Cybermedia Center
Osaka University

Cybermedia Center

Osaka University, Japan

SC18 BOOTH 1825

Access Control Based on Dynamic Network Management toward Connected-HPC

In the near future, data retrieved from IoT devices would be efficiently used and integrated in HPC simulations **(HPC x IoT)**. For the envisaged future, we have been developing an access control mechanism which an arbitrary set of IoT devices are dynamically connected to a HPC environment.

Background: Data Assimilation and IoT (Internet of Things) Era

What is Data Assimilation?

Data assimilation is a novel technique that combines observational data with numerical calculation for accurate and precise computation.

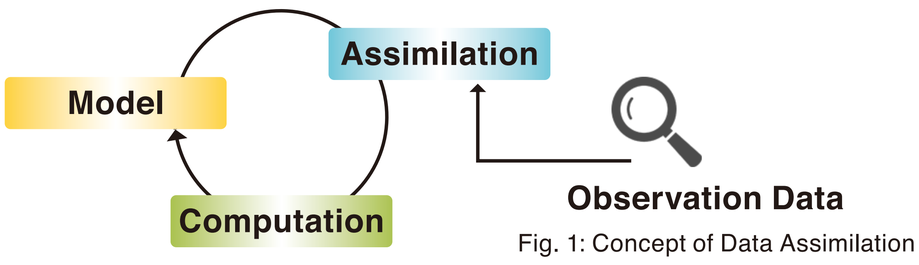

Fig. 1: Concept of Data Assimilation

It is predicted that the shorter the feeding interval of observation data to computation is, the higher the fineness of the result can be obtained. For the reason, IoT devices that capture observed data must be seamlessly integrated to HPC environments.

Sharing IoT Devices

In the IoT era, we envisage the world where an arbitrary set of IoT devices are used by an arbitrary group of users and applications. In other words, IoT devices are shared and used in an on-demand and sharing-economy fashion by an arbitrary group or community of users in the near future. For example, a set of cameras located in a lake may be used by a community of limnology scientists for a monthly period. On the other hand, the set of cameras may be used by some kind of weather monitoring application.

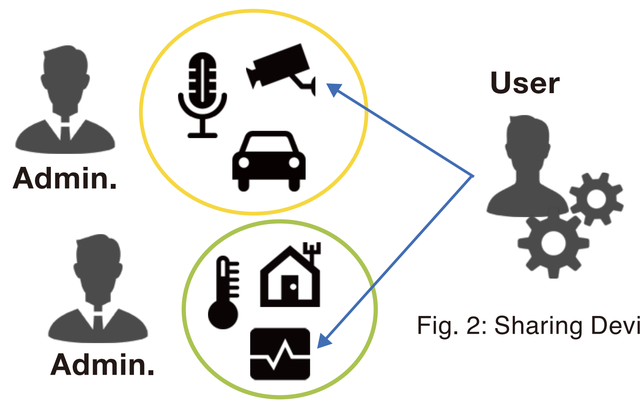

Fig. 2: Sharing Devices

Discussion: Real-time Data Assimilation and Issues

From the background described above, we work on the development of a new simulation technique that takes advantage of real-time observation data from an arbitrary set of IoT devices. We call this technique ***Real-time Data Assimilation (RDA)***. RDA can be applied to many scientific fields. To realize the simulation technique, **the HPC environment requires the connection to the outside network *(Connected-HPC)***. However, there are the following three technical issues to be tackled in today's HPC environments.

Fig. 3: Contributing to Many Fields by RDA

1. Packet reachability control is required from a security concern.

Today's HPC environment is usually a "closed" and "isolated" environment, meaning that HPC resources sit behind gateway servers connected to the external network. This situation is partly due to a fact that HPC administrators want to minimize the security risks from external networks. From this consideration, it would be helpful if we can connect and disconnect a HPC environment to the external network in an on-demand fashion.

2. Packet reachability should be controlled in a fine-grained fashion.

To minimize the security risks, the connect and disconnect of HPC resources to the external networks should be performed only when the access to IoT devices becomes necessary and unnecessary. For the reason, fine-grained control of network connection is preferred.

3. Network resources should be provided based on network administrators' policy.

Currently, a network is regarded as a shared "pipe". Taking the IoT era in the near future into consideration, an arbitrary IoT applications should be able to utilize resources of their interest including HPCs and networks based on their administrators' policy. From the idea above, a mechanism that can reflect users' needs and administrators' policy to network resources is essential.

Our Approach: Connected-HPC

For tacking the three technical issues above, we seamlessly integrate ***Software Defined Networking (SDN)*** and ***Job Scheduler*** (e.g. Slurm) for dynamic control of packet reachability between job and IoT devices for RDA technique.

To realize per-job packet reachability control as a fine-grained control of network connection, we have built a function that allows the scheduler to notify the SDN controller of the execution status of job. This notification contains the user ID on the HPC resource, and the IP address or ID of IoT devices to be connected/disconnected. Based on this notification, the SDN controller automatically searches the network path from HPC to the IoT devices and enables communication with those devices. **The function also enables the SDN controller to shut down the network path between HPC and IoT devices at the end of the job.**

Also, we have implemented **a function that the SDN controller can check whether the path adheres to the policy defined by the administrator and HPC user** before opening up the path between the HPC and the IoT devices. In other words, this check function makes it possible for the SDN controller to search for a path reflecting the policy of the stakeholders for each job.

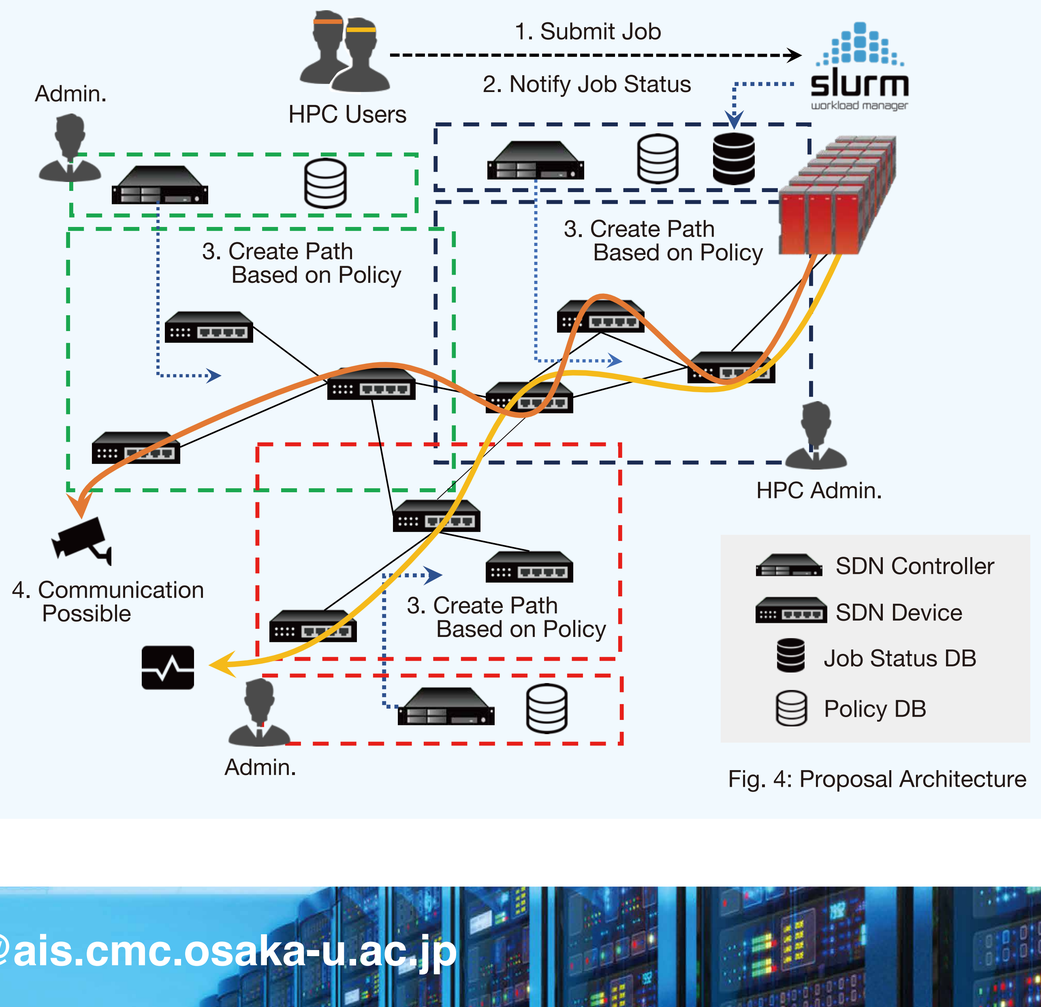

Fig. 4: Proposal Architecture

This work was partly supported by JSPS KAKENHI Grant Number JP17KT0083.

Hiroaki Morimoto: sc18@ais.cmc.osaka-u.ac.jp