



**CLOUD NATIVE &
KUBERNETES**

AI DAY

NORTH AMERICA

Dressing-up your cluster for AI in minutes with a portable network CR



CLOUD NATIVE &
KUBERNETES

AI DAY

NORTH AMERICA

November 12, 2024
Salt Lake City



Tatsuhiko Chiba

Senior Technical Staff Member
IBM Research



Sunyanan Choochotkaew

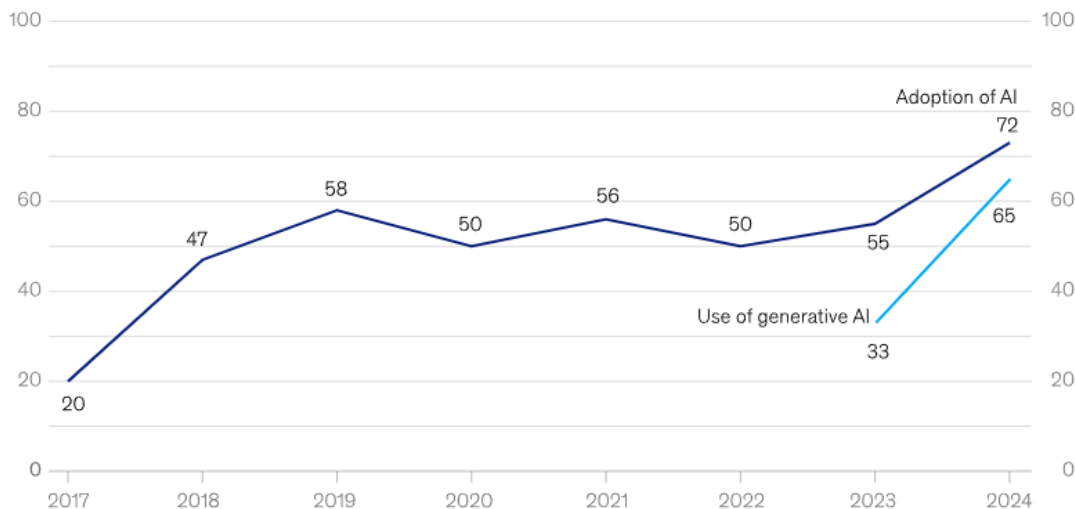
Staff Research Scientist
IBM Research

AI adoption has increased dramatically, especially after the rise of generative AI.

2X Use of GenAI in 2024

AI adoption worldwide has increased dramatically in the past year, after years of little meaningful change.

Organizations that have adopted AI in at least 1 business function,¹ % of respondents



¹In 2017, the definition for AI adoption was using AI in a core part of the organization's business or at scale. In 2018 and 2019, the definition was embedding at least 1 AI capability in business processes or products. Since 2020, the definition has been that the organization has adopted AI in at least 1 function.
Source: McKinsey Global Survey on AI, 1,363 participants at all levels of the organization, Feb 22–Mar 5, 2024

Trending: AI workloads in Containers

Consistent Environments

Rapid Experimentation

Simplified Dependency Management

Accelerating GenAI Adoption with Containers:
Unleashing AI/ML Development at Scale
- Ivan Curkovic , Engineering Manager, Docker
at Open Source Summit Japan 2024 on October 28, 2024



IBM Cloud Vela: IBM's first AI-optimized, cloud-native supercomputer



- Vela is an A100 supercomputer built on top of IBM Cloud VPC (operational by May 2022)
- 8 x A100 GPUs with 80G of memory
- Virtualization with baremetal performance (<~5%)
- 400G Ethernet network (RoCE / GPU Direct RDMA)
- Supports training models in Billions of parameters that process Trillions of tokens
- Multi-NIC CNI dynamically provides **a pod direct communication** with a secondary network

watsonx.ai

Train and validate

Pre-processing



Model training



Validation



Models

Suite of IBM trained foundation models



Tune and infer

Studio



Model serving



Hybrid cloud platform



MCAD

Job dispatching, queuing and packing

Multi-NIC CNI

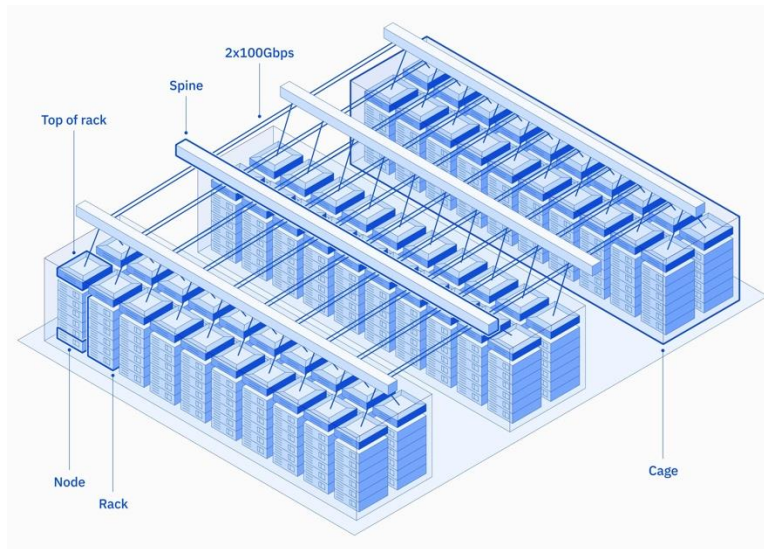
High perf network



Auto-pilot

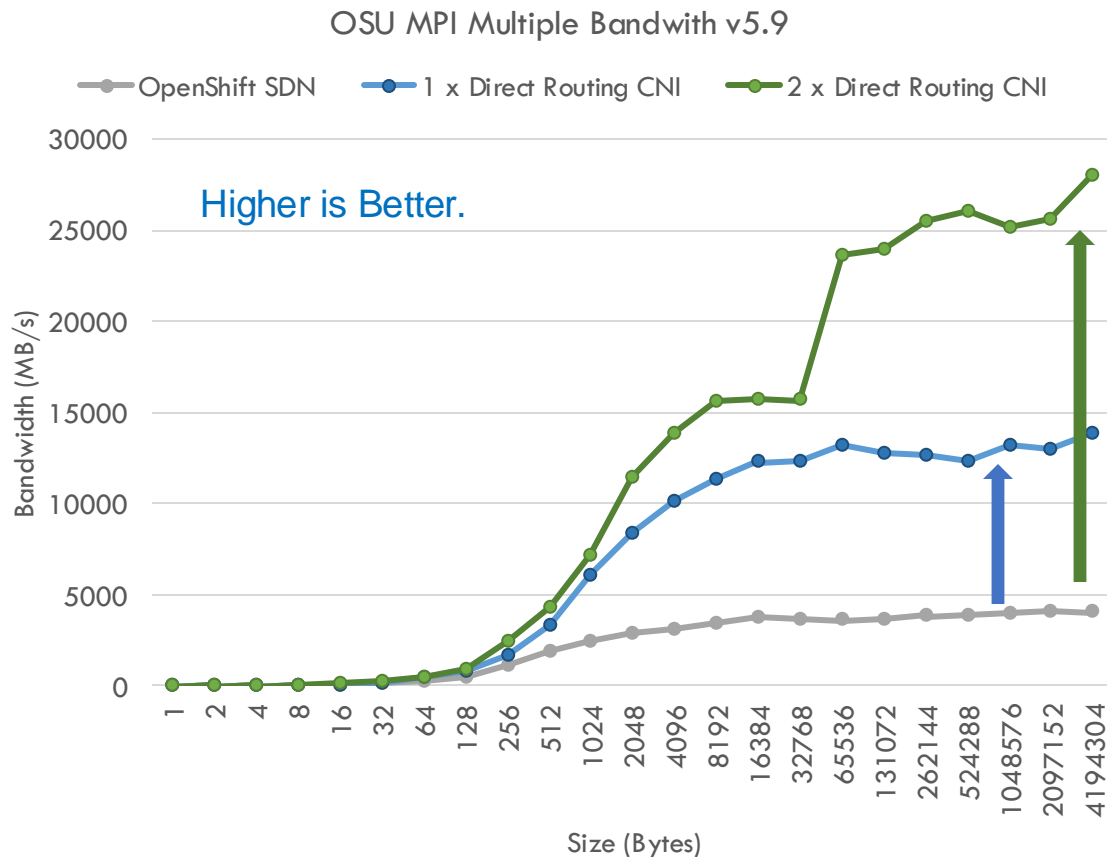
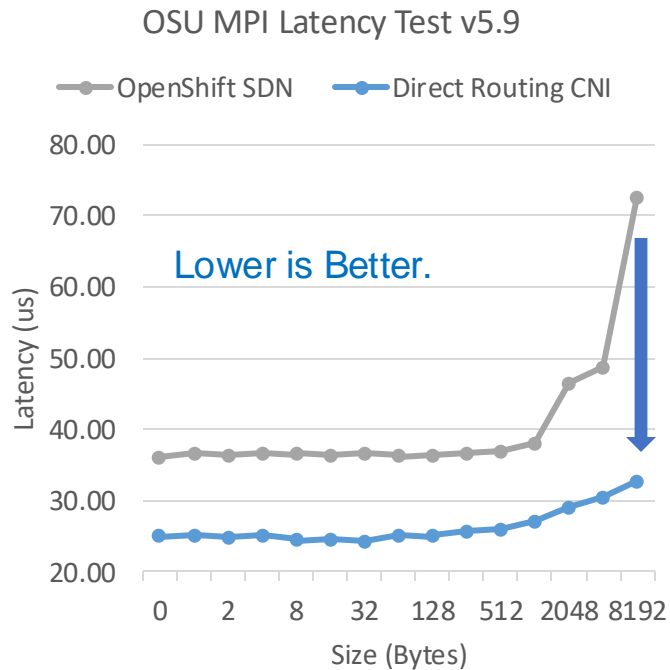


TORCHX

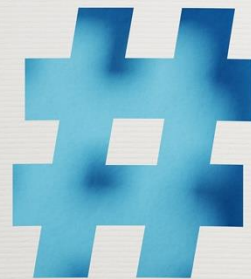


A Common Goal of AI Networking Solution in Containers

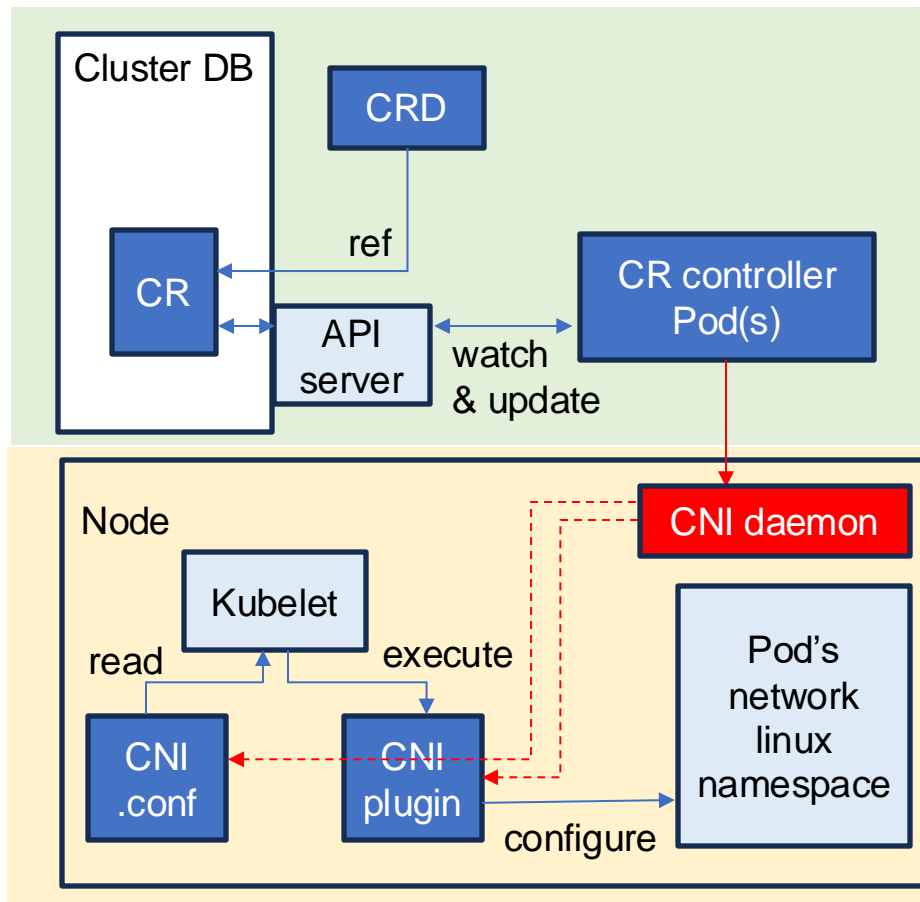
Native Performance at Scale



- Background – *CNI, CR*
- Reasons behind “A Portable” Network CR
- Idea – *in a very simplified version*
- Our implementation – *Multi-NIC CNI v1*
- What’s Next?



Network CR



CR (custom resource) is commonly used to serve inputs to CNI for more flexibility in network configuration.

Multus has **NetworkAttachmentDefinition CR** to delegate execution for secondary networks

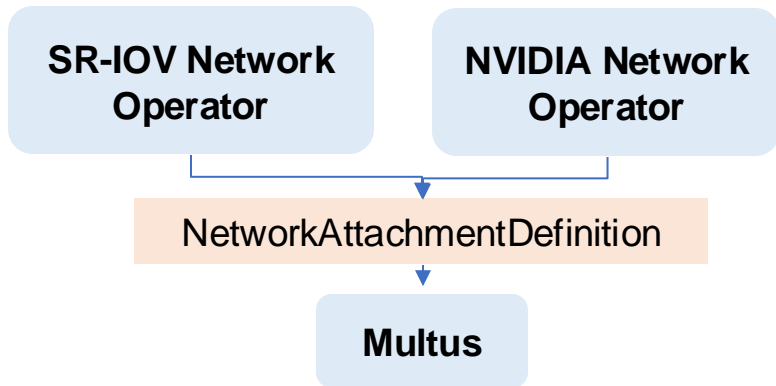
SR-IOV Network Operator has **SriovNetwork CR**

NVIDIA Network Operator has **MacvlanNetwork, HostDeviceNetwork, IPoIBNetwork**

Our CNI names Multi-NIC CNI, our CR would name ???

Basic of CNIs for High Performance Network Device

- Provide a pod direct communication of high performance network device, separating from the primary network, with a secondary network attachment by Multus.
- **Multus** attaches secondary networks by network plugins,
 - defined in NetworkAttachmentDefinition CR,
 - to be annotated in Pod Annotation.



1. Define NetworkAttachmentDefinition CR

```
apiVersion: "k8s.cni.cncf.io/v1"
kind: NetworkAttachmentDefinition
metadata:
  name: macvlan-conf
spec:
  config: '{
    "cniVersion": "0.3.0",
    "type": "macvlan",
    "master": "eth0",
    "mode": "bridge",
    "ipam": {
      "type": "host-local",
      "subnet": "192.168.1.0/24",
      "rangeStart": "192.168.1.200",
      "rangeEnd": "192.168.1.216",
      "routes": [
        { "dst": "0.0.0.0/0" }
      ],
      "gateway": "192.168.1.1"
    }
  }'
```

2. Pod Annotation

```
apiVersion: v1
kind: Pod
metadata:
  name: samplepod
  annotations:
    k8s.v1.cni.cncf.io/networks: macvlan-conf
```

Pod direct communication is a key of Pod's networking performance.
However, direct routing is not always available as-is, *especially on virtual cloud*.

	TCP/IP with L3	ENI on AWS	RoCE v2
CNI & Routing solution	Host Routing + IPVLAN	VPC API for IP Registration	Host Device + Host IP
Complexity	Options: 1. A central pool with per-pod IP routing configuration → <i>not scale</i> 2. Per-host per-interface pool with static configuration	IP in valid range of scheduled host	Static IP for each host and each device
#CR for N host with M interfaces	N xM	N	N xM

Limitation 1 of as-is solution

Host-dependent IP requirement → Host must be known at Pod annotating point.

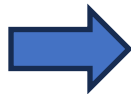
1. Define NetworkAttachmentDefinition CR (s)

**Multiple NetworkAttachmentDefinitions
for direct routing (as-is)**

NetAttachDef
for Host A

NetAttachDef
for Host B

NetAttachDef
for Host C

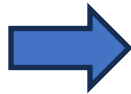


2. Pod Annotation

Pod
annotations:
k8s.v1.cni.cncf.io/networks: *either A or B or C*

A portable network CR (to-be)

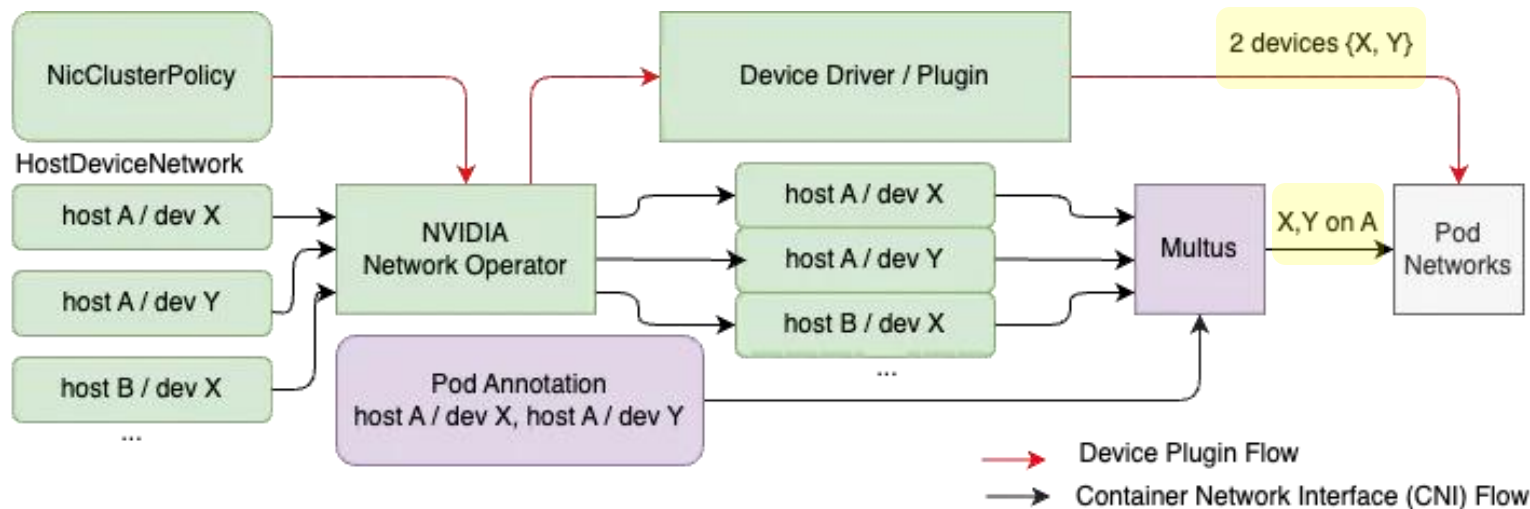
NetAttachDef X



Pod
annotations:
k8s.v1.cni.cncf.io/networks: X

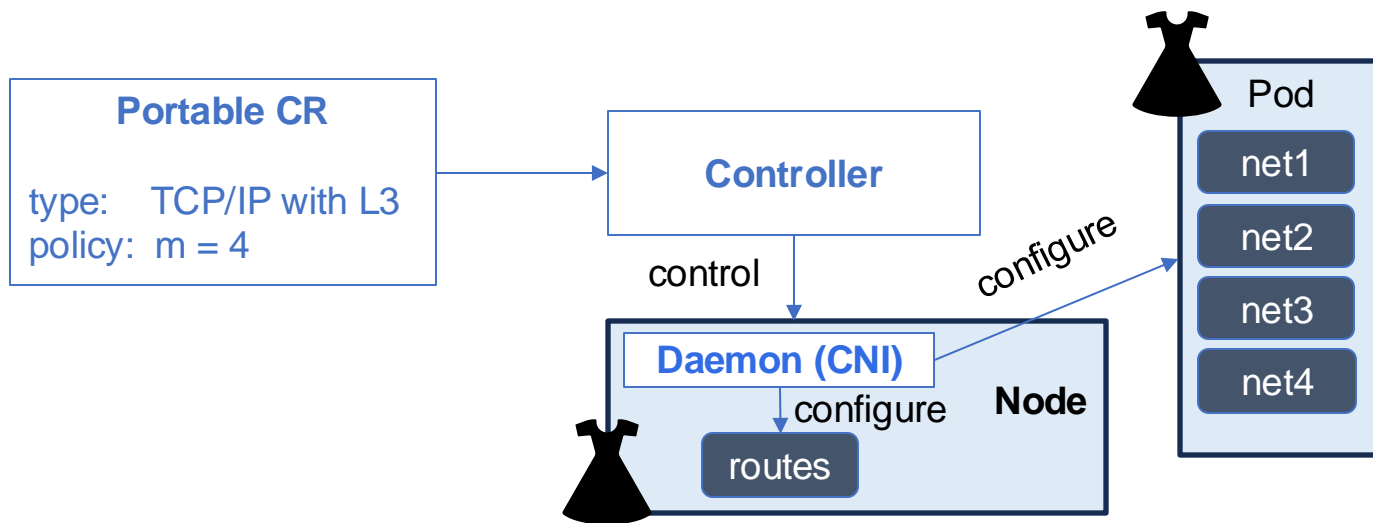
Limitation 2 of as-is solution (RoCE v2)

*RoCE device plugin does not aware of Pod's annotation
→ Misorder of IP assignment for interface-dependent valid range*



Idea of a portable network CR to dress up your cluster

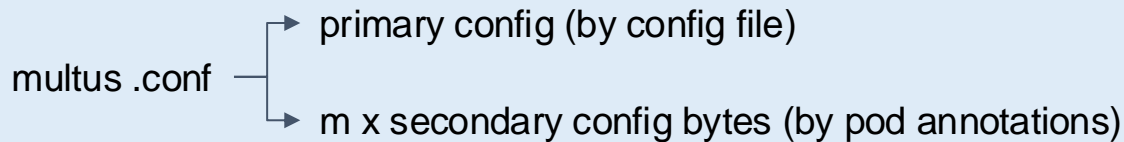
1. User annotates **a single network for m secondary interfaces**.
2. The rest tasks (interface discovery, routing, IP assignment) are **automated**.



An implementation to unpack single configuration to multiple configurations after executions of scheduler and device manager

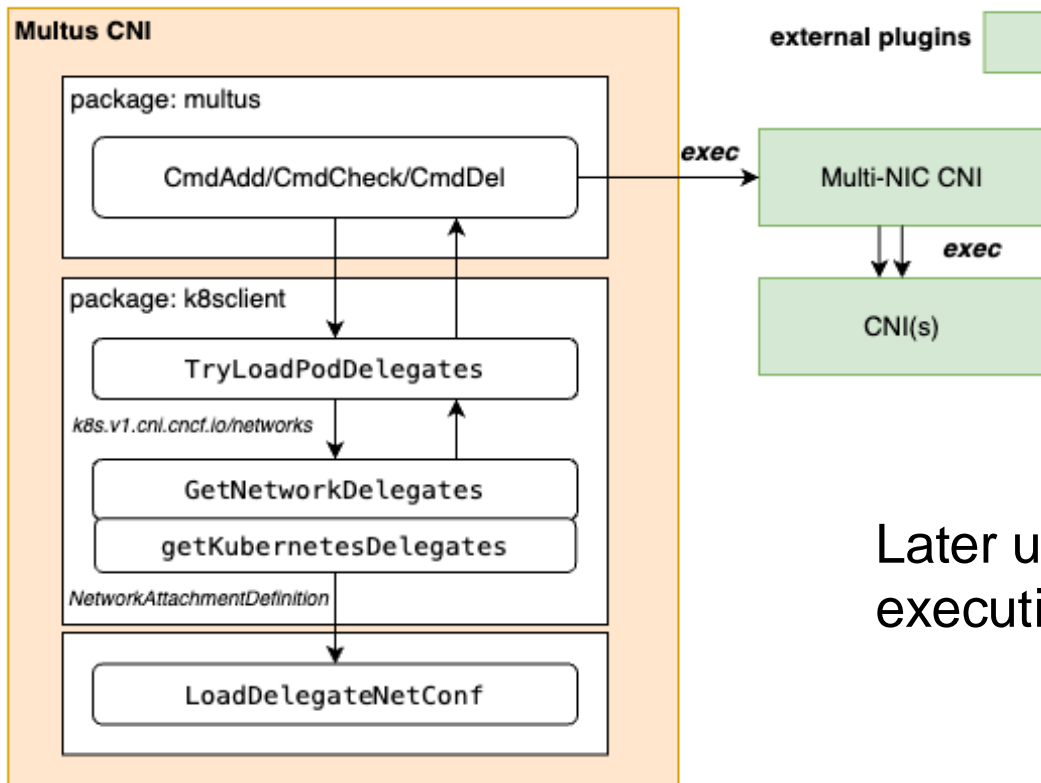
Remark:

Multus does 1-to-m configurations.



Even so, it unpacks the configuration *before host is scheduled.*

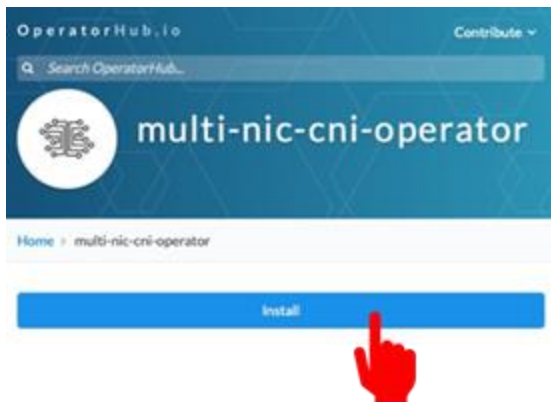
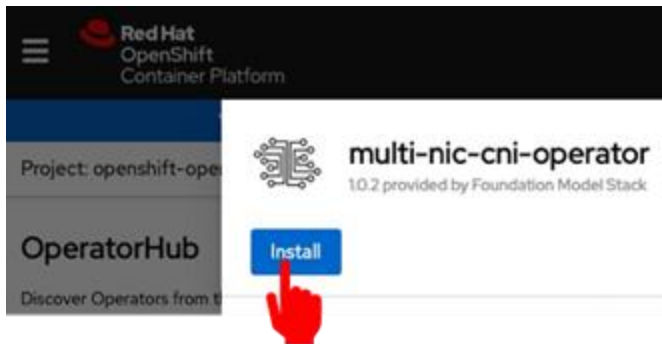
Multi-NIC CNI – Mechanism Behind



Later unpack and delegate CNI executions after Multus delegation

Multi-NIC CNI – 3 magic steps

1. Install Operator



2. Deploy MultiNicNetwork (for TCP/IP L3 cluster)

```
apiVersion: multinic.fms.io/v1
kind: MultiNicNetwork
metadata:
  name: multi-nic-sample
spec:
  subnet: "192.168.0.0/16"
  ipam: |
    {
      "type": "multi-nic-ipam",
      "hostBlock": 6,
      "interfaceBlock": 2,
      "vlanMode": "13"
    }
  multiNICIPAM: true
plugin:
  cniVersion: "0.3.0"
  type: ipvlan
  args:
    mode: 13
```

3. Attach Network to Pod

```
metadata:
  annotations:
    k8s.v1.cni.cncf.io/networks: multi-nic-sample
```

Multi-NIC CNI - DEMO



CLOUD NATIVE &
KUBERNETES
AI DAY
NORTH AMERICA

```
root@mplatlab-worker-0_josu-micro-benchmarks-5.6.3 (bash)
bash-3.2$

@cutterj (watch)
Every 2.0s: oc get po                               Sunyans-MacBook-Pro, local: Wed Nov  6 12:49:01 2024
No resources found in default namespace.

bash
bash-3.2$ watch oc get benchmark mpilat -o=jsonpath='{.status.jobCompleted}'
```

mvapich.cse.ohio-state.edu

Home Overview Download Performance Benchmarks Tools Publications Talks Users Best Practices

Support MUG Jobs

MVAPICH: MPI over InfiniBand, Omni-Path, Ethernet/iWARP, RoCE, and Slingshot

Network-Based Computing Laboratory

- OSU Micro-Benchmarks 7.5 (11/01/24) [Tarball]
 - Please see [CHANGES](#) for the full changelog.
 - You may also take a look at the appropriate README files for more information.
 - C Benchmarks [README](#).
 - Java Benchmarks [README](#).
 - Python Benchmarks [README](#).
 - The benchmarks are available under the [BSD license](#).
- Here, we list various benchmarks that are part of the OMB package in the C, Java, and Python programming languages for various parallel programming models like MPI, OpenSHMEM, UPC, UPC++, and NCCL. A high-level description of these benchmarks are provided below:
 - C Benchmarks
 - MPI
 - Host-based Benchmarks
 - Point-to-Point MPI Benchmarks: Latency, multi-threaded latency, multi-pair latency, multiple bandwidth / message rate test, bandwidth, bidirectional bandwidth
 - Blocking Collective MPI Benchmarks: Collective latency tests for various MPI collective operations such as MPI_Allgather, MPI_Alltoall, MPI_Allreduce, MPI_Barrier, MPI_Bcast, MPI_Gather, MPI_Reduce, MPI_Reduce_Scatter, MPI_Scatter and vector collectives.
 - Non-Blocking Collective (NBC) MPI Benchmarks: Collective latency and Overlap tests for various MPI collective operations such as MPI_lallgather, MPI_lallreduce, MPI_lalltoall, MPI_lbarrier, MPI_lbroadcast, MPI_lgather, MPI_lreduce, MPI_lscatter and vector collectives.
 - One-sided MPI Benchmarks: one-sided put latency, one-sided put bandwidth, one-sided put bidirectional bandwidth, one-sided get latency, one-sided get bandwidth, one-sided accumulate latency, compare and swap latency, fetch and operate and get_accumulate latency for MVAPICH2 (MPI-2 and MPI-3).
 - Startup Benchmarks: `osu_init`, `osu_hello`
 - Device-based Benchmarks
 - CUDA, ROCm, and OpenACC Extensions to OSU Micro Benchmarks
 - Support for CUDA Managed Memory
 - OpenSHMEM Benchmarks
 - Point-to-Point OpenSHMEM Benchmarks: put latency, get latency, message rate, atomics,
 - Collective OpenSHMEM Benchmarks: collect latency, broadcast latency, reduce latency,

Multi-NIC CNI – Currently Supported Solutions



CLOUD NATIVE &
KUBERNETES
AI DAY
NORTH AMERICA

TCP/IP with L3

```
apiVersion: multinic.fms.io/v1
kind: MultiNicNetwork
metadata:
  name: multinic-ipvlanl3
spec:
  subnet: "192.168.0.0/16"
  ipam: |
    {
      "type": "multi-nic-ipam",
      "hostBlock": 8,
      "interfaceBlock": 2,
      "vlanMode": "13"
    }
  multiNICIPAM: true
  plugin:
    cniVersion: "0.3.0"
    type: ipvlan
    args:
      mode: 13
```

ENI on AWS




```
apiVersion: multinic.fms.io/v1
kind: MultiNicNetwork
metadata:
  name: multinic-aws-ipvlan
spec:
  ipam: |
    {
      "type": "multi-nic-ipam",
      "hostBlock": 8,
      "interfaceBlock": 2,
      "vlanMode": "12"
    }
  multiNICIPAM: true
  plugin:
    cniVersion: "0.3.0"
    type: aws-ipvlan
    args:
      mode: 12
```

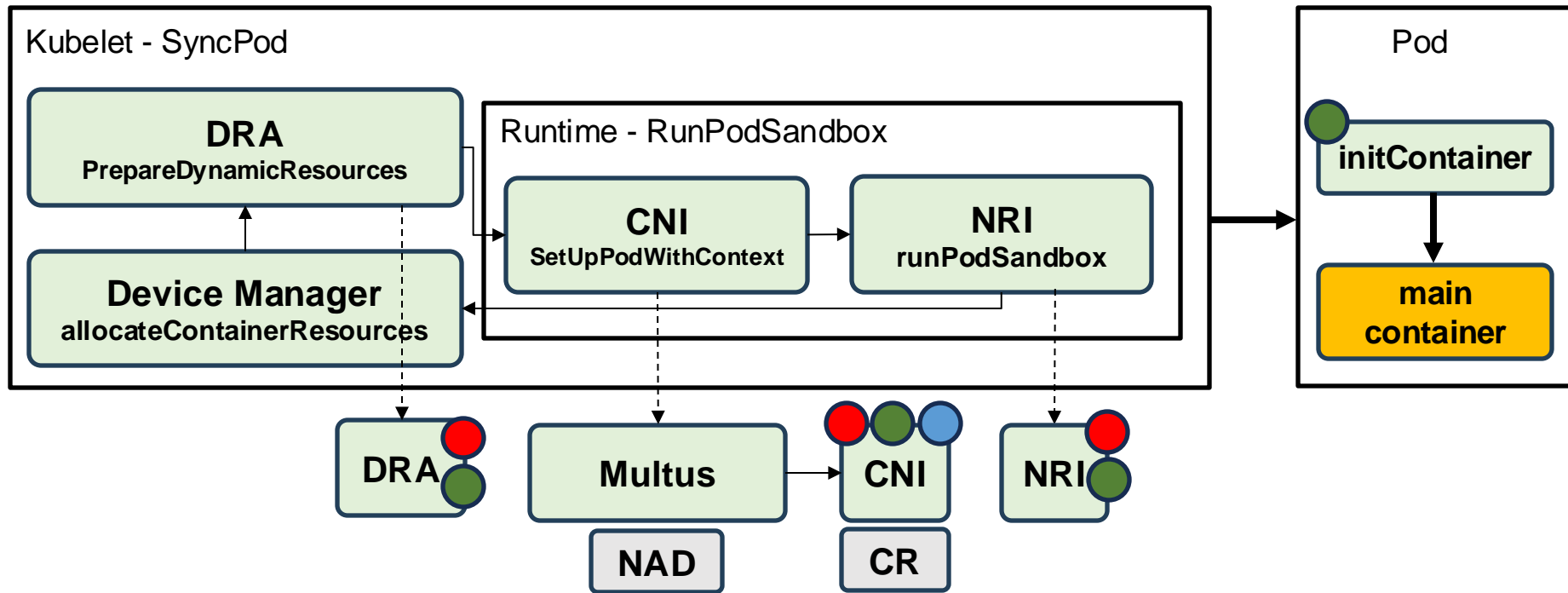
RoCE v2

```
apiVersion: multinic.fms.io/v1
kind: MultiNicNetwork
metadata:
  name: multinic-mellanox-hostdevice
spec:
  subnet: ""
  ipam: |
    {
      "type": "host-device-ipam"
    }
  multiNICIPAM: false
  plugin:
    cniVersion: "0.3.1"
    type: mellanox
```

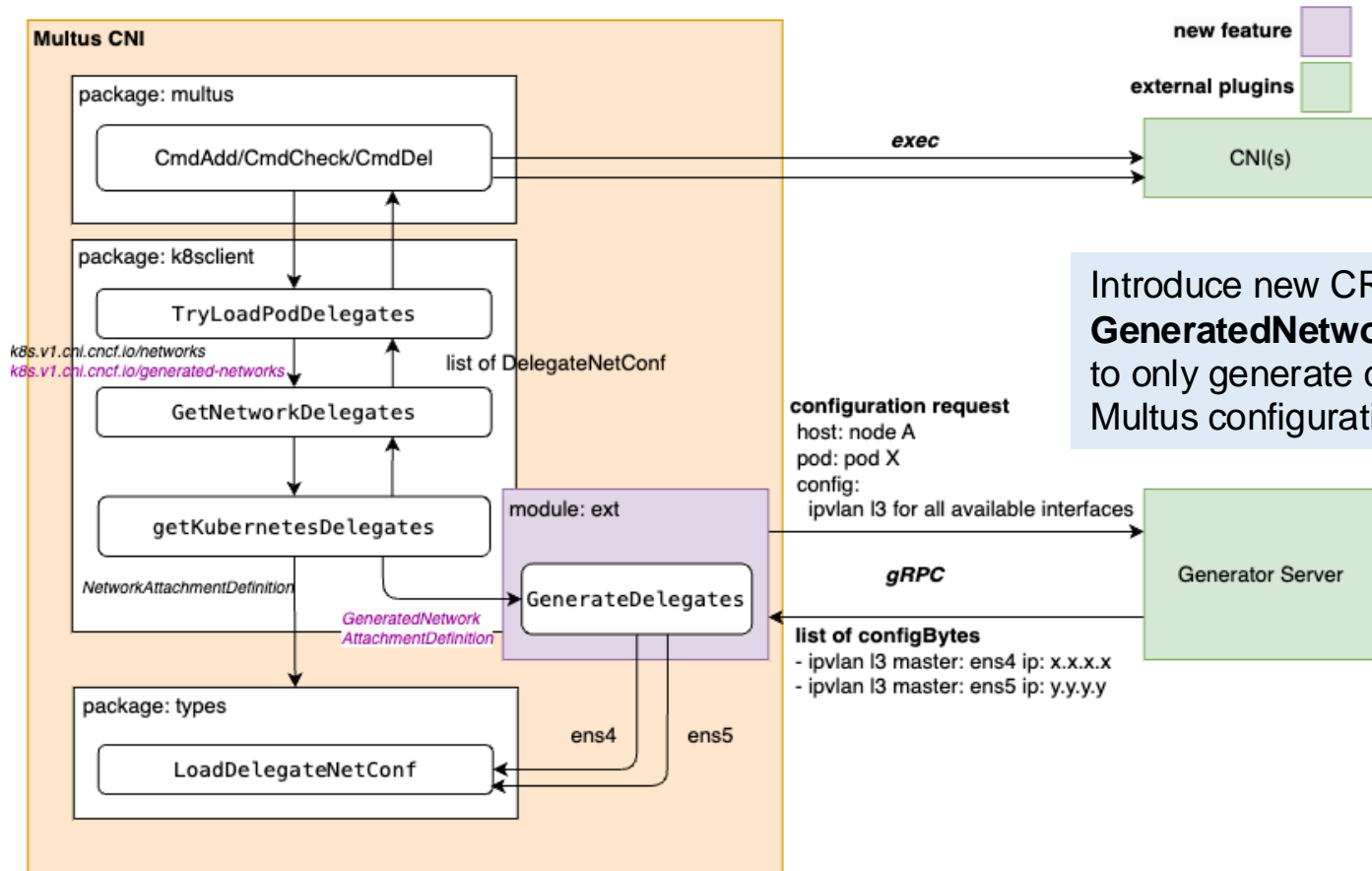
Alternative implementation – with current k8s

Key Functions

-  select interfaces and assign dynamic host-dependent IP
-  configure Pod's network namespace
-  register the IP/configure route for IP



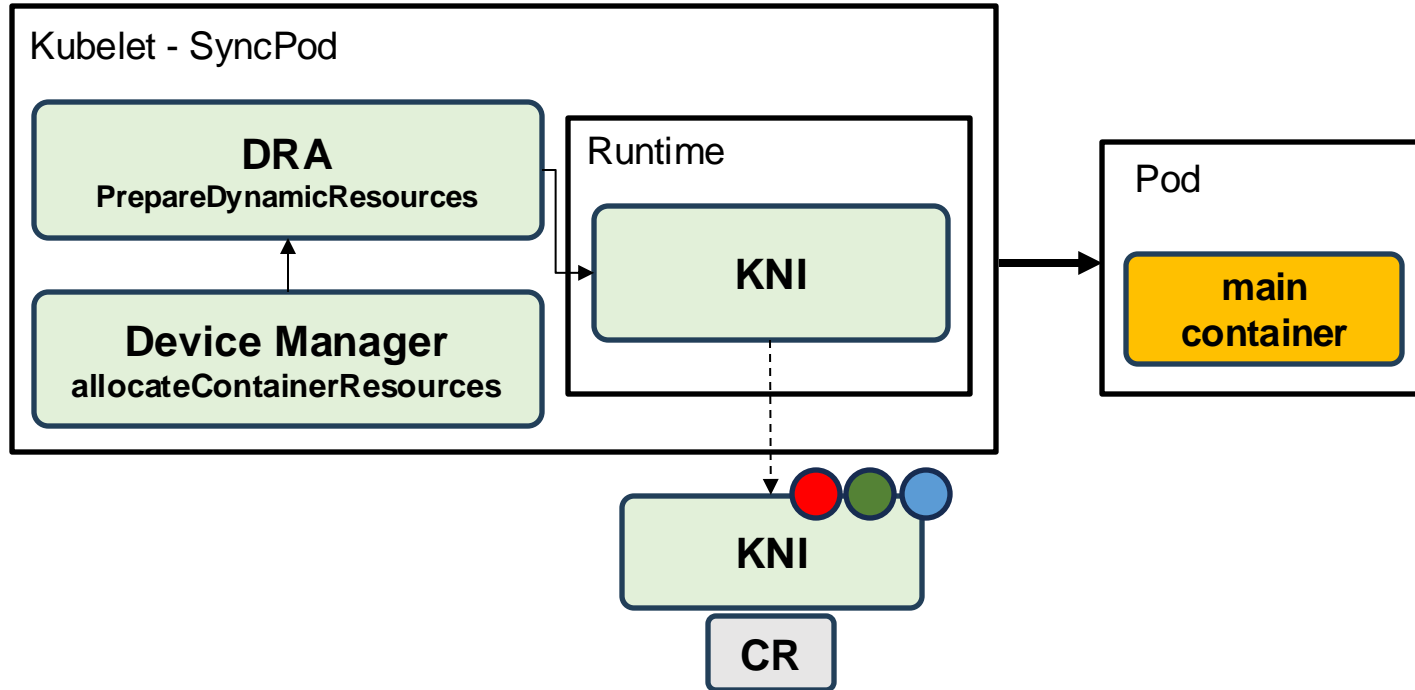
Alternative implementation – with Multus extension



Introduce new CR
GeneratedNetworkAttachmentDefinition
to only generate config dynamically within
Multus configuration load function.

Alternative implementation – with future k8s

- assign dynamic host-dependent IP
- register the IP/configure route for IP
- configure the dynamic IP





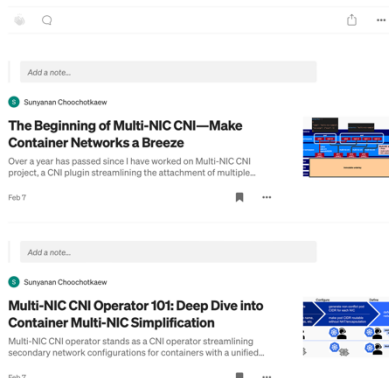
Open Source Project

[https://github.com/
foundation-model-stack/
multi-nic-cni](https://github.com/foundation-model-stack/multi-nic-cni)



Medium Blogs

Multi-NIC CNI Series



arXiv:2407.05467v1 [cs.DC] 7 Jul 2024

IBM

Vela and Blue Vela AI Infrastructure

The infrastructure powering IBM's Gen AI model development

Talia Gershon* Seetharami Seelam* Brian Belgodere* Milton Bonilla* Lan Hoang Danny Barnett I-Hsin Chung Apoorve Mohan Ming-Hung Chen Lixiang Luo Robert Walkup Constantinos Evangelinos Shweta Salaria Marc Dombrowa Yoonho Park Apo Kayi Liran Schour Alim Alim Ali Sydney Pavlos Maniotis Laurent Schares Bernard Metzler Bengi Karacali-Akyamac Sophia Wen Tatsuhiko Chiba Sunyanan Choochotkaew Takeshi Yoshimura Claudia Misale Tonia Elengikal Kevin O'Connor Zhuoran Liu Richard Molina Lars Schneiderbach James Caden Christopher Laibinis Carlos Fonseca Vasily Tarasov Swaminathan Sundararaman Frank Schmuck Scott Guthridge Jeremy Cohn Marc Eshel Paul Muench Runyu Liu William Pointer Drew Wyskida Bob Krull Ray Rose Brent Wolfe William Cornejo John Walter Colm Malone Clifford Perucci Frank Franco Nigel Hinds Bob Calio Pavel Druyan Robert Kilduff John Kienle Connor McStay Andrew Figueroa Matthew Connolly Edie Fost Gina Roma Jake Fonseca Ido Levy Michele Payne Ryan Schenkel Amir Malki Lion Schneider Aniruddha Narkhede Shekha Moshref Alexandra Kisin Olga Dodin Bill Rippon Henry Wrieth John Ganci Johnny Colino Donna Habeger-Rose Rakesh Pandey Aditya Gidh Aditya Gaur Dennis Patterson Samsuddin Salmani Rambilas Varma Rumana Rumana Shubham Sharma Aditya Gaur Mayank Mishra Rameswar Panda Aditya Prasad Matt Stallone Gaoyuan Zhang Yikang Shen David Cox Ruchir Puri Dakshi Agrawal IBM Research

Drew Thorstensen Joel Belog Brent Tang Saurabh Kumar Gupta Amitabha Biswas Anup Maheshwari Eran Gampel Jason Van Patten Matthew Runion Sai Kaki Yigal Bogin Brian Reitz Steve Pritko Shahan Najam Surya Nambala Radhika Chirra Rick Welp Frank DiMitri Felipe Telles Amilcar Arvelo King Chu Ed Seminario Andrew Schram Felix Eickhoff William Hanson Eric McKeever Dinakaran Joseph Piyush Chaudhary Piyush Shivam Puneet Chaudhary Wesley Jones Robert Guthrie Chris Bostic Rezaul Islam Steve Duersch Wayne Sawdon John Lewars Matthew Klos Michael Spriggs Bill McMillan George Gao IBM Infrastructure

Ashish Kamra Gaurav Singh Marc Curry Tushar Katarki Joe Talerico Zenghui Shi Sai Sindhur Malleni Erwan Gallen Red Hat

*Corresponding Authors:

tsgersho@us.ibm.com, sseelam@us.ibm.com, bmbelgod@us.ibm.com, bonillam@us.ibm.com

Thank you !



Sched.com - Rate & Feedback

Thank you to our Diamond Sponsors!



CLOUD NATIVE &
KUBERNETES
AI DAY
NORTH AMERICA



Red Hat

Thank you to our Platinum Sponsor!



CLOUD NATIVE &
KUBERNETES
AI DAY
NORTH AMERICA

sysdig

Thank you to our Gold Sponsor!



CLOUD NATIVE &
KUBERNETES
AI DAY
NORTH AMERICA



Thank you to our Recording Sponsor!



CLOUD NATIVE &
KUBERNETES
AI DAY
NORTH AMERICA

