IoTcloudServe@TEIN 3rd Collaboration Community Meeting
8 September 2020, Thailand-Laos-Korea

Maintaining SmartX Operations for OF@TEIN+ Distributed Cloud-native Edge Boxes

Muhammad Ahmad Rathore
ahmad@smartx.kr

Networked Intelligence Lab
School of Electrical Engineering and Computer Science (EECS)
Gwangju Institute of Science and Technology (GIST)
Gwangju, South Korea

Research Work

- [Conference][Published] *"Persistent Operation of OF@TEIN+ Playground Verified by SmartX Multi-View Leveraged Visualization"*, in Proc. 2020 International Conference on Future Computer and Communication (ICFCC 2020), Yangon, Myanmar, Feb. 2020

- Journal][Accepted] *"Maintaining SmartX Multi-View Visibility for OF@TEIN+ Distributed Cloud-native Edge Boxes"*, Transactions on Emerging Telecommunications Technologies (ETT), WILEY, (2020)

Outline

- OF@TEIN+ Playground: Overview
  - Cloud-native Computing
  - Production-focused Playground Tower
- Maintaining Operations of OF@TEIN+ Playground
  - Step 0: Requirements and Issues
  - Step 1: Maintaining persistence SmartX Multi-View Visibility
  - Step 2: Design for Maintaining Persistent Playground Visibility
  - Step 3: Maintaining Persistent Playground Operations (Visibility) for Service Developers and Operators
  - Step 4: Results and Verification
    - [4-1] Pre-processing Visibility Data
    - [4-2] Maintaining SmartX Multi-View Visibility : Results of Implementation Use Cases
    - [4-3] Spatio-temporal Summarized Visualization
    - [4-4] Verifying SmartX Containerized IoT-Cloud service with IoT Devices

CHULA ENGINEERING Foundation toward Innovation

NECTEC a member of NSTDA

UNIVERSITY OF MALAYA

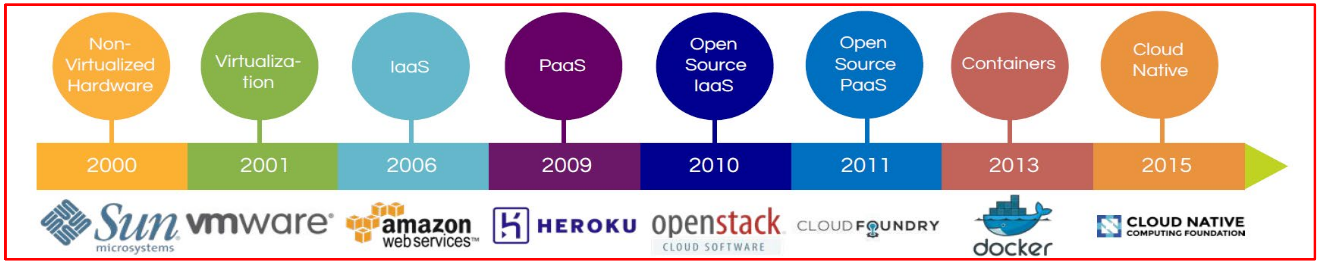
Asi@Connect

European Union

TEIN*CC COOPERATION CENTER

Cloud-native Computing

CLOUD NATIVE COMPUTING FOUNDATION



Application Definition / Development

Orchestration & Management

Runtime

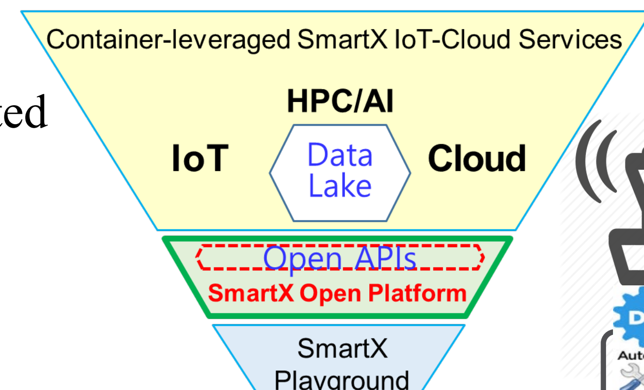
Provisioning

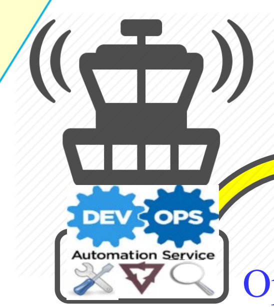
*Infrastructure (Bare Metal/Cloud)

microservices + automation + DevOps
➔➔
Agility of software teams + Resilience of applications.

Cloud native computing uses an open source software stack to be:

- Containerized
- Dynamically orchestrated
- Microservices oriented





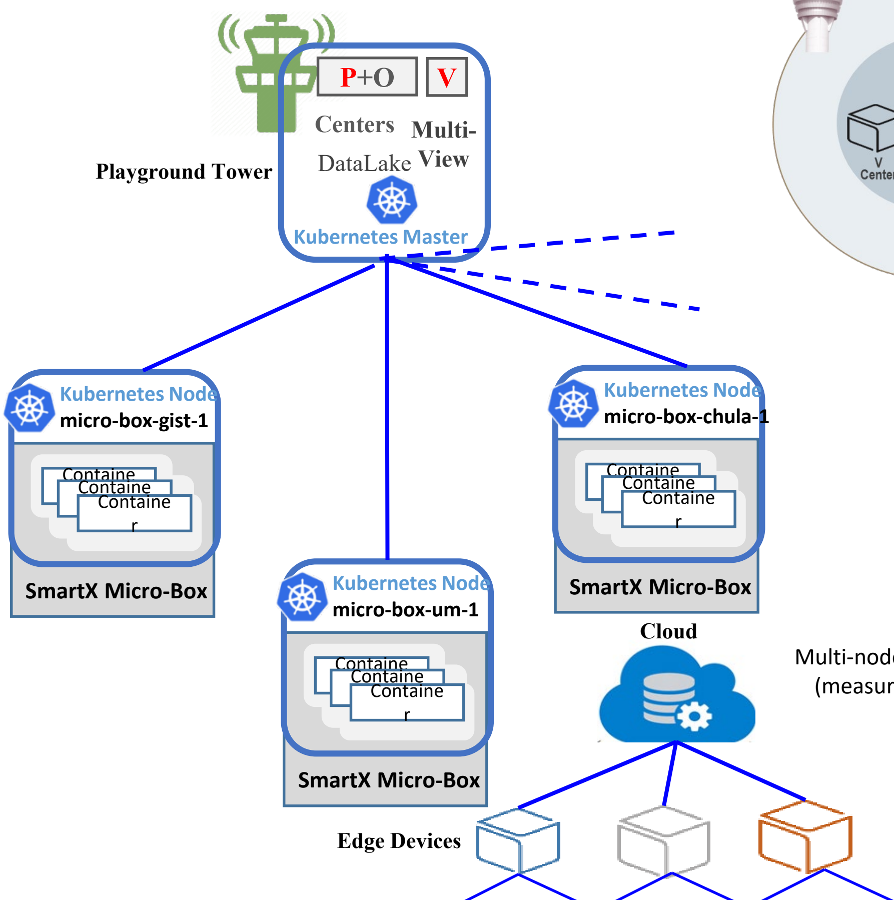
rkt

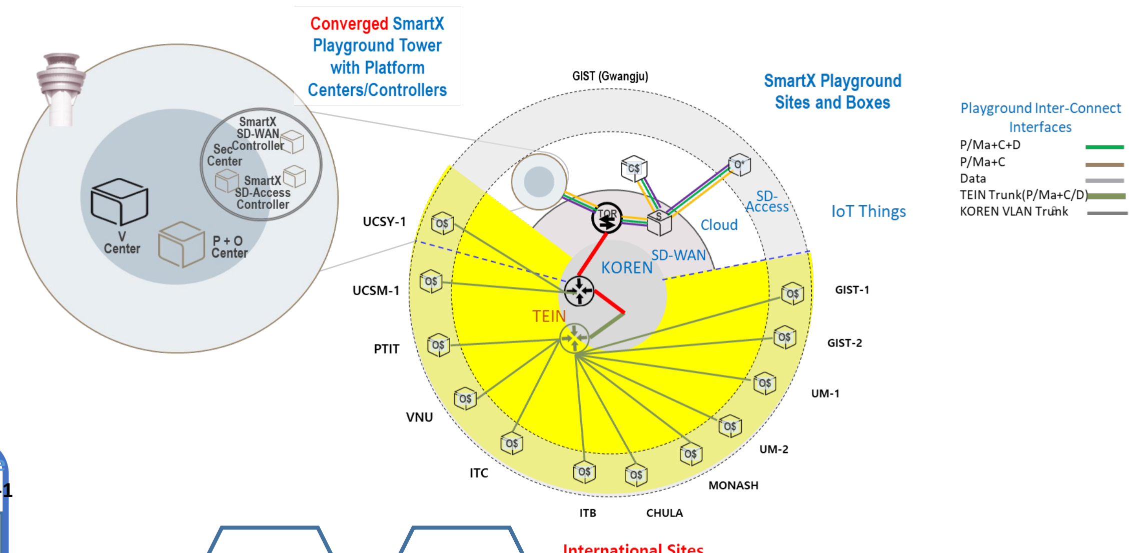
Service Mesh Interface

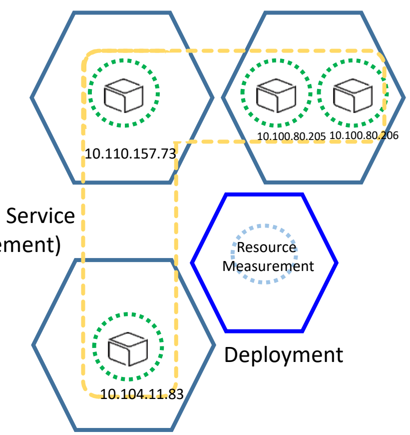
SmartX Open Playgrounds

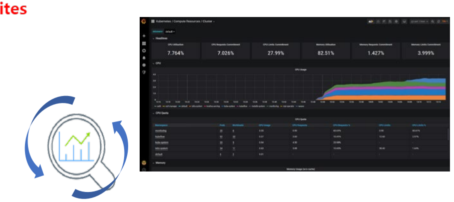
OF@TEIN+ Playground (2019)

(Production-focused Playground Tower for SmartX Micro-Boxes Operation)









- 14 SmartX Micro-Boxes
- Enable distributed cloud-native (Kubernetes-orchestrated) playground
- Visibility-enabled (i.e., monitored) and
- Jointly operated by federated playground towers

Cloud-native Edge Playground: Requirements and Issues

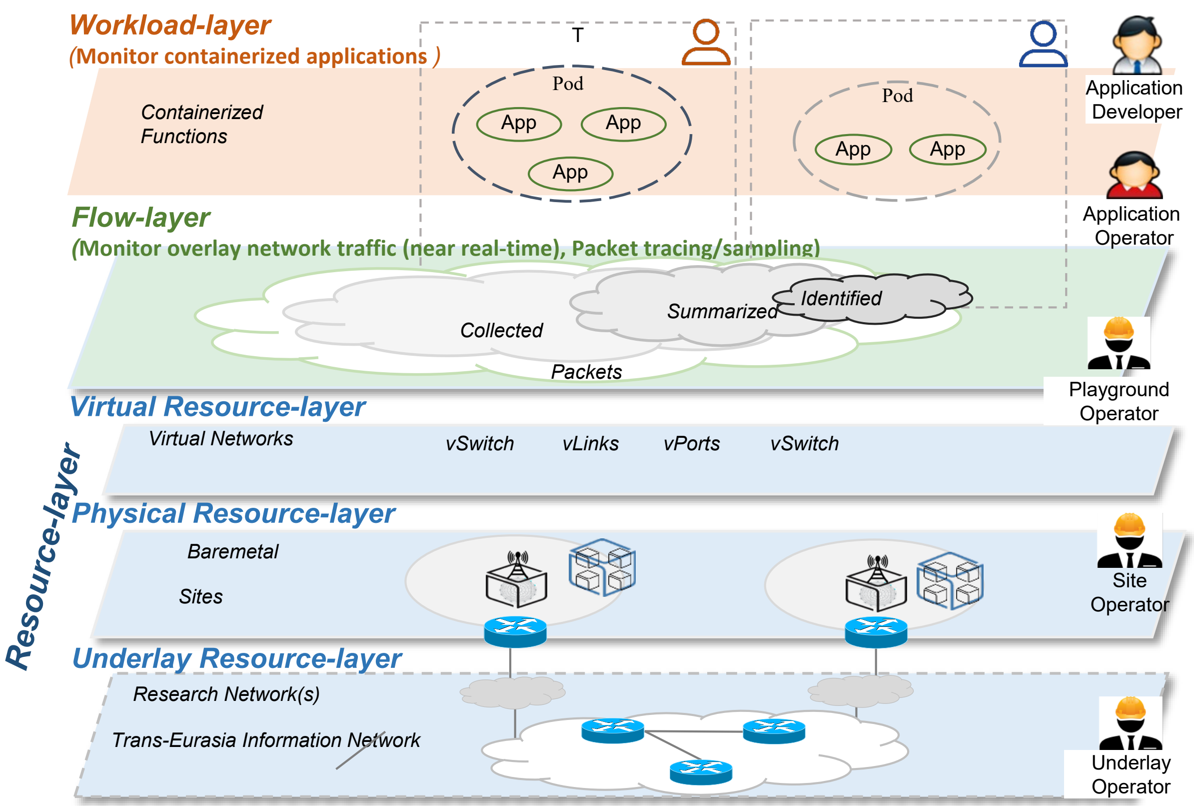
- Requirement 1
  - Edge computing leverages cloud-native applications to process/analyze some of their **computing/storage close to the location where the data is generated** by a large number of heterogeneous devices
- Requirement 2
  - Edge Computing leveraged the benefits of cloud-native by keeping what is great about the **cloud-native model (Microservices architectures, Containerized applications, and dynamically orchestrated)**, Yet applying it in the harsh physical and environment of IoT devices.
- Requirement 3
  - **Physical clusters** in an edge-cloud infrastructure **should be reconfigurable**, which can properly configure **baremetal and virtualized (p+v) resources on demand** for developers to easily acquire their dedicated cloud-native testbeds

- Issues

In a distributed environment, these resources/applications have **varying running conditions** depending on resource availability, quality of network connection, and being geo-distributed.

Identifying efficient methods for monitoring operational environment with continuity in order to maintain and sustain services

Maintaining persistence SmartX Multi-View Visibility

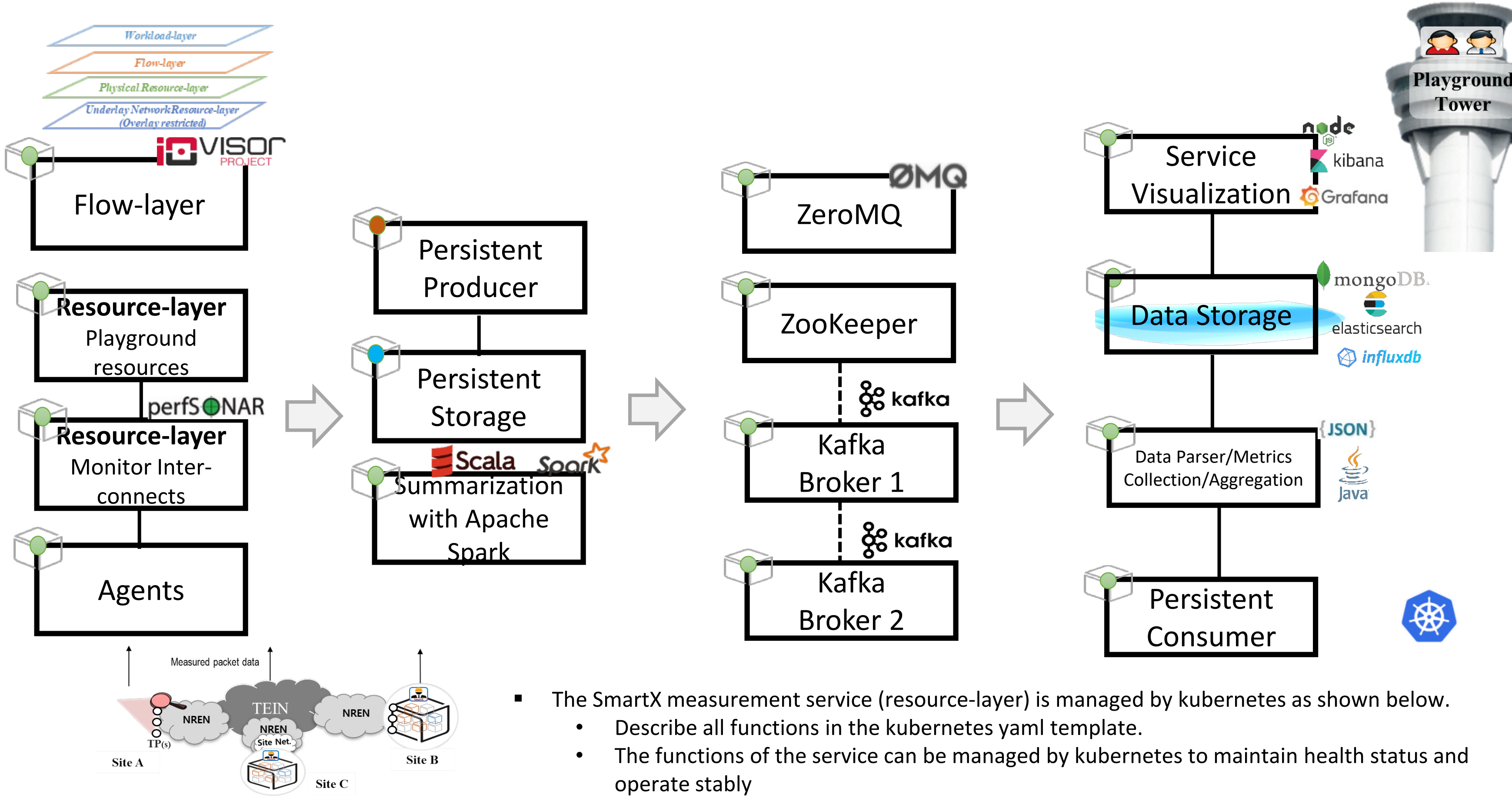


1. Setup Cloud-native Edge Boxes
   - OF@TEIN+ playground with cloud-native edge supported Micro-Boxes
2. Resilient measurement
   - Visibility Data and Applications are recoverable under failure.
3. Efficient and timely measurement
   - Low network Bandwidth and
   - Low storage consumption
4. Verification:
   - Multiple Visualization schemes

Multi-layer visibility

(underlay, physical, and virtual resource-layers, flow-layer, and workload-layer) solution denoted as **SmartX multi-view visibility framework (MVF)**

Design for Maintaining Persistent Playground Visibility

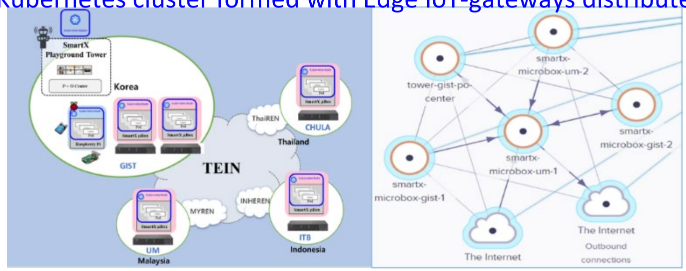


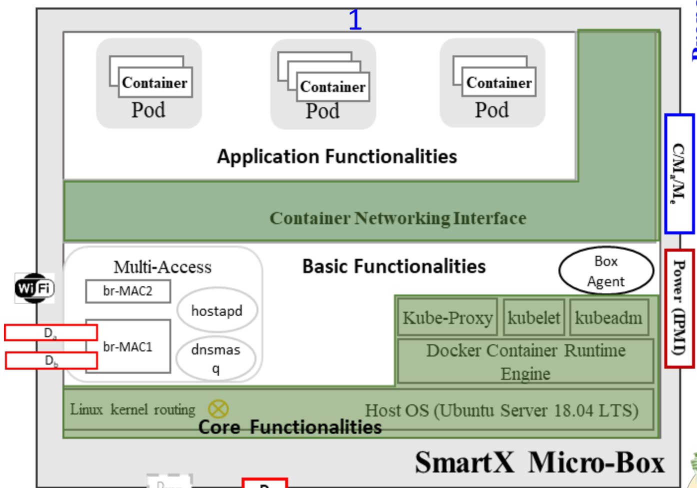
- The SmartX measurement service (resource-layer) is managed by kubernetes as shown below.
  - Describe all functions in the kubernetes yaml template.
  - The functions of the service can be managed by kubernetes to maintain health status and operate stably

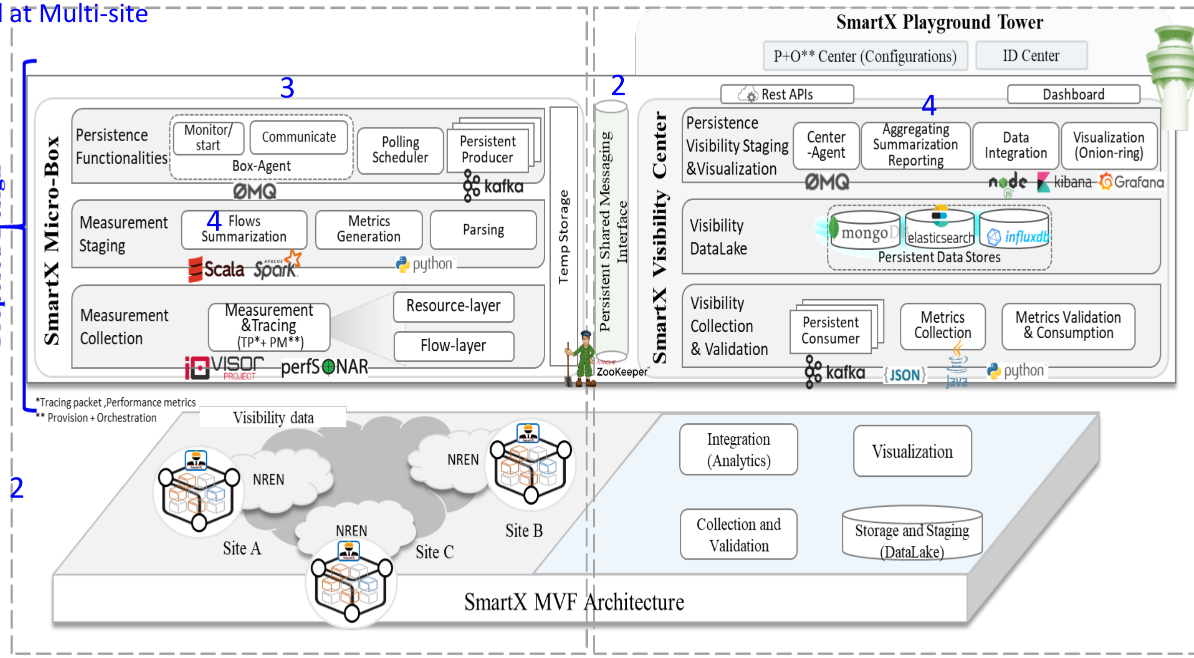
• Design of Cloud-native Edge Boxes

Maintaining Persistent Playground Operations (Visibility) for Service Developers and Operators

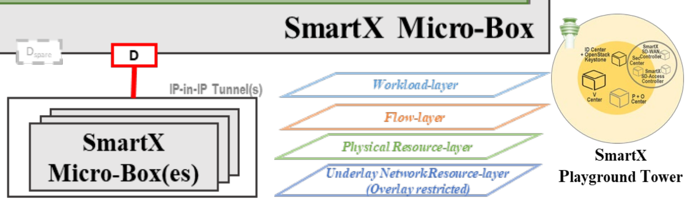
Kubernetes cluster formed with Edge IoT-gateways distributed at Multi-site







Design of SmartX Micro-Box



1. **Setup Cloud-native Edge Boxes**
2. **Resilient measurement**
   i. **Multi Agent &**
   ii. **Shared Messaging Interface**
3. **Efficient and timely measurement**
   i. **Flows Summarization**
   ii. **Optimal polling scheme**
4. **Verification: Visualization through**
   i. **Onion-ring visualization,**
   ii. **Time-series based visualization, Historical summarized report**

3rd Collaboration Community Meeting, 8 Sep 2020, Thailand-Laos-Korea

Pre-Processing For Maintaining Persistent Visibility

Raw Visibility Data for cpu/disk

[{"values":[0,0], "dstypes":["derive","derive"], "dsnames":["io_time","weighted_io_time"], "time":1589692165.948, "interval":10.000, "host":"smartx-microbox-gist-2", "plugin":"disk", "plugin_instance":"loop0", "type":"disk_io_time", "type_instance":""}]

[{"values":[63.9020263657957], "dstypes":["derive"], "dsnames":["value"], "time":1589692165.948, "interval":10.000, "host":"smartx-microbox-gist-2", "plugin":"cpu", "plugin_instance":"1", "type":"cpu", "type_instance":"idle"}]

[{"values":[0,0.30004865136956], "dstypes":["derive","derive"], "dsnames":["read","write"], "time":1589692165.948, "interval":10.000, "host":"smartx-microbox-gist-2", "plugin":"disk", "plugin_instance":"nvme0n1", "type":"disk_time", "type_instance":""}]

Packet header Tags

Measurement_boxname	smartx-microbox-gist-2
src_host	xxx.xx.xxx.xx
dest_host	xxx.xx.xxx.xx
src_host_port	52034
dest_host_port	9092
protocol	6
net_plane	0

Flows summarization Tags

protocol_count	757
min_tcp_window_size	2425
max_tcp_window_size	2426
avg_tcp_window_size	2426
std_dev_tcp_window_size	0.04
min_data_bytes	67
max_data_bytes	11636
avg_databytes	1510.18
std_dev_databytes	985.06
total_data_bytes	1143203
flow_duration	290.016
Processing_time	2020-01-14T11:00:31.136+09:00

IOVISOR Daily Collection

Date	Total Expected Collection	microbox-gist-1	microbox-gist-2	microbox-um-1	microbox-um-2	microbox-chula-1	microbox-itc-1	microbox-vnu-1
2019/09/20	288	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

Liveliness of Box with Visibility Center

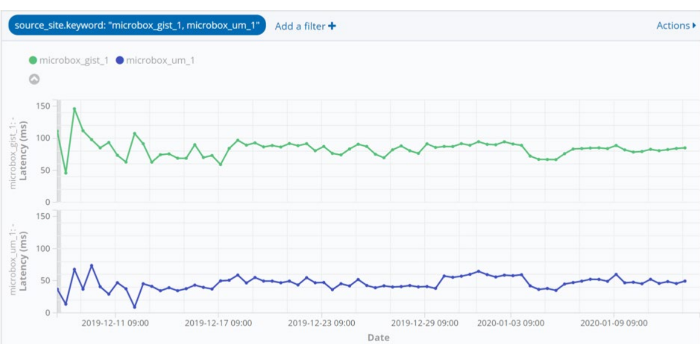
Date	Source	microbox-gist-1	microbox-gist-2	microbox-um-1	microbox-um-2	microbox-chula-1	microbox-itc-1	microbox-vnu-1
2019/09/20	Visibility_Center	100.0%	100.0%	97.92%	97.92%	96.53%	83.33%	98.61%

PING Generated Daily Collection

Date	Total Expected Collection	microbox-gist-1	microbox-gist-2	microbox-um-1	microbox-um-2	microbox-chula-1	microbox-itc-1	microbox-vnu-1
2019/09/20	144	144	144	144	144	144	144	144

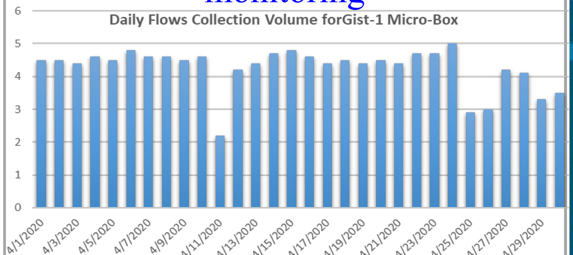
Daily Uptime(percentage) Report based on Ping

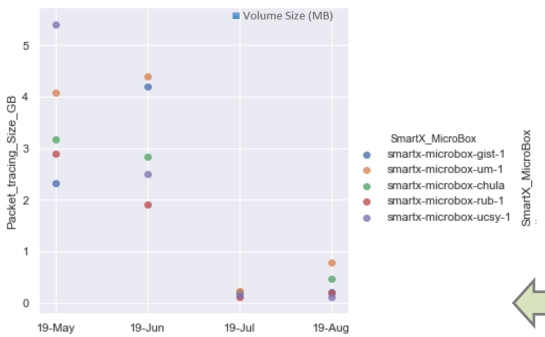
Date	Total Expected Collection	microbox-gist-1	microbox-gist-2	microbox-um-1	microbox-um-2	microbox-chula-1	microbox-itc-1	microbox-vnu-1
2019/09/20	microbox-gist-1	-	100.0%	98.61%	99.31%	97.92%	79.86%	98.61%

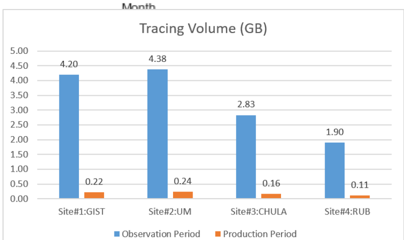


Maintaining SmartX Multi-View Visibility : Results of Implementation Use Cases

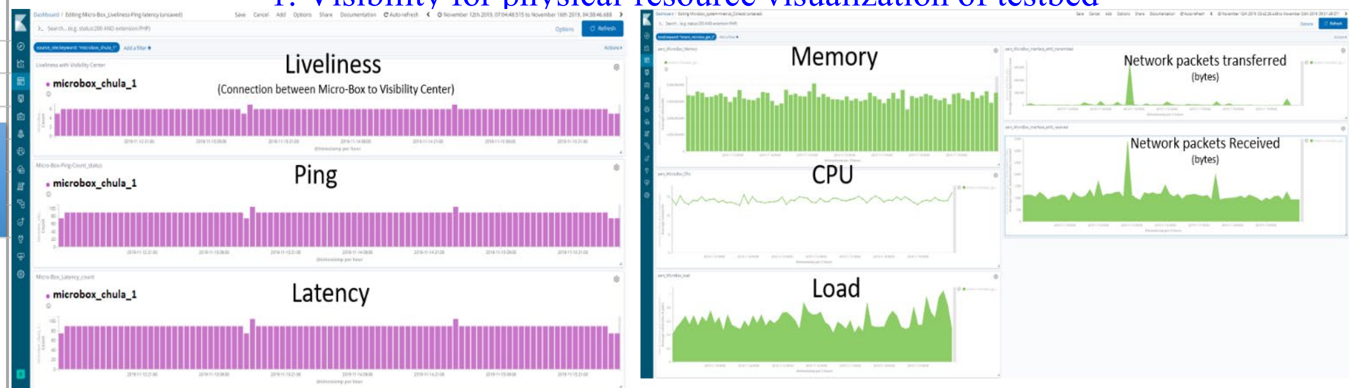
2. Visibility for resilient and efficient monitoring

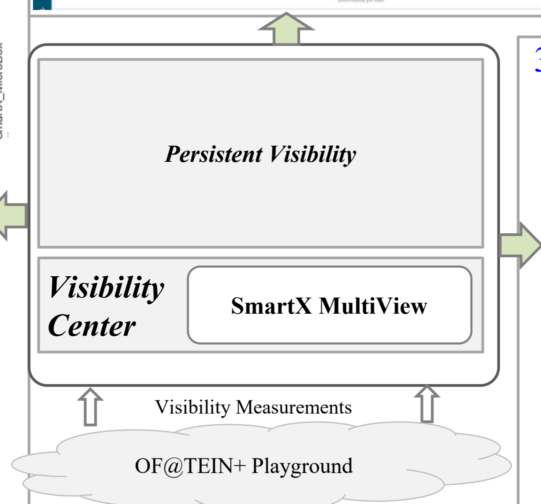




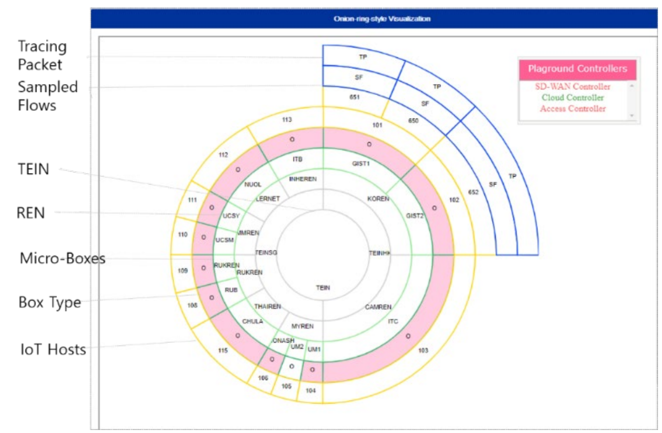


1. Visibility for physical-resource visualization of testbed

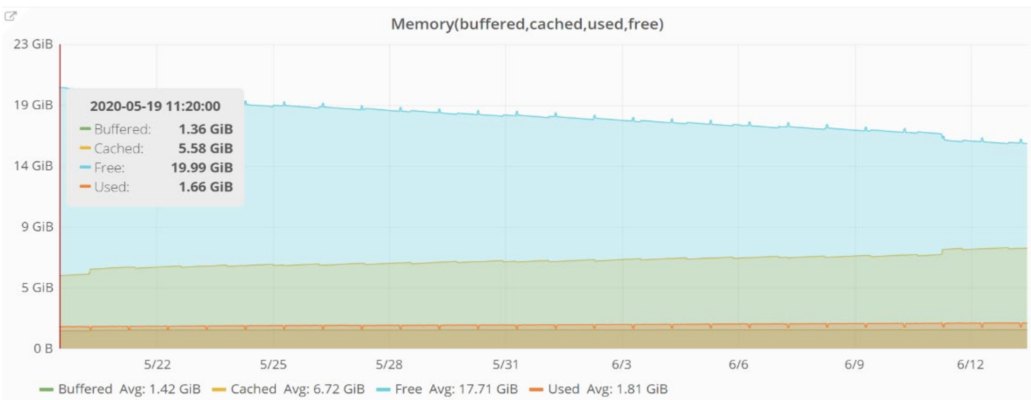




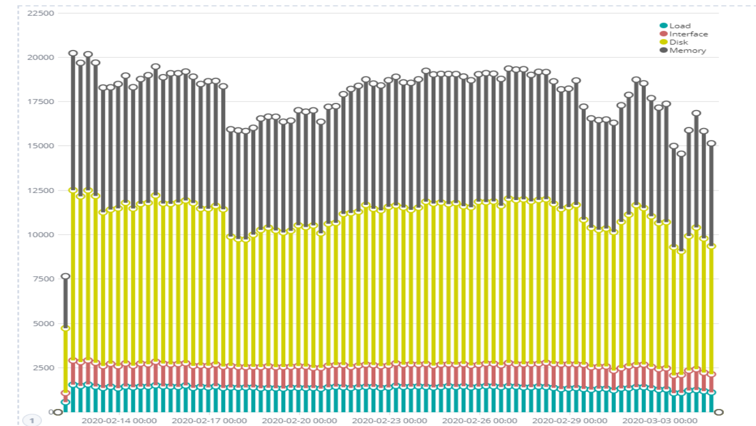
3. Visibility for Onion-ring visualization for OF@TEIN+



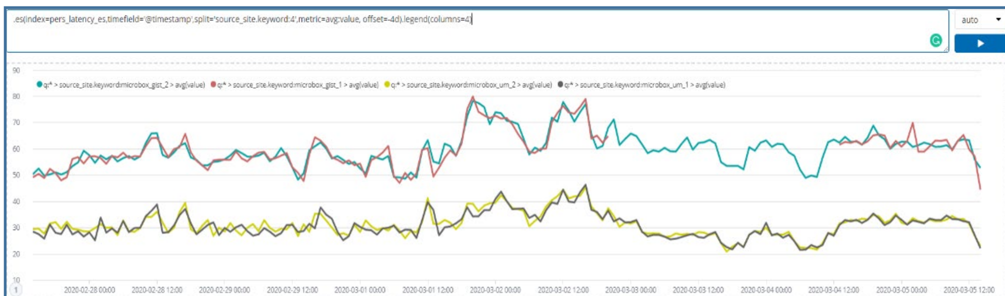
Spatio-temporal Summarized Visualization : Verification



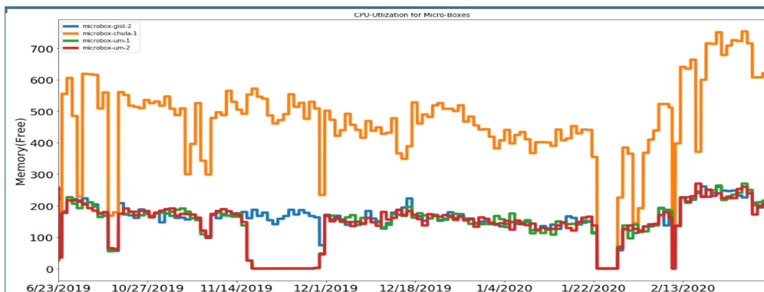
1. Results of Visualization for **single** metrics of system Utilization metrics (Memory) for a **single site**



2. Results of visualization for **multiple metrics** of physical resource layers from a **single site over** time-line

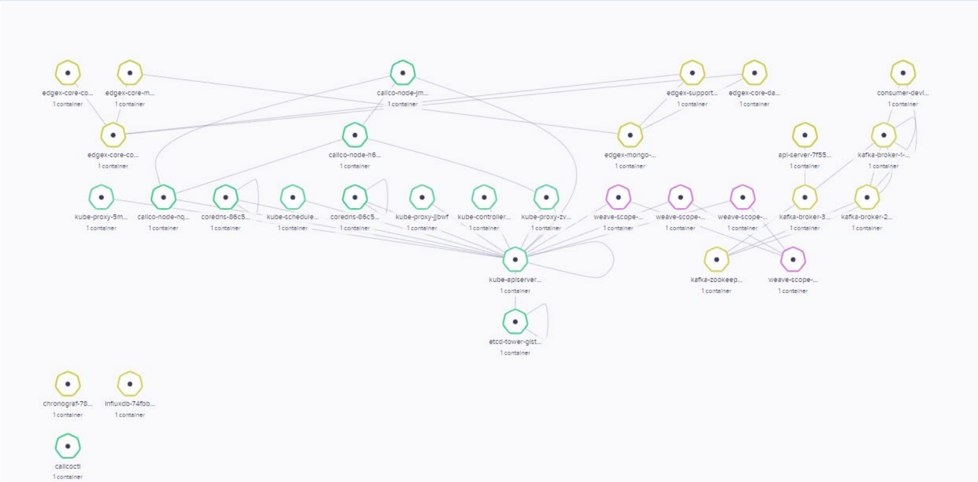


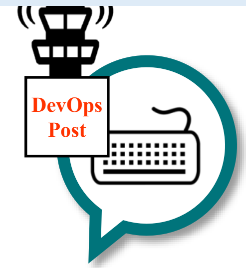
3. Result for comparison of visibility measurements from **multiple sites** over time-line



4. Results of **trends pattern** based visualization of memory utilization for **multiple sites** over time line

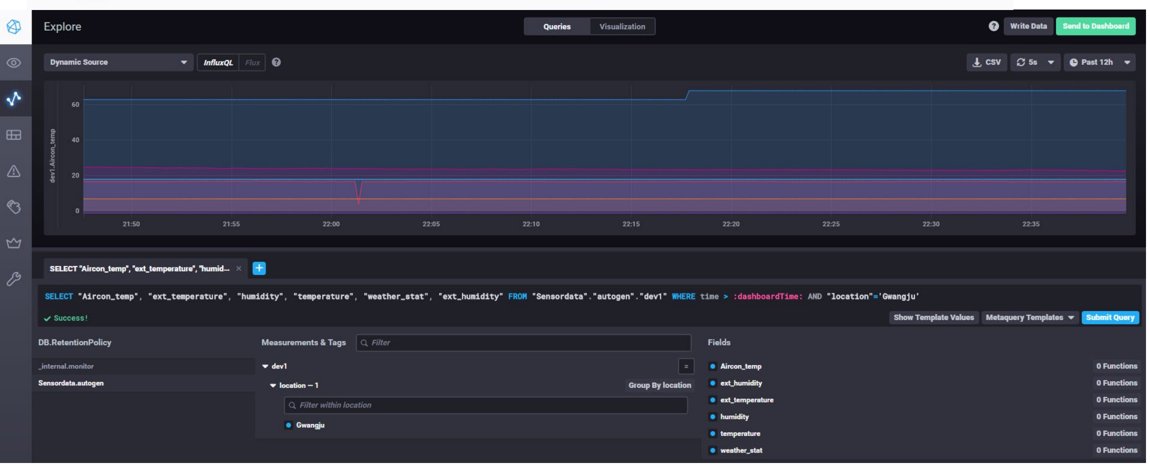
Verifying SmartX Containerized IoT-Cloud service with IoT Devices



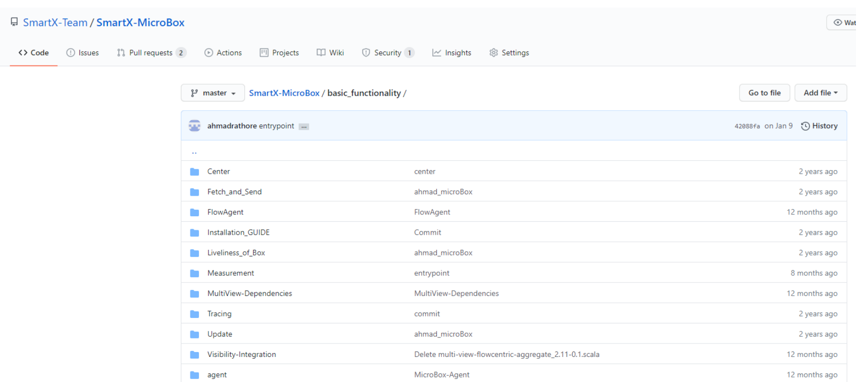


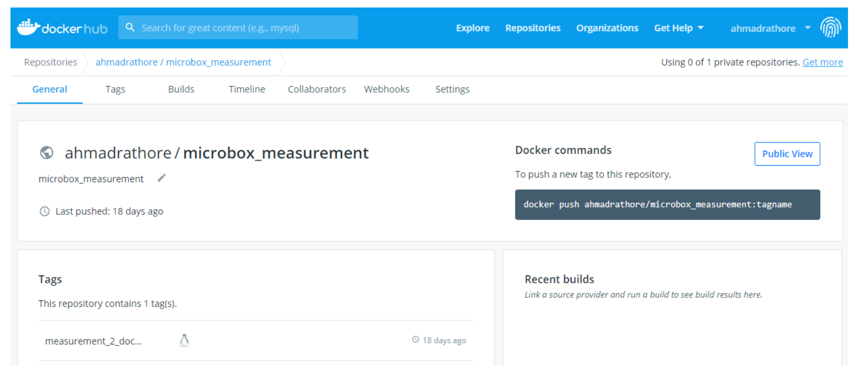
You can check the application service data visualization at the chronograf connected with influxdb

Please Access https://DevOps Post_IP:32016



Project Resources





https://github.com/SmartX-Team/SmartX-MicroBox/

Facebook Page

https://www.facebook.com/iotcloudserve/

Github

https://github.com/IoTcloudServe