OpenShift/K8s CNF disaggregation from CPU centric computing across core and edge

Hyde Sugiyama
Chief Architect, Red Hat
email: hyde@redhat.com

www.isocore.com/2021

NET-CENTRIC

Disclaimer

This presentation discusses technology trends and evolving concepts and does not aim to provide a committed feature roadmap or any kind of product announcement.

All content is subject to change based on status of each open source upstream project.

Data Makes Hardware Matter Again

The Issue: Baseline Power Necessary Per Transistor.

Moore's Law...and Dennard Scaling?

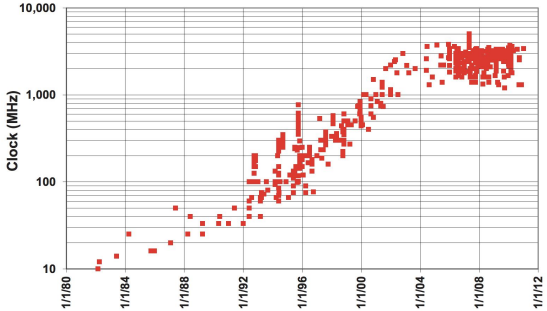
Dennard (1974): "voltage and current should be proportional to the linear dimensions of a transistor."

Reducing transistor size & voltage increases transistor density & speed while maintaining power density

Reality: Dennard Scaling has a limit -- as transistors get smaller, power density eventually increases.

Result is a "Power Wall" that limits processor frequency.

CPUs are not getting 'faster' anymore.



NIC evolution - Everyday NICs getting Smarter

Everyday NICs

Flow steering - RSS, multi queue | TCP/UDP segmentation TSO, GRO | Encapsulation with Checksum -IP, IPv6, VLAN, VXLAN, GRE, Geneve | SR-IOV - scalable interfaces VFs, rate limiting | Timing and Synch - PTP | eXpress Datapath XDP native(Drop, Receive, Maps) | vDPA - virtio offload

Crypto Offload(Crypto Engine on chip)

Inline IPsec offload | TLS/kTLS Acceleration | eBPF sockmap redirect TLS | QUIC - HTTP/3(HTTP+TLS+UDP)

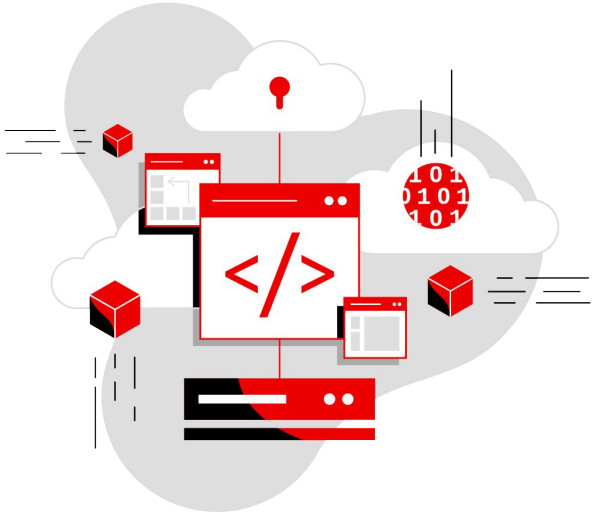
Programmable Flow Offload

Flow ASIC or pre-programed FPGA **OVS TC/flower offload** - vSwitch acceleration
Flow match and action; 5-tuple flow match | Encap/Decap - VLAN,QinQ,VXLAN,GRE, Geneve | MPLS Segment Routing , SRV6 | Bonding, remote mirroring | Quality of service - marking, rate limiting, bandwidth guarantees | Network policy - ACLs, Connection tracking

Programmable FPGA Load custom firmware new or custom protocols

Internet service provider vBNG
5GC UPF
5G Edge vRAN acceleration(FEC, eCPRI), etc

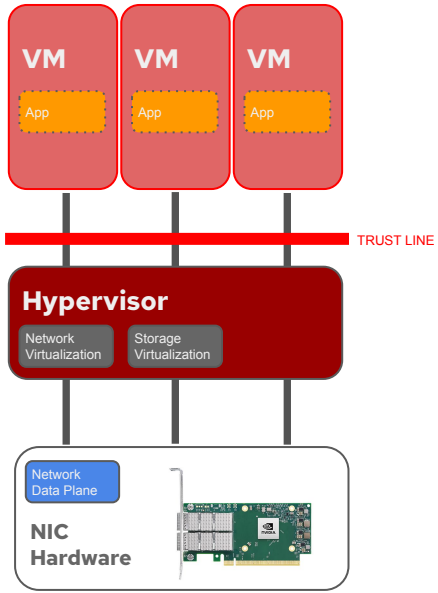
Making Use of Domain-Specific Hardware



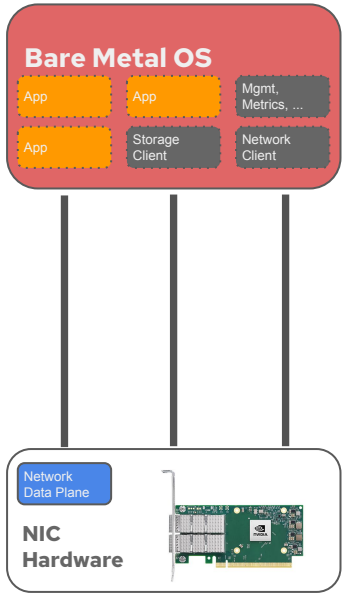
Hypervisors are unable to effectively abstract domain-specific hardware

Security Isolation & Trust

Virtualization



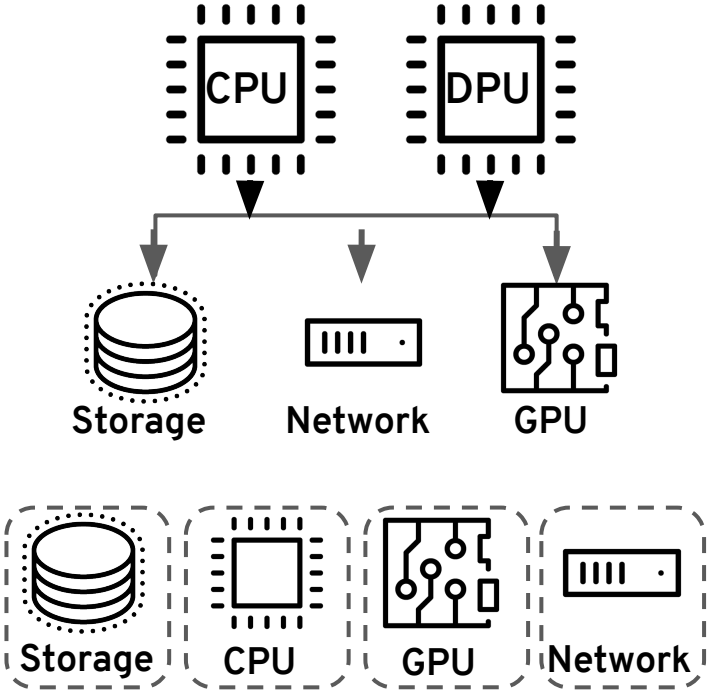
Bare Metal



Need for Architectural Compartmentalization

Mismatch of software to hardware abstractions and trust boundaries

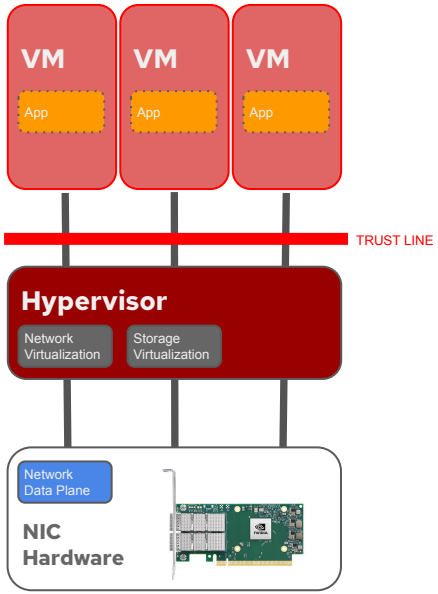
Disaggregated and Composable System Architecture



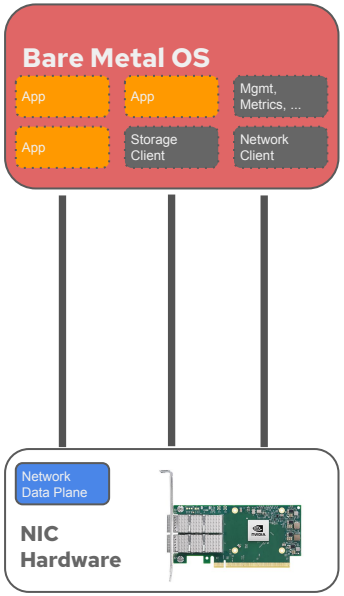
Move from CPU-centric architecture to collection of independent devices and SW-defined device functions

Security Isolation / Cloud

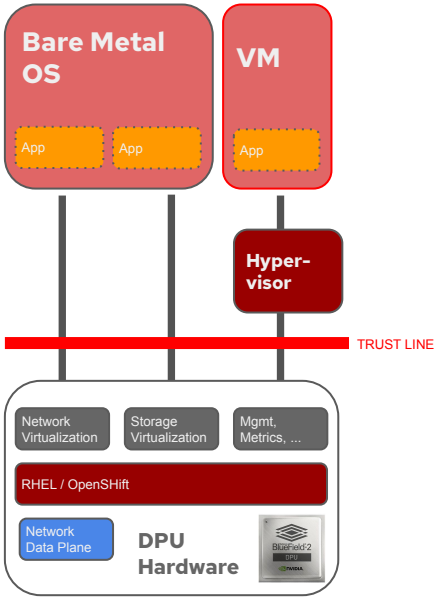
Virtualization



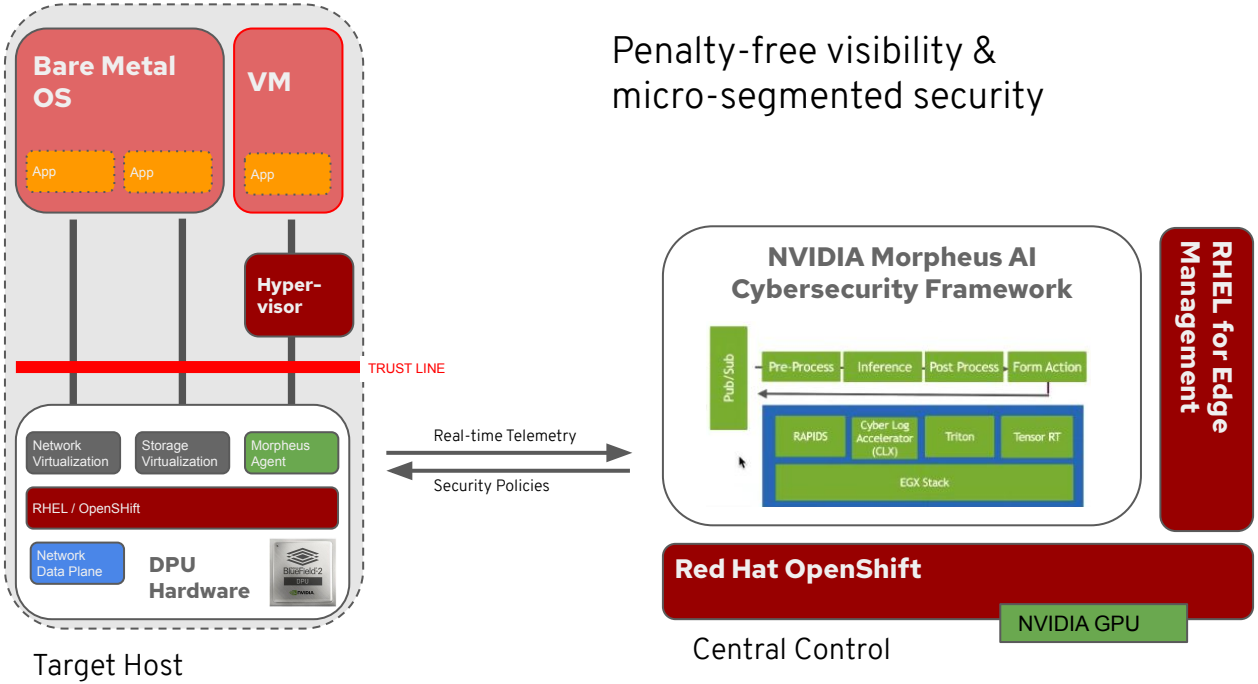
Bare Metal



DPU + B/M, Containers, Virt



NVIDIA Morpheus AI Cybersecurity Framework & Red Hat Enterprise Linux

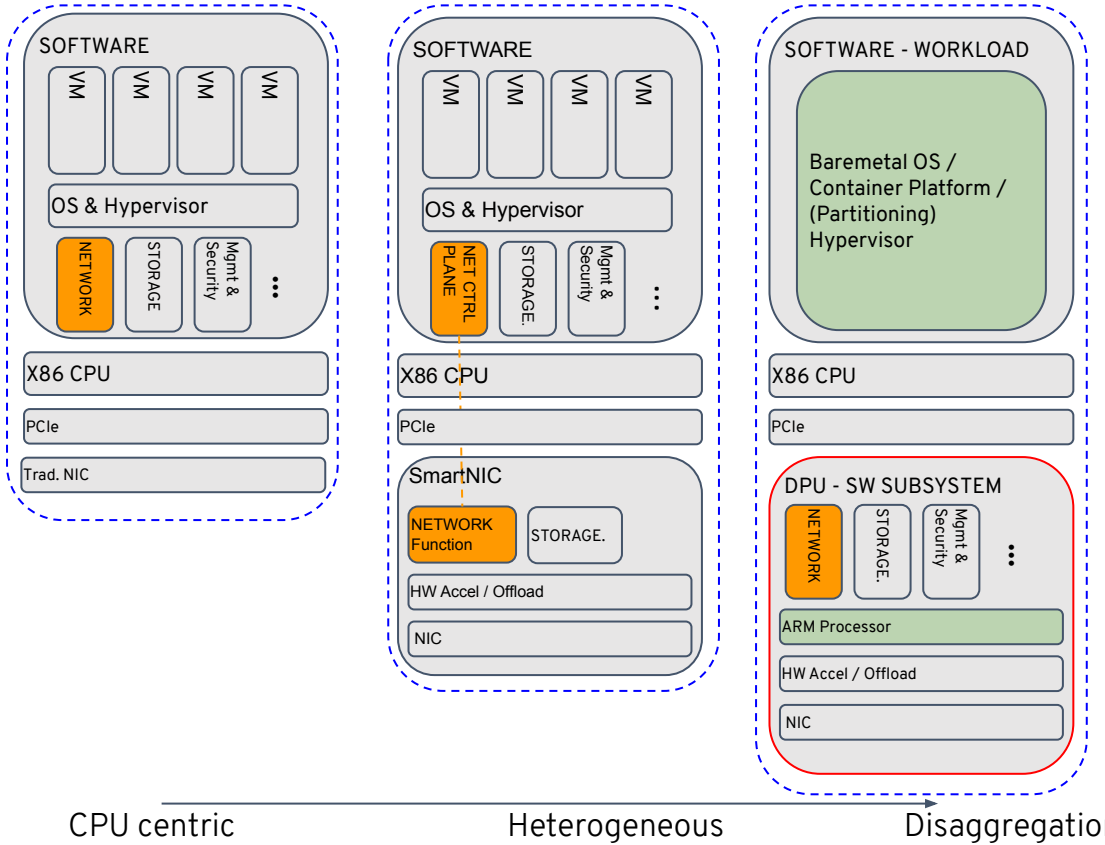


Penalty-free visibility & micro-segmented security

Hybrid cloud computing Architecture Evolution

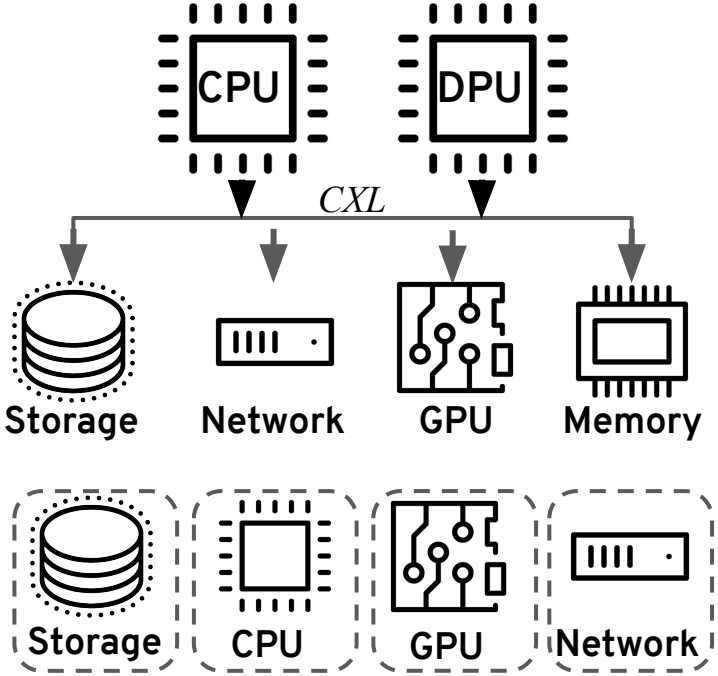
Evolution from **software implemented subsystems** over hardware acceleration and **offloading of specific functions** to the **offloading of complete subsystems** on domain-specific HW with **software defined device function.**

Replaces the **software Hypervisor** with **virtualization** and **isolation** implemented through **software defined device functions** on the smart subsystems running the same components from an open ecosystem on a general purpose OS.



Longer term disaggregation of system components

Growth in complexity of workloads and resource requirements necessitates novel solutions



Today we are taking the first steps on a journey that will redefine the traditional datacenter environment.

Ultimately it will become possible to disaggregate entire classes of system components, such as DRAM and persistent memories from where compute happens.

CPUs and DPUs will become peers(with other agents) using future datacenter interconnect technologies

DPU Use Cases

Use cases for offloading to these subsystems include...

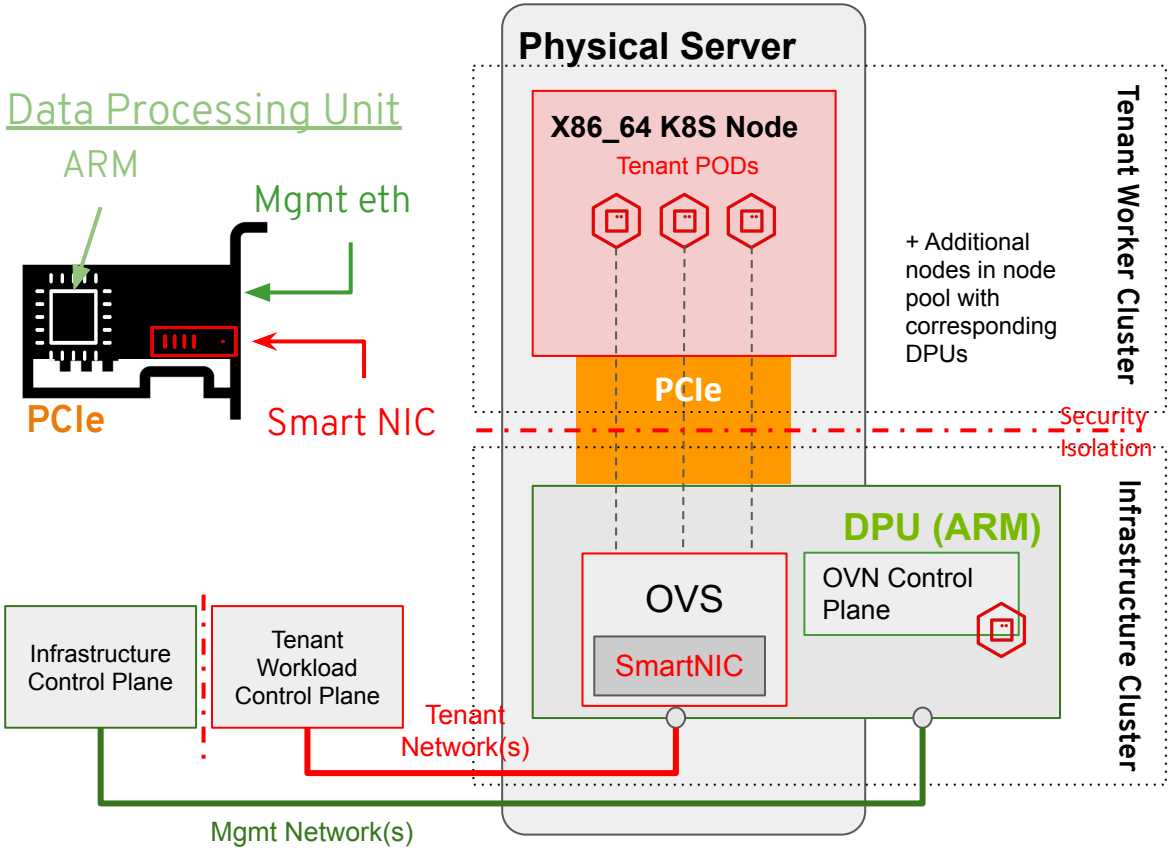
System Services

- Networking / SDN (OVN example)
- Encryption, VPN
- Storage endpoints
- Secure enclaves
- Advanced network security
- Management functions

Application-level offloading

- Stream processing
- Serverless function
- Custom services

Example: OpenShift OVN Offload



Use Case for Workload

- OVN control plane and OVS run on the DPU, using the acceleration and lower-level offloading features on the device.
- OVN on the DPU is isolated from the workload cluster and managed by an infrastructure cluster.
- Networking capabilities are exposed to the workload cluster through a CNI plugin.
- Host sees OVN functions on DPU as SRIOV virtual functions.

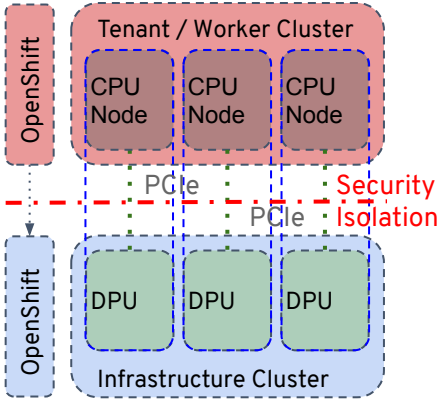
Additional capabilities are built on top of OVN on the DPU:

- VPN / VPC / IPSec incl. Accel / offload.
- Next Gen Firewall.
- Secure enclave.

DPU Deployment Models in each scenario

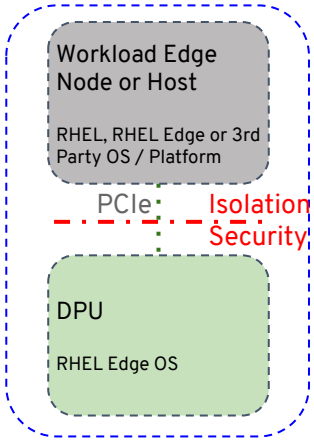
Dynamically Orchestrated

- Dynamically managed in OpenShift.
- Separate management of tenant workload cluster and DPU infrastructure cluster.
- Secure delegation of management.
- Independent life cycles.



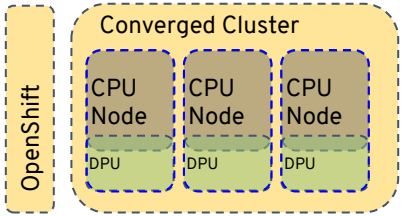
Statically Orchestrated

- Worker nodes and DPUs managed separately.
- RHEL Edge on DPU.
- Tenant workload can be RHEL or 3rd Party Platform.



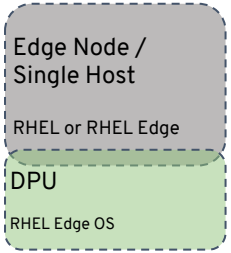
Converged Cluster

- DPUs managed from the workload cluster nodes.
- Single controlplane



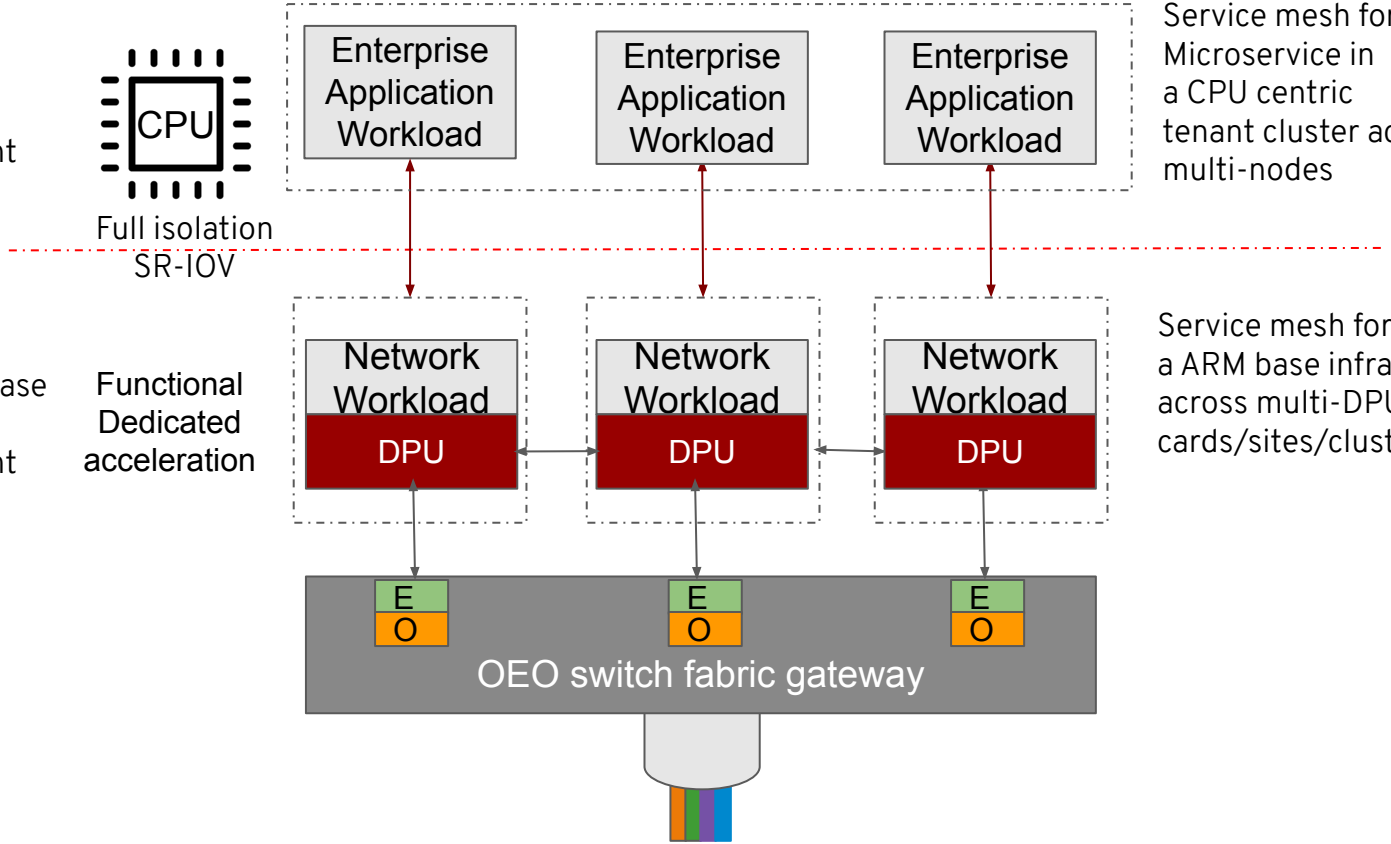
Converged Standalone

- Standalone RHEL / RHEL Edge host with DPU managed from workload host.



In all scenarios DPUs run RHEL (Edge / RHCOS), software defined functions deployed as containers from open ecosystem.

Hybrid computing with DPU and CPU - Isolate Infrastructure & Workload



OpenShift/K8s possibility for Management of Isolated Environments

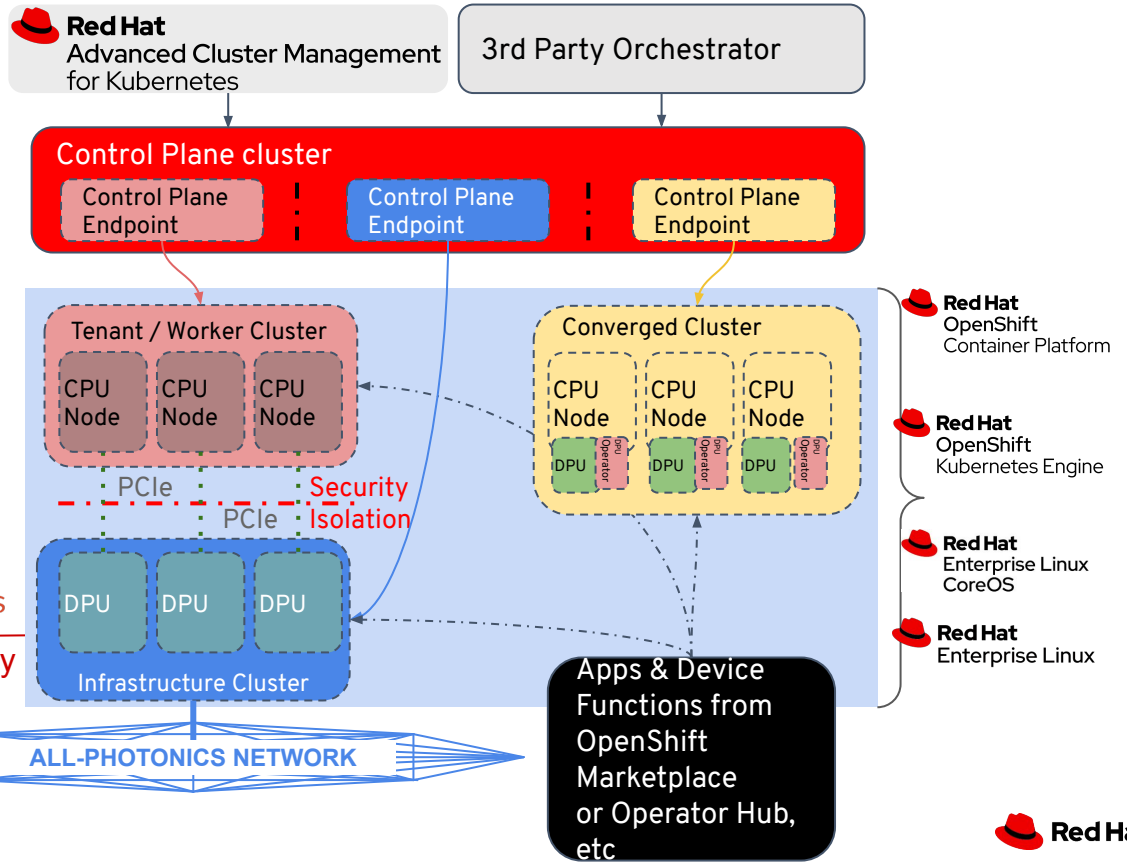
Multi-types of providers, Multi clusters and Heterogeneous clusters for domain-specific hardware

Open Cluster Management

https://github.com/open-cluster-management

Separate control planes for workload and infrastructure clusters allow for clean security isolation and delegation.

Control Plane cluster allows for consolidated deployment of the different control planes on shared back-end infrastructure.



Summary

- Due to end of **Moore's Law,** everyday NICs getting Smarter

There is an emergent change in system architecture from CPU-centricity to independently **intelligent subsystems** with their own **composable specialized compute capabilities** enabling **multi-cluster scale acceleration.**
Benefits include:

- **Enhanced security** through isolation and delegation.
- Cleaner **architectural compartmentalization**.
- **Better performance** through aligned abstractions and locality.
- Improved life cycle management and **stability**.
- Access to a broader **ecosystem** and faster **innovation**.

This leads to a hardware system design that matches the Container & Kubernetes model of orchestrated, compartmentalized services.
It defines the **future datacenter architecture** for the open hybrid cloud.
NVIDIA DPU is first example of hardware accessible to everyone. Other vendors have similar plans.

- Flexible disaggregation design (example scenario)
  - Container workload
    - Enterprise application workload continues deploy on OpenShift/RHEL CoreOS(Fedora CoreOS) in CPU main system
      - TLS can be offloaded in SmartNIC/DPU as needed
    - SP's CNF workloads can be deployed on OpenShift/RHEL edge(Fedora IoT) based ARM platform in DPU(xPU subsystem)
  - Service mesh for microservice
    - Enterprise application workloads in Istio service mesh on OpenShift CPU main system
      - mTLS can be offloaded in SmartNIC/DPU as needed
    - CNF workloads in service mesh on ARM in DPU


Thank you!
Red Hat