

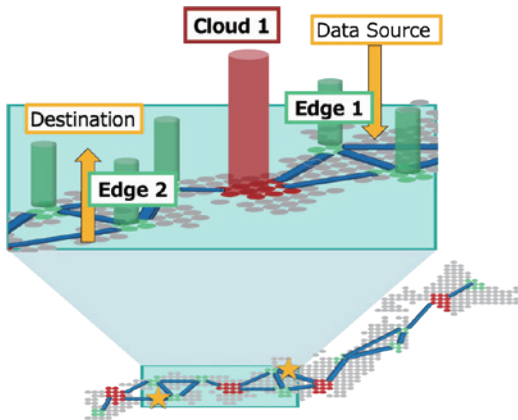
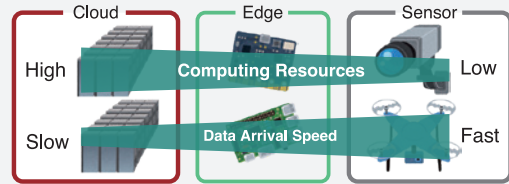
Towards On-Time Network and Container Management in Cloud-Edge Continuum

Key Factors for Resource Scheduling in Cloud-Edge Continuum

Cloud-Edge Continuum is a new computing paradigm for data processing.

- **Cloud** has **much computing resources** and far from sensor devices .
- **Edge** has a few computing resources and **close to sensor devices**.

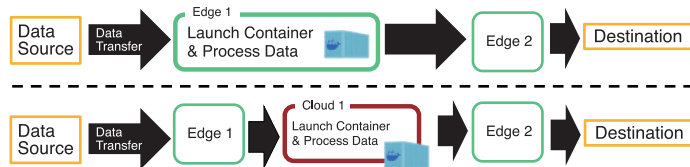
Cloud-Edge Continuum intelligently utilizes both the **Cloud and Edge** resources based on required latency and resource availability.



Resource scheduler plans the strategy of data processing

Strategy 1 (Edge-only): Edge 1 (Process) → Edge 2

Strategy 2 (Edge and Cloud): Edge 1 → Cloud 1 (Process) → Edge 2



Which strategy is better? In addition to the available resources,

- **Predictable data transfer time to each edge and cloud**
 - **On-time container creation for computing resources**
- are essential to make the correct decision.

Deterministic Networking (DetNet) Assisted Data Transfer

[What is DetNet] DetNet is a protocol suite for latency-guaranteed network over Ethernet.

In DetNet, packets are delivered in time, resulting in the data transfer time become predictable.

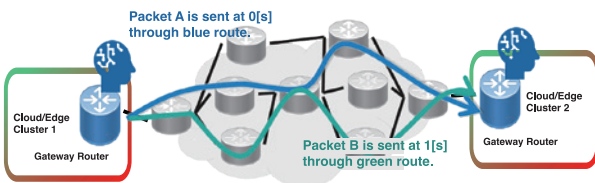
[What we do] DetNet is under developing and lack of scalability.

We are improving its scalability through dynamic traffic control and makes it free from time synchronization.

1. Dynamic Routing and Traffic Shaping with DRL

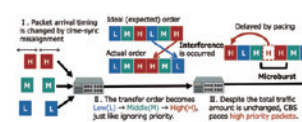
Latency due to traffic congestion can be prevented through isolation of route and transmission timing.

We are working on **the machine learning model to select the best route and transmission timing** for every packet.

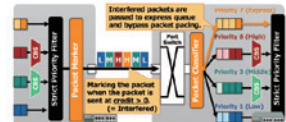


2. Time-Synchronization-Free DetNet

Credit-Based Shaper (CBS) based DetNet is relatively resilience to time-sync issues. However, **in some cases, CBS becomes unstable due to time-sync issues**. We eliminate such cases through a mathematical approach and protocol improvements.



Identifying the case when CBS becomes unstable



Proposed extension for CBS to improve its stability (CBS-Bypass)

On-Time Container Creation with Checkpoint and Restore

Container creation consumes a significant amount of time due to the initialization processes (called Cold-Start Problem).

There is potential to reduce container creation time through the reuse of initialized containers.

We are developing a container management system that applies checkpoint and restore features.

When the container is created from scratch, the cold start occurs during the initialization processes.

[Solution] Avoiding the container initialization by restoring a fully initialized container from checkpoint when creating the new container.



- After container initialization, the container manager takes a checkpoint.
- Restoring the container's checkpoint when it is newly initialized can mitigate the cold start problem.

