

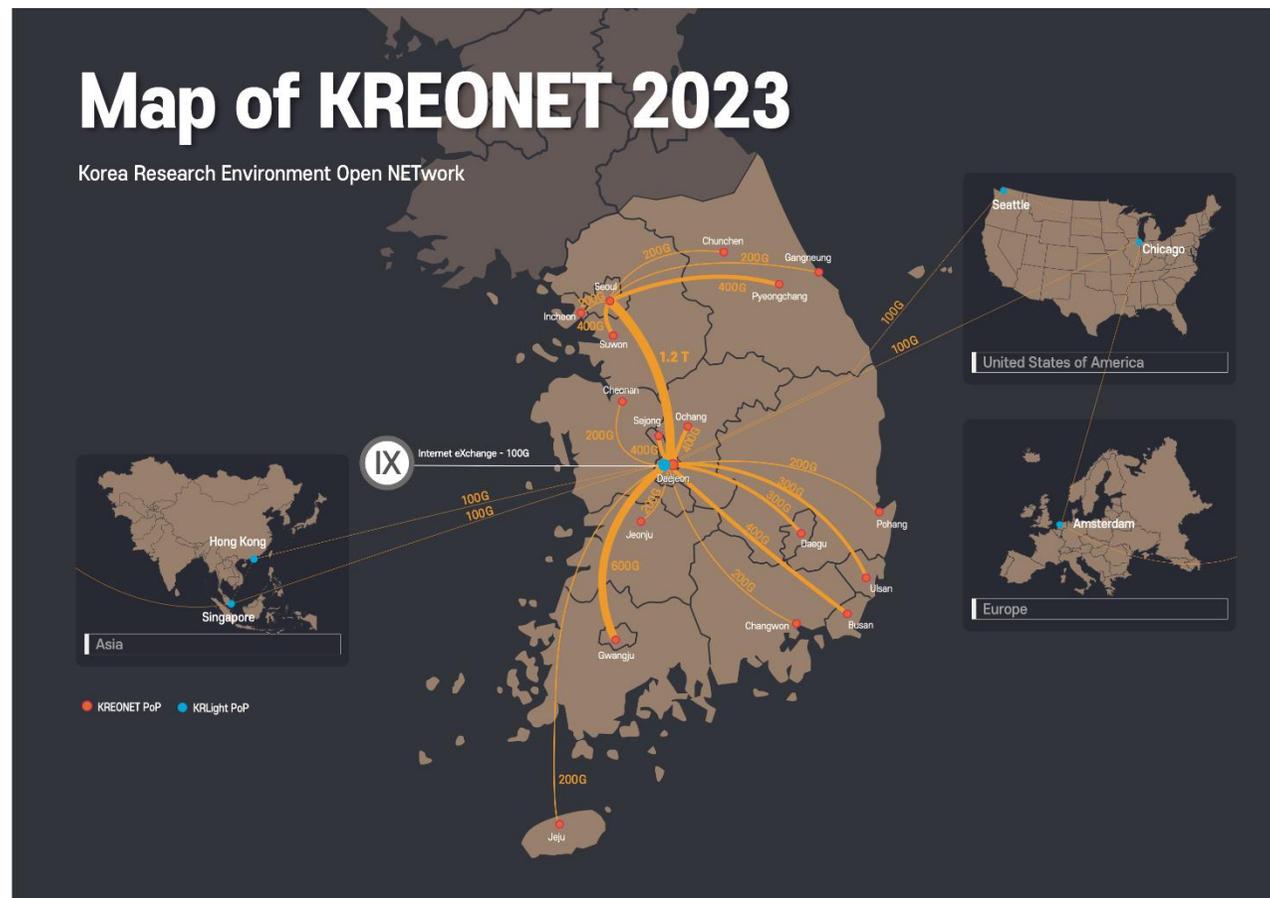
Electricity Power Management Utilizing AI in KREONET-S

Seongjin Park, KISTI

xkqpekdl@kisti.re.kr

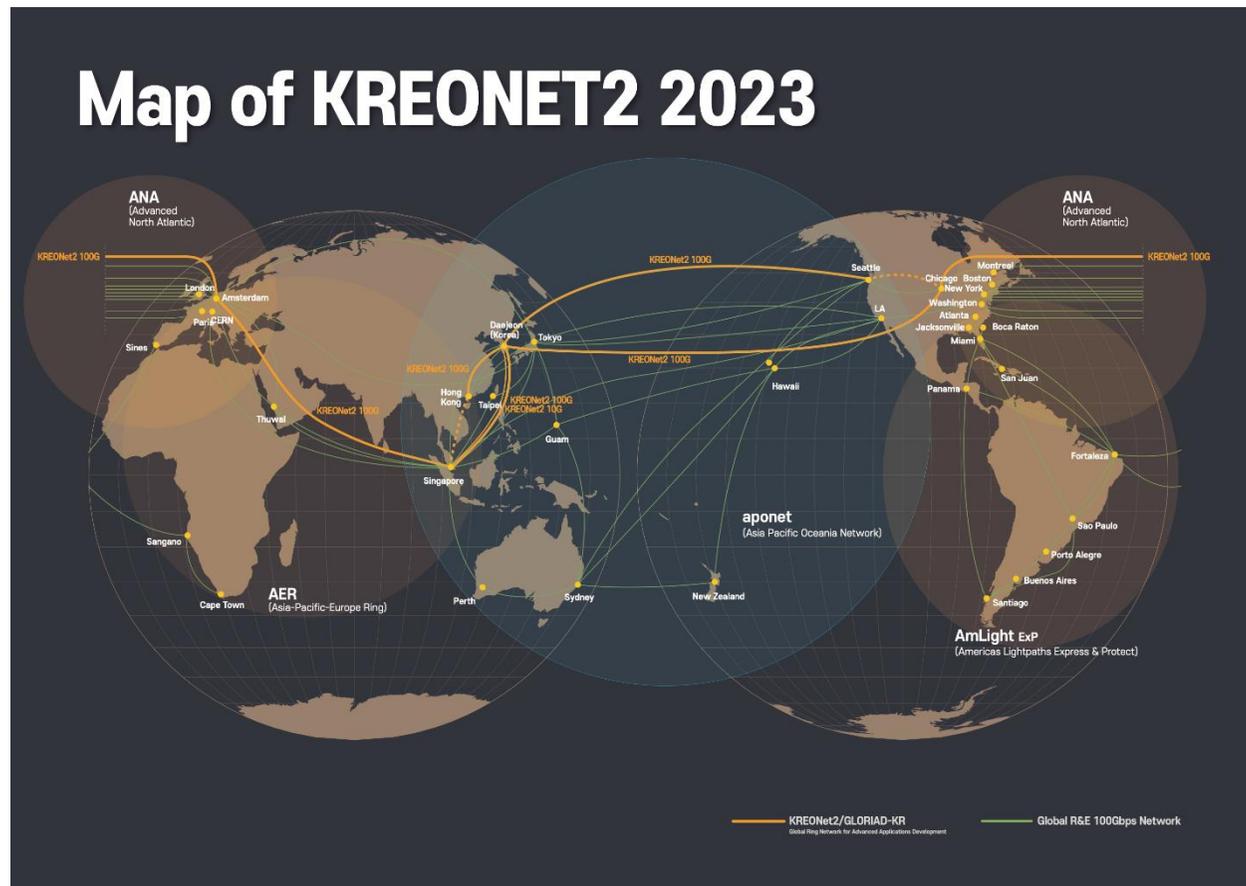
KREONET / KREONET2 Intro.

- KREONET network (domestic)
 - Ring topology (600Gbps per lambda)



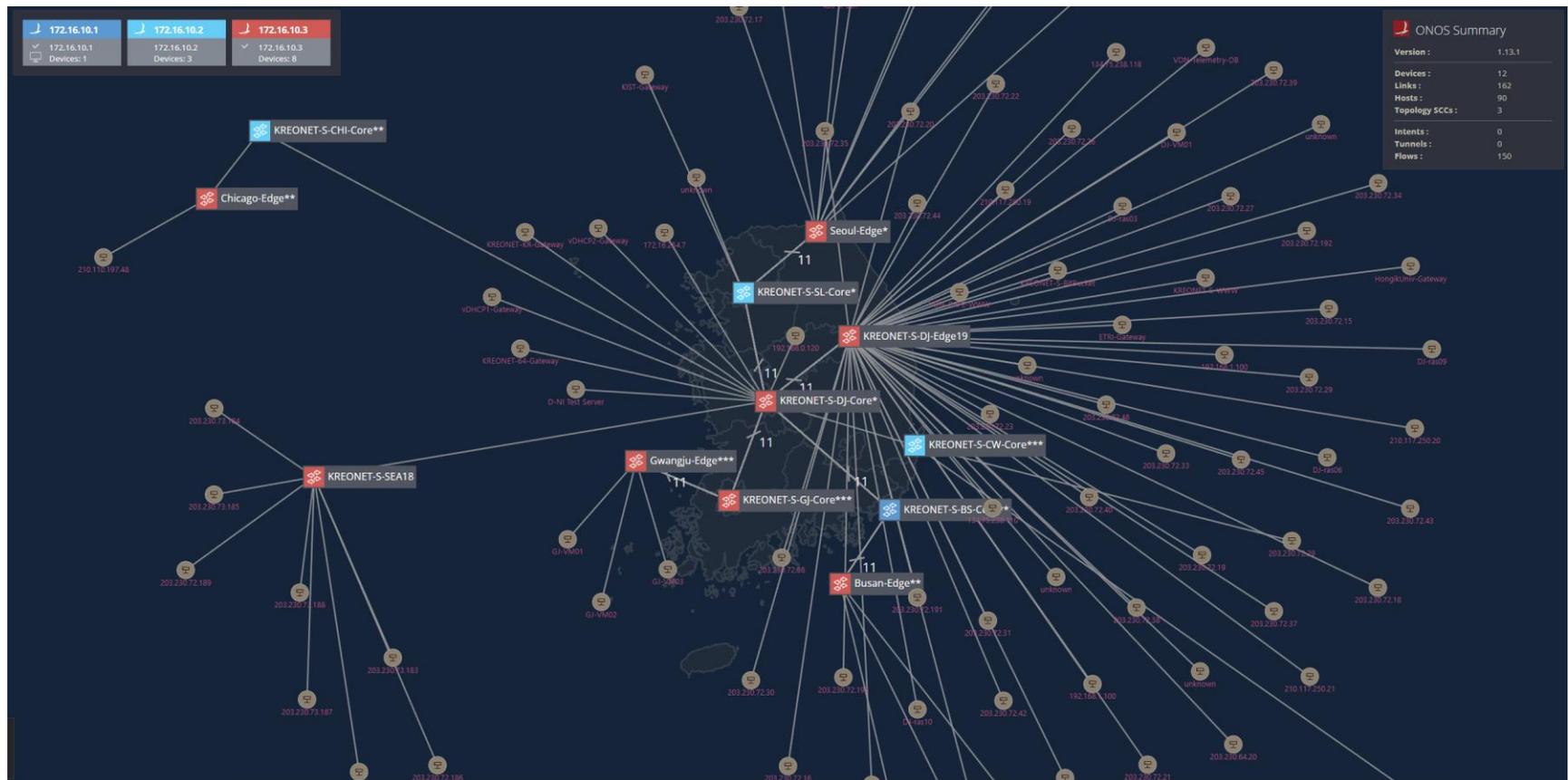
KREONET / KREONET2 Intro.

- KREONET2 network (international)
 - Ring topology (Update: Daejon - Singapore / Singapore - Amsterdam)



KREONET-S Intro.

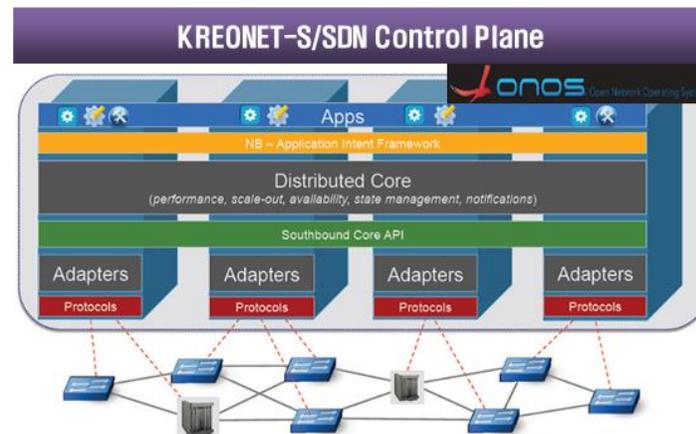
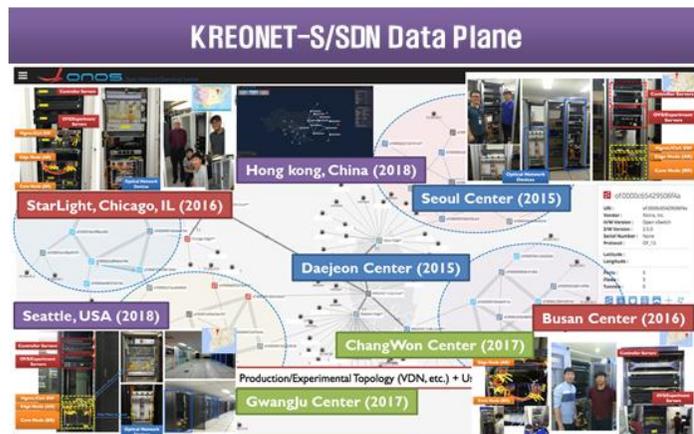
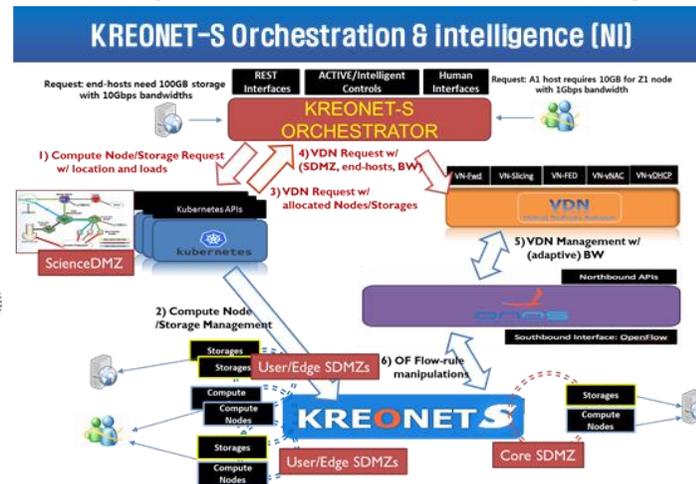
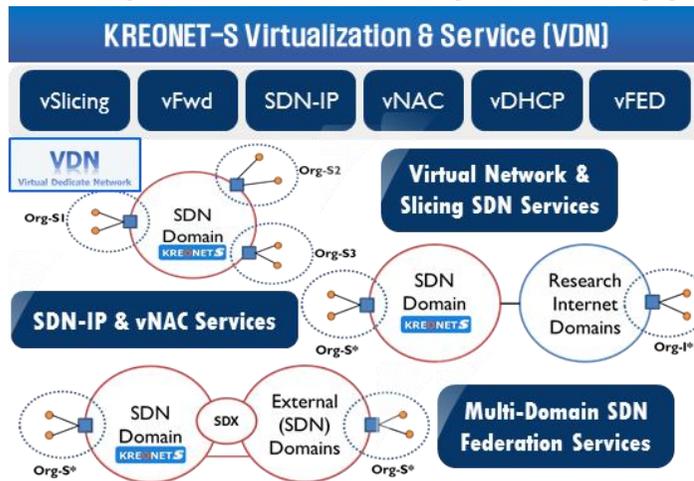
- KREONET-S topology
 - SDN switches in 7 Pops: Domestic(5) / International(2)



KREONET-S Intro.

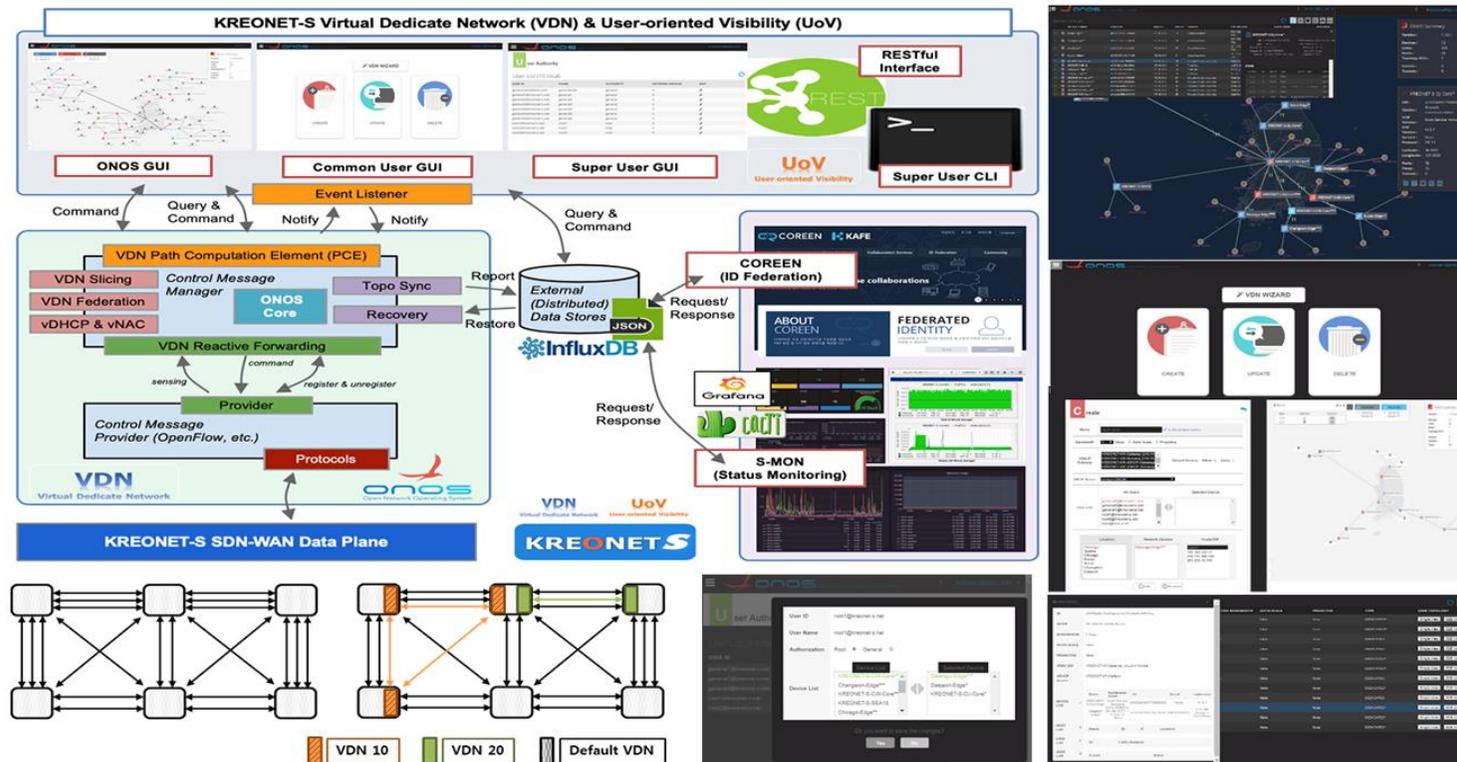
- KREONET-S architecture

- Data plane/ Control plane / Application plane / Orchestration plane



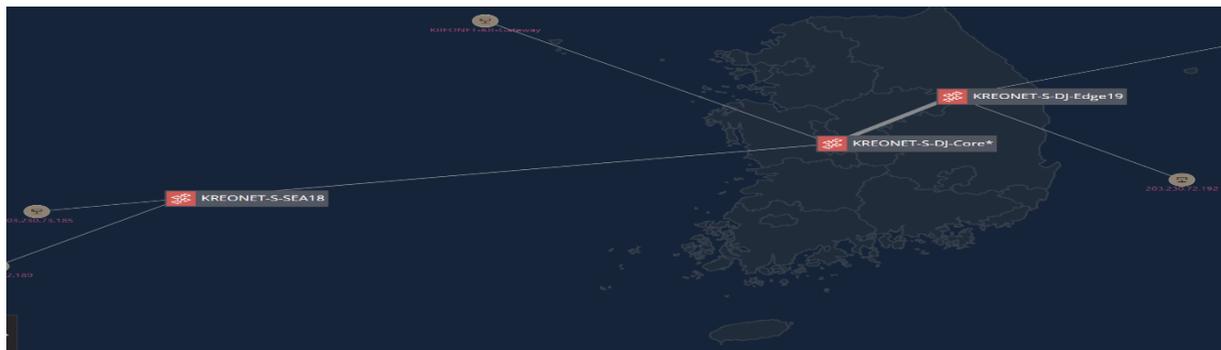
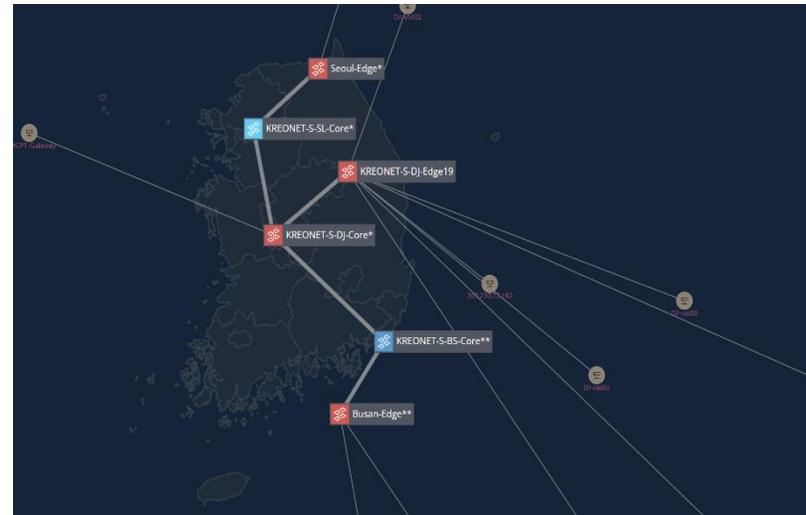
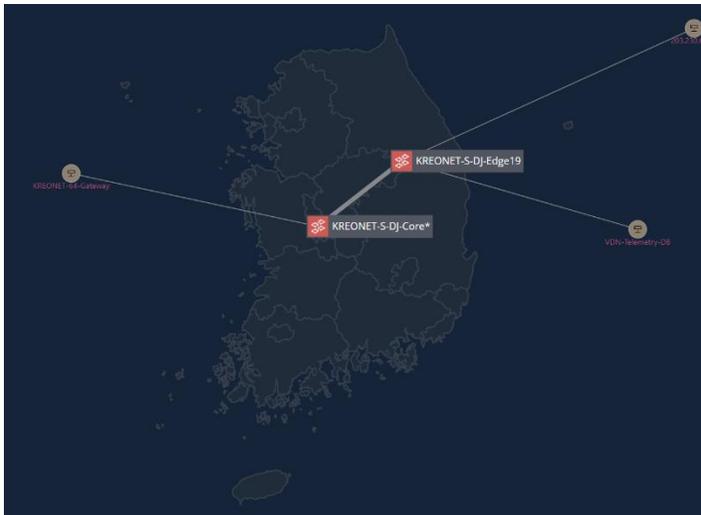
KREONET-S Intro.

- Virtually Dedicated Network (VDN)
 - User based Dynamic on-demand Virtual network managements



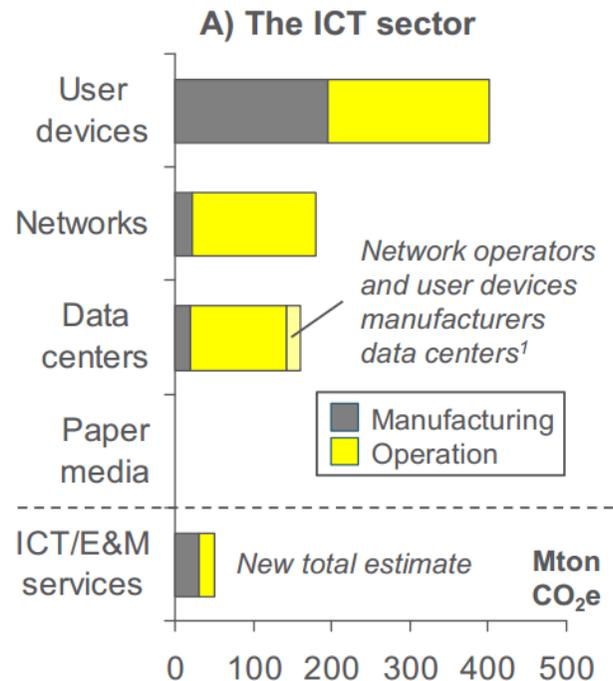
KREONET-S Intro.

- Virtually Dedicated Network (VDN)
 - Logically isolated and dedicated networking with high performance



Electricity Power Management in KREONET-S

- Network intelligence in KREONET-S
 - Orchestration plane
 - Carbon footprint of networks in the ICT sector
 - SDN-based electricity power management on network operation

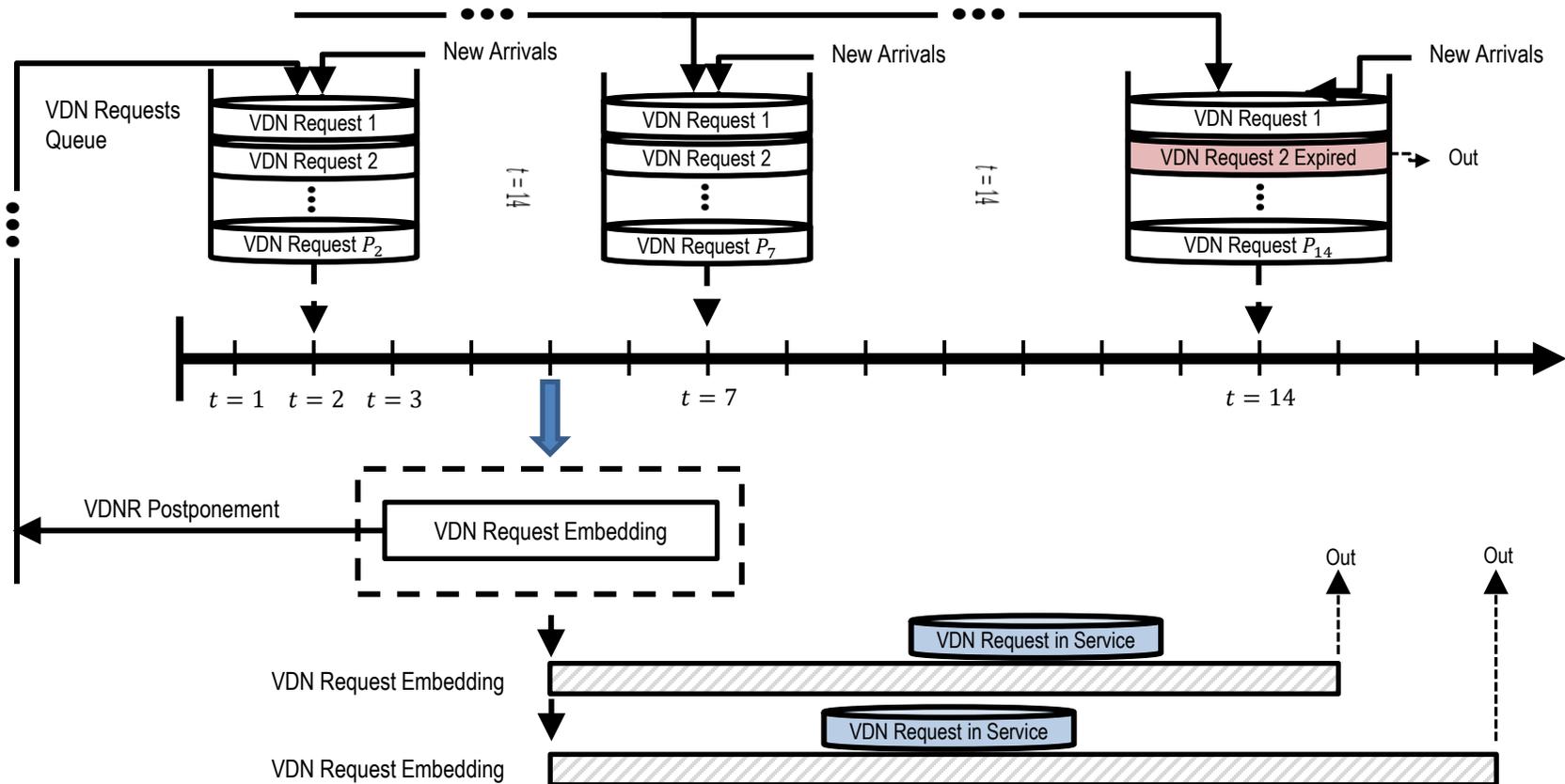


Electricity Power Management in KREONET-S

- VDN(Virtual Dedicated Network) requests
 - Set of hosts
 - Required bandwidth
 - Required service duration
 - Maximum queue delay
- Objective function
 - Minimize energy consumption in KREONET-S (energy efficiency)
 - Satisfying VDN requests (resource utilization)

Electricity Power Management in KREONET-S

- VDN(Virtual Dedicated Network) embedding
 - VDN requests, embedding, and postponement



Electricity Power Management in KREONET-S

- Power consumption model for OpenFlow switches
 - $P_{switch} = P_{base} + P_{config} + P_{control} + P_{OF}$
 - Power consumption of configuration
 - Number of active ports
 - Configured line speed
 - Power consumption of control traffic
 - PacketIn messages
 - FlowMod messages
 - Power consumption of the processed OpenFlow traffic
 - Per each processed packet
 - Number of matches
 - Number of actions
 - Almost independent of the traffic: (P_{OF})

Electricity Power Management in KREONET-S

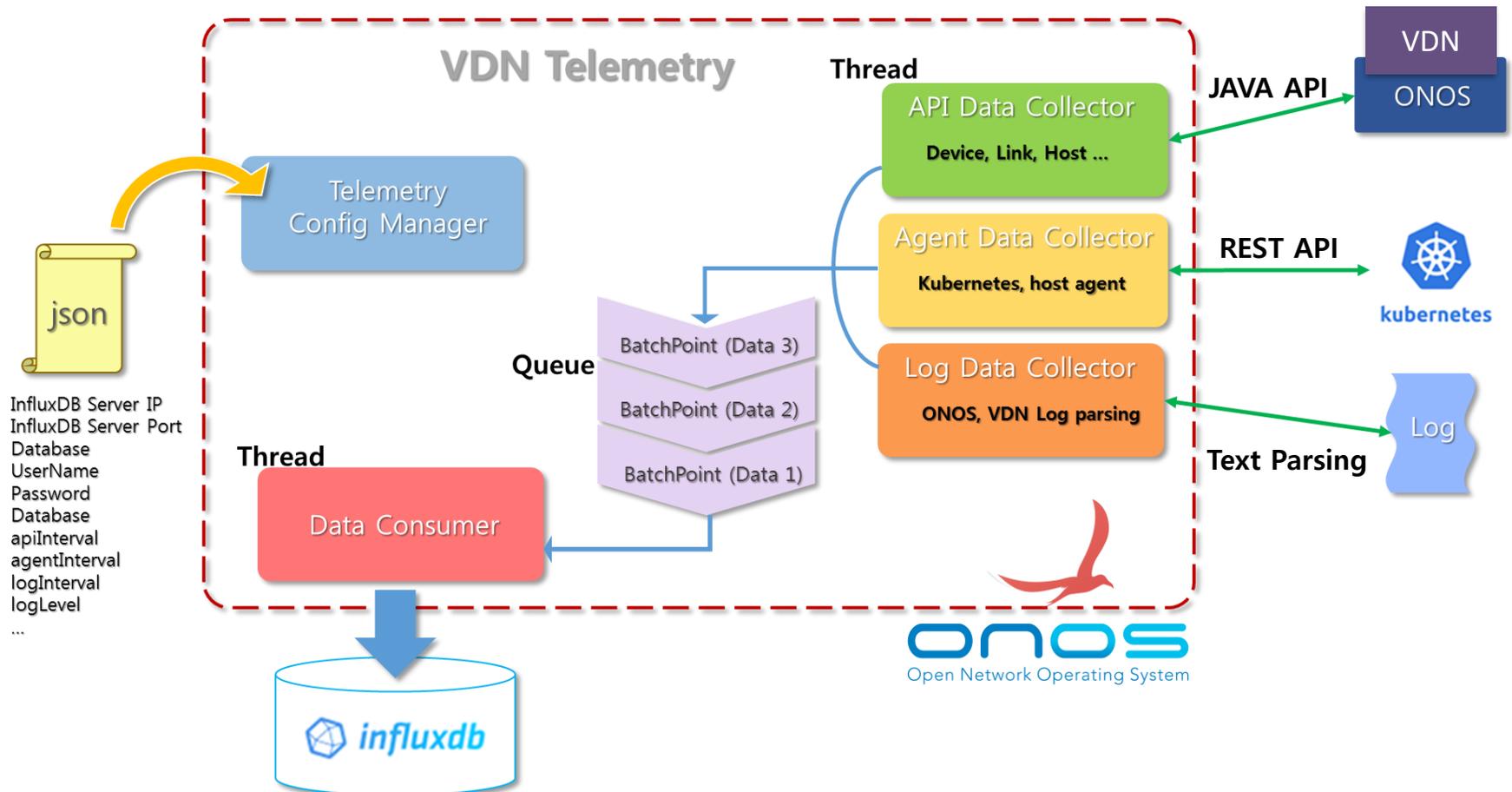
- SDN-based switches in KREONET-S
 - Specifications of Switches
 - 6 Brocade Communications (MLXe-8)
 - Max: 4,060W
 - 3 Arista (DCS-7050SX-64)
 - Typical(50% load): 140W
 - Max: 220W
 - 3 Edge-Core networks (as5712-54x)
 - Max (without pluggable optics): 282W

- References

- Charalampou, P. & Sykas, E. D. An SDN Focused Approach for Energy Aware Traffic Engineering in Data Centers. *Sensors Basel Switz* **19**, 3980 (2019).
- Assefa, B. G. & Özkasap, Ö. A survey of energy efficiency in SDN: Software-based methods and optimization models. *J Netw Comput Appl* **137**, 127–143 (2019).
- Assefa, B. G. & Özkasap, Ö. RESDN: A Novel Metric and Method for Energy Efficient Routing in Software Defined Networks. *Ieee Trans Netw Serv* **17**, 736–749 (2020).
- Etengu, R., et. Al., AI-Assisted Framework for Green-Routing and Load Balancing in Hybrid Software-Defined Networking: Proposal, Challenges and Future Perspective. *Ieee Access* **8**, 166384–166441 (2020).

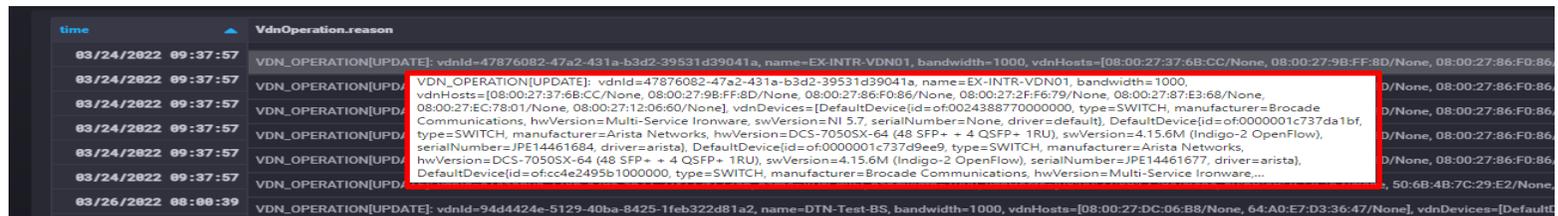
Data acquisition in KREONET-S

- KREONET-S telemetry



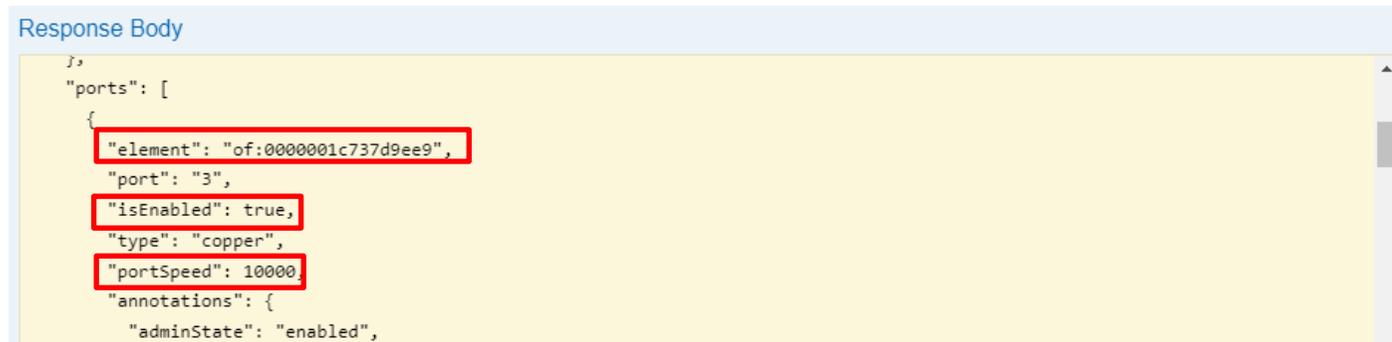
Data acquisition in KREONET-S

- KREONET-S telemetry
 - VDN operation (InfluxDB)
 - VDN ID, Bandwidth, VDN hosts, Devices, Links(port), etc.



```
time VdnOperation_reason
03/24/2022 09:37:57 VDN_OPERATION[UPDATE]: vdnId=47876082-47a2-431a-b3d2-39531d39041a, name=EX-INTR-VDN01, bandwidth=1000, vdnHosts=[08:00:27:37:6B:CC/None, 08:00:27:9B:FF:8D/None, 08:00:27:86:F0:86:
03/24/2022 09:37:57 VDN_OPERATION[UPDATE]: vdnId=47876082-47a2-431a-b3d2-39531d39041a, name=EX-INTR-VDN01, bandwidth=1000,
vdnHosts=[08:00:27:37:6B:CC/None, 08:00:27:9B:FF:8D/None, 08:00:27:86:F0:86/None, 08:00:27:2F:F6:79/None, 08:00:27:87:E3:68/None,
08:00:27:EC:78:01/None, 08:00:27:12:06:60/None], vdnDevices=[DefaultDevice[id=of00024388770000000, type=SWITCH, manufacturer=Brocade
Communications, hwVersion=Multi-Service Ironware, swVersion=NI 5.7, serialNumber=None, driver=default], DefaultDevice[id=of0000001c737da1bf,
type=SWITCH, manufacturer=Arista Networks, hwVersion=DCS-7050SX-64 (48 SFP+ + 4 QSFP+ 1RU), swVersion=4.15.6M (Indigo-2 OpenFlow),
serialNumber=JPE14461684, driver=arista), DefaultDevice[id=of0000001c737d9ee9, type=SWITCH, manufacturer=Arista Networks,
hwVersion=DCS-7050SX-64 (48 SFP+ + 4 QSFP+ 1RU), swVersion=4.15.6M (Indigo-2 OpenFlow), serialNumber=JPE14461677, driver=arista),
DefaultDevice[id=ofc4e2495b1000000, type=SWITCH, manufacturer=Brocade Communications, hwVersion=Multi-Service Ironware,
03/24/2022 09:37:57 VDN_OPERATION[UPDATE]: vdnId=47876082-47a2-431a-b3d2-39531d39041a, name=EX-INTR-VDN01, bandwidth=1000, vdnHosts=[08:00:27:37:6B:CC/None, 08:00:27:9B:FF:8D/None, 08:00:27:86:F0:86:
03/24/2022 09:37:57 VDN_OPERATION[UPDATE]: vdnId=47876082-47a2-431a-b3d2-39531d39041a, name=EX-INTR-VDN01, bandwidth=1000,
vdnHosts=[08:00:27:37:6B:CC/None, 08:00:27:9B:FF:8D/None, 08:00:27:86:F0:86/None, 08:00:27:2F:F6:79/None, 08:00:27:87:E3:68/None,
08:00:27:EC:78:01/None, 08:00:27:12:06:60/None], vdnDevices=[DefaultDevice[id=of00024388770000000, type=SWITCH, manufacturer=Brocade
Communications, hwVersion=Multi-Service Ironware, swVersion=NI 5.7, serialNumber=None, driver=default], DefaultDevice[id=of0000001c737da1bf,
type=SWITCH, manufacturer=Arista Networks, hwVersion=DCS-7050SX-64 (48 SFP+ + 4 QSFP+ 1RU), swVersion=4.15.6M (Indigo-2 OpenFlow),
serialNumber=JPE14461684, driver=arista), DefaultDevice[id=of0000001c737d9ee9, type=SWITCH, manufacturer=Arista Networks,
hwVersion=DCS-7050SX-64 (48 SFP+ + 4 QSFP+ 1RU), swVersion=4.15.6M (Indigo-2 OpenFlow), serialNumber=JPE14461677, driver=arista),
DefaultDevice[id=ofc4e2495b1000000, type=SWITCH, manufacturer=Brocade Communications, hwVersion=Multi-Service Ironware,
03/24/2022 09:37:57 VDN_OPERATION[UPDATE]: vdnId=47876082-47a2-431a-b3d2-39531d39041a, name=EX-INTR-VDN01, bandwidth=1000, vdnHosts=[08:00:27:37:6B:CC/None, 08:00:27:9B:FF:8D/None, 08:00:27:86:F0:86:
03/26/2022 08:08:39 VDN_OPERATION[UPDATE]: vdnId=94d4424e-5129-40ba-8425-1feb322d81a2, name=DTN-Test-BS, bandwidth=1000, vdnHosts=[08:00:27:DC:06:BB/None, 64:A0:E7:D3:36:47/None], vdnDevices=[Default
```

- Port states per each device information (REST API)
 - Port ID, isEnabled, port Speed, etc.



```
Response Body
{
  "ports": [
    {
      "element": "of:0000001c737d9ee9",
      "port": "3",
      "isEnabled": true,
      "type": "copper",
      "portSpeed": 10000,
      "annotations": {
        "adminState": "enabled",
      }
    }
  ]
}
```

Data acquisition in KREONET-S

- KREONET-S telemetry
 - Flow states per each device
 - Flow ID, priority, packets, etc.

```
"id": "281477277096117",
"tableId": 0,
"appId": "org.onosproject.core",
"groupId": 0,
"priority": 40000,
"timeout": 0,
"isPermanent": true,
"deviceId": "of:cc4e249599000000",
"state": "ADDED",
"life": 26012679,
"packets": 3948,
```

- Processing packets (PacketIn+)

- srcMac, dstMac, device, srcPort, dstPort, etc.

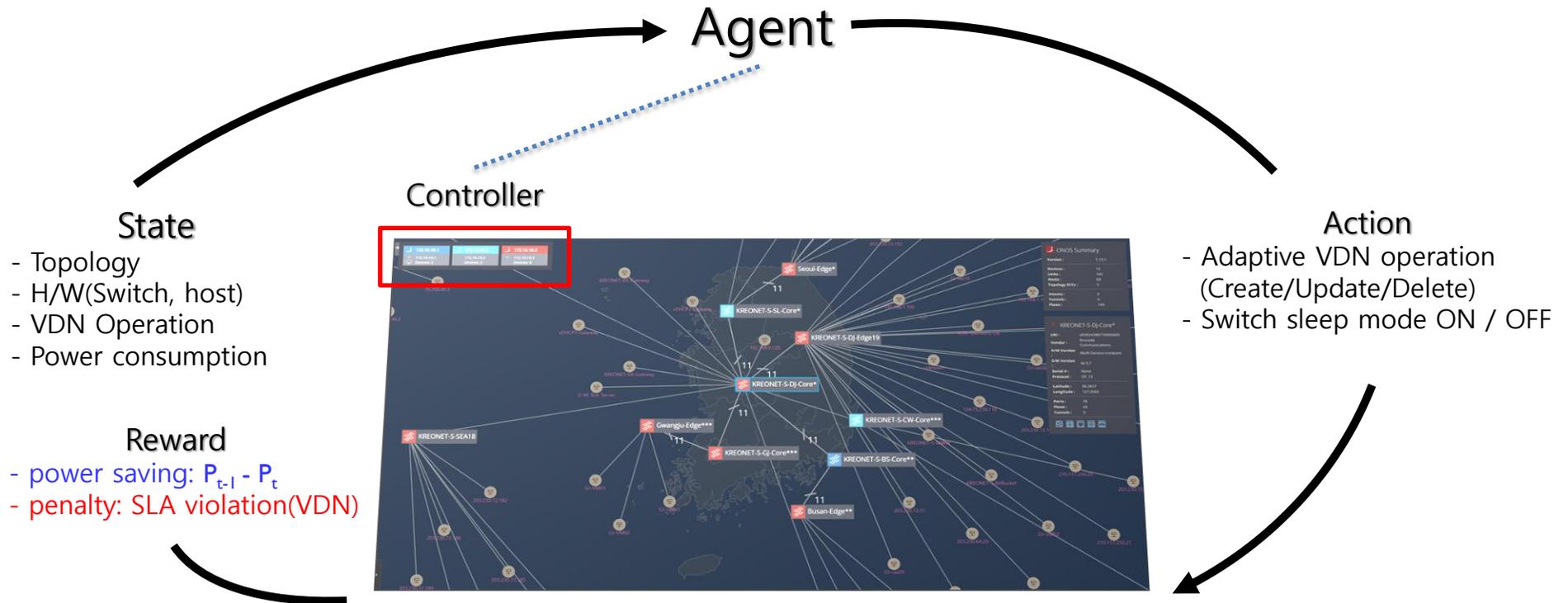
```
time PacketIn.reason
01/19/2022 12:19:29 PACKET_IN: [Ethernet] etherType=2048, vlanID=-1, srcMac=64:A0:E7:D3:36:47, dstMac=08:00:27:E0:C2:0D, device=of:0024388770000000, srcPort=101, dstPort=null, [IPv4] identification=23743, flags=2, ttl=63, protocol=6, srcAddress=210.110.196.42, dstAddress=203.230.64.11, [TCP] srcPort=56960, dstPort=8086, flags=2)
01/19/2022 12:19:29 PACKET_IN: [Ethernet] etherType=2048, vlanID=-1, srcMac=64:A0:E7:D3:36:47, dstMac=08:00:27:E0:C2:0D, device=of:0024388770000000, srcPort=101, dstPort=null, [IPv4] identification=23743, flags=2, ttl=63, protocol=6, srcAddress=210.110.196.42, dstAddress=203.230.64.11, [TCP] srcPort=56960, dstPort=8086, flags=2)
01/19/2022 12:19:29 PACKET_IN: [Ethernet] etherType=2048, vlanID=-1, srcMac=64:A0:E7:D3:36:47, dstMac=08:00:27:E0:C2:0D, device=of:0024388770000000, srcPort=101, dstPort=null, [IPv4] identification=23743, flags=2, ttl=63, protocol=6, srcAddress=210.110.196.42, dstAddress=203.230.64.11, [TCP] srcPort=56960, dstPort=8086, flags=2)
01/19/2022 12:19:29 PACKET_IN: [Ethernet] etherType=2048, vlanID=-1, srcMac=64:A0:E7:D3:36:47, dstMac=08:00:27:E0:C2:0D, device=of:0024388770000000, srcPort=101, dstPort=null, [IPv4] identification=4164, flags=2, ttl=63, protocol=6, srcAddress=210.110.196.41, dstAddress=203.230.64.11, [TCP] srcPort=33754, dstPort=8086, flags=2)
01/19/2022 12:19:29 PACKET_IN: [Ethernet] etherType=50, vlanID=-1, srcMac=44:4C:A8:B6:49:32, dstMac=01:00:0C:CC:CC:CD, device=of:0000001c737da1bf, srcPort=31, dstPort=null
```

AI algorithm in KREONET-S

- Agile procedure
 - Phase 1. AI based power management model design
 - MDP model definition
 - Applying simple network environment & RL algorithm
 - Phase 2. evaluation using hybrid testbed / simulation
 - Applying complex network environment & service scenario
 - Refinement AI algorithms
 - Designing and applying various AI & DRL algorithms

AI algorithm in KREONET-S

- Reinforcement Learning (RL) based overall architecture



AI algorithm in KREONET-S

- Proposed algorithm in simple environment
 - MDP definition
 - State
 - Port status of switches by time (energy consumption)
 - Status of VDN operation by time
 - Action : adjustable VDN operation (Discrete / Continuous)
 - Create:VDN list X (bandwidth / set of hosts)
 - Update:VDN list X (bandwidth / set of hosts)
 - Delete:VDN list
 - Switch on/off
 - Reward
 - Total electricity power saving by time: $P_{t-1} - P_t$ (P: energy consumption)
 - Penalty(soft): $BW_{required} - BW_{allocated} < 0$ (SLA violation)
 - Penalty(hard): $PT_{used} - PT_{allocated} < 0$ (threshold) (PT: Peak Traffic)
 - Development environment (AI algorithm)
 - Tensorflow / Keras(deep learning model) / gym (RL environment)
 - Create custom gym environment

AI algorithm in KREONET-S

- VDN generation using REST API
 - VDN create / delete / update

C reate

Name: Is the proper name. Upload

Bandwidth: Gbps Auto Scale Proactive

SDN-IP Gateway: Default Access: Allow Deny

DHCP Server:

User List: All Users Selected User(s)

beta

general1@kreonet-s.net

laurosilva

tien

Chanjin_Park

mirr@kisti.re.kr

Location

Network Devices

Hosts/GW

Gwangju
Seattle
Chicago
Seoul
Busan
Changwon
Daejeon

+ Add
- Remove

Active VDN Information

Location	Network Devices	Hosts/GW

U pdate

ID: 13b76a9c-f1c2-433d-a824-26ac8f0d5745

Name: Is the proper name. Upload

Bandwidth: Gbps Auto Scale Proactive

SDN-IP Gateway: Default Access: Allow Deny

DHCP Server:

User List: All Users Selected User(s)

beta

general1@kreonet-s.net

laurosilva

tien

Chanjin_Park

mirr@kisti.re.kr

yh.kim086@kisti.re.kr

Location

Network Devices

Hosts/GW

Gwangju
Seattle
Chicago
Seoul
Busan
Changwon
Daejeon

+ Add
- Remove

Active VDN Information

Location	Network Devices	Hosts/GW
Seoul	Seoul-Edge*	A0:36:9F:C5:52:A2
Seoul	Seoul-Edge*	203.230.72.176
Daejeon	KREONET-S-DJ-Edge19	134.75.238.118
Daejeon	KREONET-S-DJ-Edge19	203.230.72.31

D elete

ID: 13b76a9c-f1c2-433d-a824-26ac8f0d5745

Name:

Required Bandwidth: 1 Gbps

Allocated Bandwidth: 1 Gbps

Auto Scale: false

Proactive: false

DHCP Server: KREONET-KR-Default

User List:

Active VDN Information

Location	Network Devices	Hosts/GW
Seoul	Seoul-Edge*	A0:36:9F:C5:52:A2/None
Seoul	Seoul-Edge*	203.230.72.176
Daejeon	KREONET-S-DJ-Edge19	134.75.238.118
Daejeon	KREONET-S-DJ-Edge19	203.230.72.31

Create
Back

Delete
Back

AI algorithm in KREONET-S

- VDN generation using REST API
 - Queueing system
 - Maximum queue delay
 - Service request rate
 - Service duration
 - Type of VDN service requests
 - High-bandwidth
 - Low-latency
 - Security / connectivity
 - Traffic generation (Ostinato)
 - According to type of VDN requests
 - Traffic volume
 - Traffic pattern

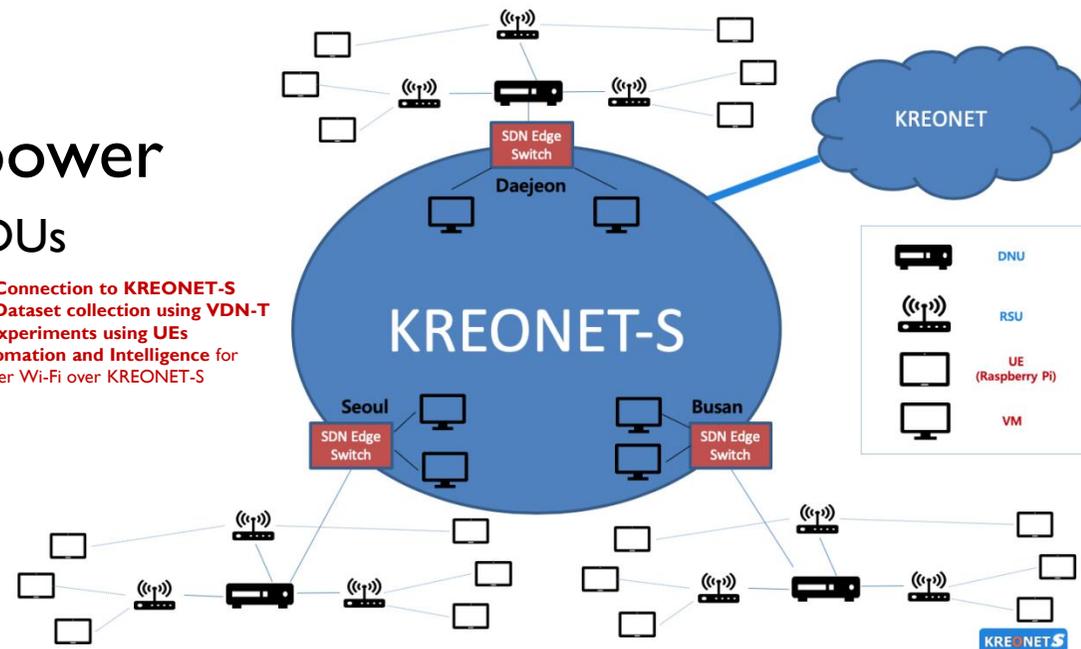
AI algorithm in KREONET-S

- Proposed algorithm
 - Deep Reinforcement Learning model
 - Agent (Discrete / Continuous action)
 - DQN, DDPG, PPO, etc.
 - Neural network model (critic / actor network)
 - RNN(LSTM) layer, Input/output layer, etc.
 - Training parameters
 - Optimizer, learning rate, discount factor, batch size, etc.
- Experiment & validation
 - Hybrid simulation (KREONET-S / simulation environment)

Future works

- Extended network & service scenario
 - Wireless environment (5G, WiFi, etc)
 - Traffic generation with real-time VDN requests
 - Prediction of time-series data (RNN)
 - Traffic usage
 - User requests (VDN)
 - Wireless environments
- Monitoring electricity power
 - Real-time power metering of PDUs

- UE Connection to KREONET-S
- UE Dataset collection using VDN-T
- NI experiments using UEs
- Automation and Intelligence for Carrier Wi-Fi over KREONET-S



Thank You!

Questions and/or Comments to
xkqpekdl@kisti.re.kr