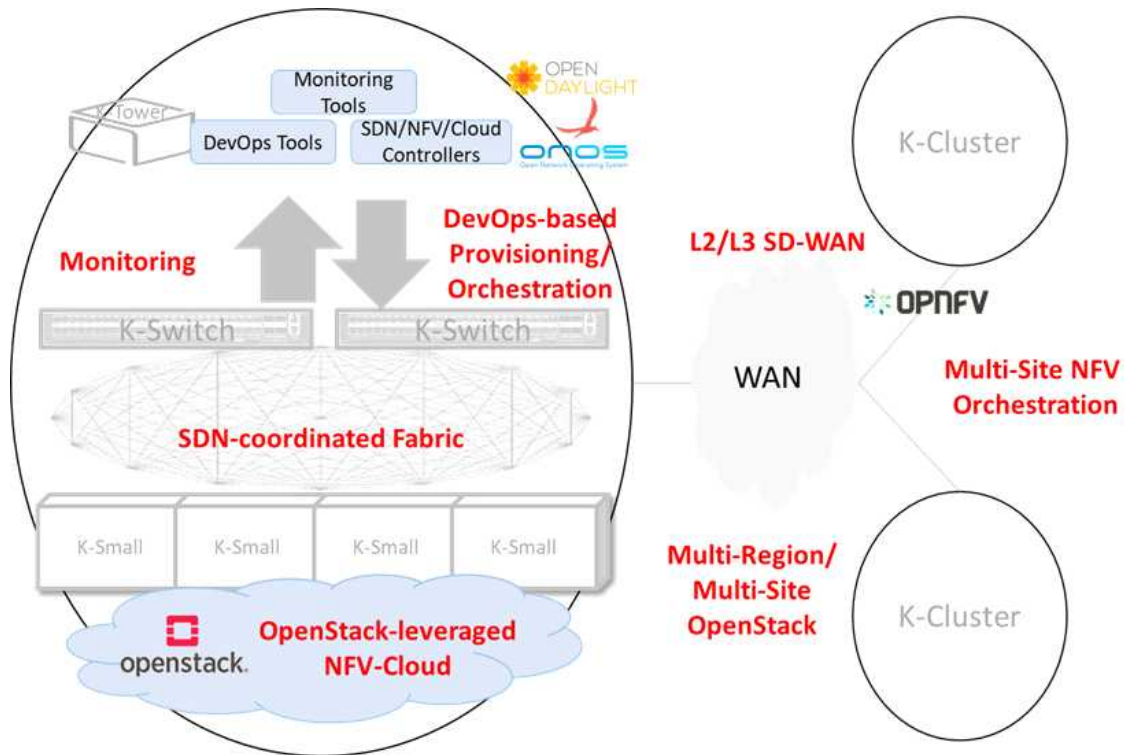
관점에서 규모 확장 이슈를 바라보는 시야도 필요하다. K-Cluster가 대응하는 예지 클라우드 관점에서 바라보면, K-Cluster는 중단에 산재한 IoT환경에 직접적으로 대면하는 소규모의 클라우드 환경으로써 각각의 클러스터가 다수의 사이트에 분산 배치된다. 따라서 다수의 사이트들에 배포된 다수의 K-Cluster들을 SDI 전체적인 시점에서 바라본다면, K-Cluster들의 자유로운 Scale-out을 통해 SDI 전체적인 규모의 확장도 용이해 진다. K-Cluster 중심의 SDI에서는 중앙에 위치한 DevOps Tower가 멀티사이트에 산재한 K-Cluster들을 총괄하여 관리한다. K-Cluster의 Scale-out을 위해 K-Cluster의 K-Tower에 위치하는 SmartX Agent가 DevOps 타워와 상호 통신을 함으로써, 인프라 운영자는 하나의 DevOps 타워를 통해 멀티사이트 인프라를 손쉽게 확장하고 관제할 수 있다.

- o 추가로 SDI 전체적인 규모의 확장은 이슈의 복잡성과 대상 환경의 다양성으로 인하여 이를 체계적으로 지원하는 기반 기술이 준비되어야 한다. 규모 확장에서 클러스터의 Scale-up은 자원의 클러스터링 및 오케스트레이션이 핵심적으로 요구된다. 또한 클러스터의 Scale-out은 박스들 간이나 멀티사이트 간의 자동화된 고속의 연결성 확보가 기반이 된다. 관련하여 단일 또는 복수개의 사이트들의 수많은 자원 박스들을 클라우드 차원에서 연계(federation)하는 OpenStack 클라우드 운영체제도 중요하다. 규모를 줄여서 단일 사이트에 위치한 다수의 이질적인 자원 박스들을 클러스터링 하도록 지원하는 Apache Mesos, Kubernetes 등의 컨테이너 지향적인 오픈소스 소프트웨어 프로젝트들이 성장함에 따라 클러스터 수준의 Scale-up도 상당 부분 진행된 상태이다. 최근에는 멀티사이트 환경에서 클러스터들을 연계하는 방식의 클러스터 Scale-out을 위한 멀티사이트 기술도 화두가 되고 있으며, 이를 위한 다양한 시도들이 여러 SDN/NFV/Cloud 관련 오픈소스 소프트웨어 프로젝트들에서 모멘텀을 얻고 있다.

### 5.3. K-ONE Playground 활용 실증: SDN/NFV/Cloud 통합 실증방안

- o 그림 8은 K-ONE Playground를 활용하여 실증 가능한 다양한 서비스 예시를 통합적으로 정리하여 보여주고 있다.
- o 단일 사이트에서 발생하는 SDN/NFV/Cloud 이슈에 대응하기 위한 방안을 우선 고려해 보자면, K-Cluster 내에 K-Cube 상에 오픈소스 클라우드 운영체제인 오픈스택 클라우드 환경을 구성한다. 그리고 VNF들의 라이프사이클 관리를 위해 오픈스택 내 VNF Manager/NFV Orchestrator 프로젝트인 오픈스택 Tacker를 활용함으로써 NFV-Cloud 통합 환경을 구축한다. 그리고 K-Fabric을 중심으로 한 클러스터 패브릭 상의 플로우를 SDN 제어를 통해 제어하도록 구성한다. 이 때 K-Cluster 구축하고 관리하는 전반적인 작업을 자동화하는 DevOps 도구들을 K-Tower 상에 준비하여, 운영자는 K-Tower에서 K-Cluster의 SDN/NFV/Cloud

전체를 자동화 기반으로 관제할 수 있다. 이처럼 K-Cluster를 활용하면 K-Tower를 통해 SDN/NFV/ Cloud 통합 인프라 환경을 DevOps 소프트웨어 도구를 활용하여 자동화된 관제를 수행하는 소프트웨어-정의 인프라 패러다임에 대응하는 실증이 가능하다.

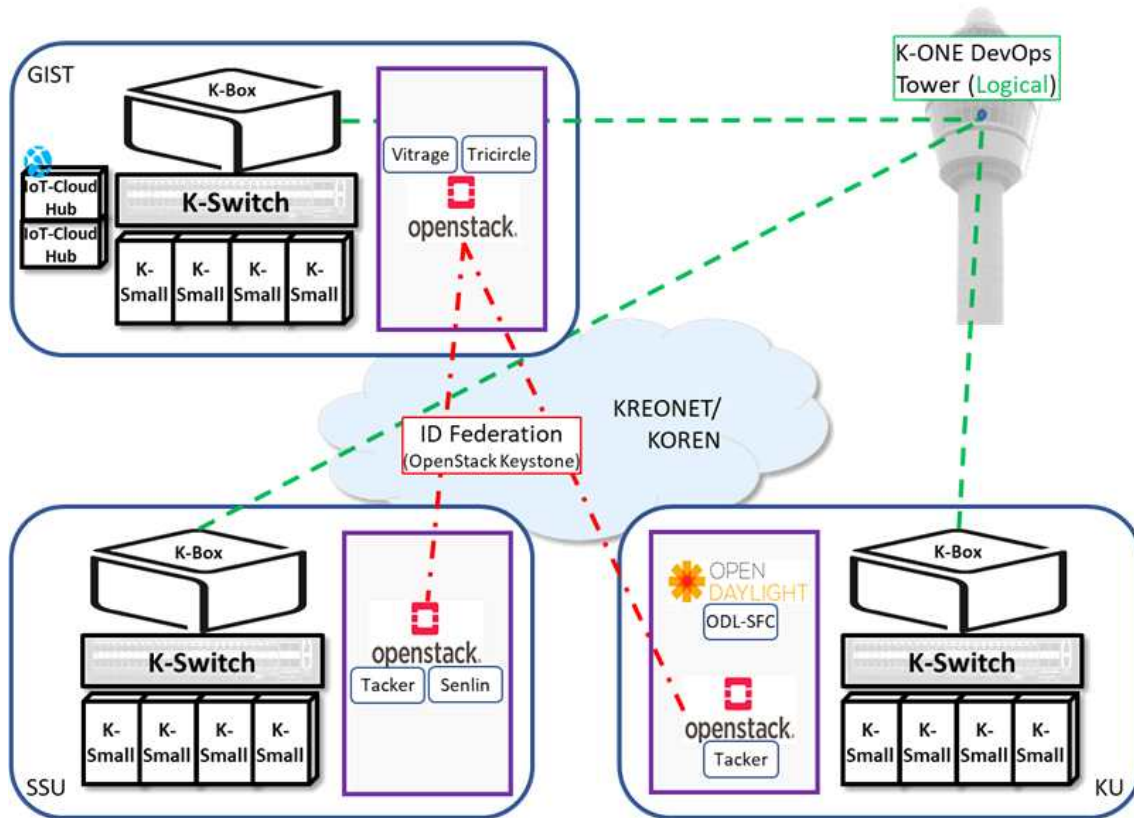


<그림 8: K-ONE Playground를 활용한 SDN/NFV/Cloud 통합실증방안>

- o 멀티사이트 기반의 K-ONE Playground를 활용함으로써 다음과 같은 소프트웨어-정의 인프라의 MultiX 실증이 가능하다. 멀티사이트 실증을 위해 사이트 별 클라우드 환경을 구성한 후 사이트 간 상호연결성(Inter-connect)을 위해 각 사이트별 가상스위치/라우터 VNF (Virtual Network Functions)를 자동으로 배치한다. 그리고 SDN 제어기로 터널링 기반 L2 연결 또는 BGP 기반 L3 연동을 자동화하는 SD-WAN 기술을 적용함으로써 멀티사이트 SDN/NFV/Cloud 인프라 실증이 가능하다. 또는 각 사이트별로 구성된 독립적인 오픈스택 클라우드의 인증 정보를 중앙에서 관리하는 멀티리전/멀티사이트 실증을 할 수 있다. 각 리전은 기본적으로 별개의 독립 환경으로 다른 리전 상의 VM간 네트워킹이 불가하지만, SD-WAN 기술과 멀티리전 네트워킹 자동화 프로젝트인 오픈스택 Tricircle 프로젝트를 적용함으로써 멀티사이트 환경을 실증할 수 있다.
- o 이처럼 단일 또는 다수의 K-Cluster를 활용하면 SDN/NFV/Cloud의 다양한 통합 서비스를 적용한 유연한 실증이 가능하며, 나아가 다수의 K-Cluster를 활용하면

실제 소프트웨어-정의 인프라의 MultiX 대응하는 실증이 가능하다.

#### 5.4. K-ONE Playground 활용 실증: SDN/NFV/Cloud 오픈소스 개발 및 검증



<그림 9: K-ONE Playground를 활용한 오픈소스 소프트웨어 개발계획>

- o K-ONE 과제 측면에서 K-ONE Playground는 국내 다양한 연구개발 기관에서 개발하는 SDN/NFV/IoT-Cloud 관련 오픈소스 소프트웨어를 공유, 협업하기 위한 시스템을 제공하는 기반 환경을 제공함으로써 한국형 오픈소스 생태계를 조성하는 것을 목표로 하고 있다. 따라서 K-ONE 컨소시엄 참여기관 별로 그림 9의 구도에 따라 K-ONE Playground 상에서 NFV, Cloud 오픈소스 프로젝트를 개발 및 검증하고 이를 Code 수준에서 Contribution하는 노력을 수행하고 있다.
- o 현재 K-ONE Playground가 주로 NFV, Cloud 오픈소스 프로젝트를 개발하는 환경으로 활용되고 있으나, 향후 K-ONE Playground가 SDN을 다루는 다른 두 참여기관으로 확장될 계획이고, 참여기관의 SDN의 Contribution 범위가 멀티사이트로 확장되고 있으므로, 조만간 SDN을 포함하는 SDN/NFV/Cloud 통합 실증이 K-ONE Playground 상에서 수행될 것으로 기대하고 있다.

## 6. 결론

- o 본 기술문서를 통해 소프트웨어 정의 인프라 패러다임에 준비하기 위한 Edge-Cloud에 대응하는 테스트베드 모델인 K-Cluster을 설계에 대해 상세히 설명하였다. 그리고 설계에 따라 구축한 K-Cluster 하드웨어 시제품의 구성 및 상세 스펙에 대해 기술하였다.
- o K-ONE 컨소시엄이 목표로 하는 멀티사이트 SDN/NFV/Cloud 실증을 유연하고 효율적으로 지원하기 위한 K-ONE Playground의 설계를 제시하였으며, 본 설계에 따라 일차적으로 GIST, 숭실대학교, 고려대학교를 대상으로 구축한 K-ONE Playground 환경을 상세히 설명하였다.
- o 구축한 멀티사이트 K-ONE Playground를 활용한 다양한 SDI 실증 방안에 대해 제시하였다. 따라서 본 기술문서에서 제안한 K-Cluster/K-ONE Playground 모델에 따라 인프라 환경을 구축하면 현재 SDI 패러다임에 대응하는 다양한 기술을 적용할 수 있음을 확인할 수 있다.

## References

- [1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer networks, 2010.
- [2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Communications of ACM, vol. 53, no. 4, pp. 50-58, April 2010.
- [3] Peter C. Evans and Marco Annunziata, "Industrial Internet: Pushing the boundaries of minds and machines," General Electric White Paper, Nov. 2012.
- [4] M. Satyanarayanan, et al., "The Case for VM-Based Cloudlets in Mobile Computing," IEEE Pervasive Computing, vol. 8, no. 4, pp. 14-23, Oct. 2009.
- [5] A. Bremler-Barr, Y. Harchol, and D. Hay, "OpenBox: Enabling Innovation in Middlebox Applications," In Proc. of the 2015 ACM SIGCOMM Workshop on Topics in Middleboxes and Network Function Virtualization, ACM, pp. 67-72, Aug. 2016
- [6] L. Peterson, A. Al-Shabibi, T. Anshutz, S. Baker, A. Bavier, S. Das, J. Hart, G. Palukar, and W. Snow, "Central Office Re-Architected as a Data Center," IEEE Communications Magazine, IEEE, vol. 54, no. 10, Oct. 2016.

## *K-ONE* 기술 문서

- K-ONE 컨소시엄의 확인과 허가 없이 이 문서를 무단 수정하여 배포하는 것을 금지합니다.
- 이 문서의 기술적인 내용은 프로젝트의 진행과 함께 별도의 예고 없이 변경될 수 있습니다.
- 본 문서와 관련된 문의 사항은 아래의 정보를 참조하시길 바랍니다.  
(Homepage: <http://opennetworking.kr/projects/k-one-collaboration-project/wiki>, E-mail: [k1@opennetworking.kr](mailto:k1@opennetworking.kr))

작성기관: K-ONE Consortium  
작성년월: 2017/04