Service Function Chaining, DPUs, and Shared OVN

Tim Rozet

What is Service Function Chaining (SFC)?

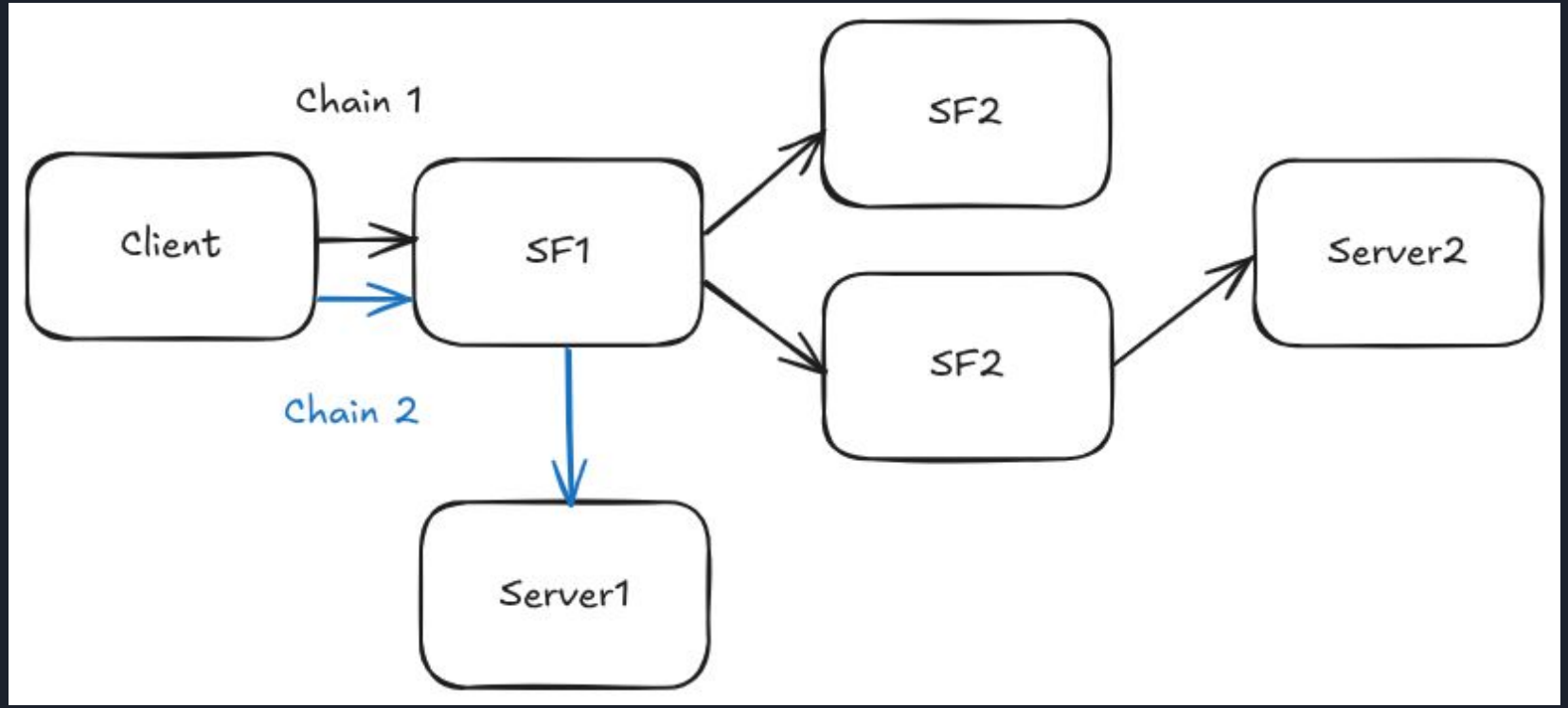
From https://datatracker.ietf.org/doc/html/rfc7665 -

```
The definition and instantiation of an ordered set of service functions and
subsequent "steering" of traffic through them is termed Service Function
Chaining (SFC)
```

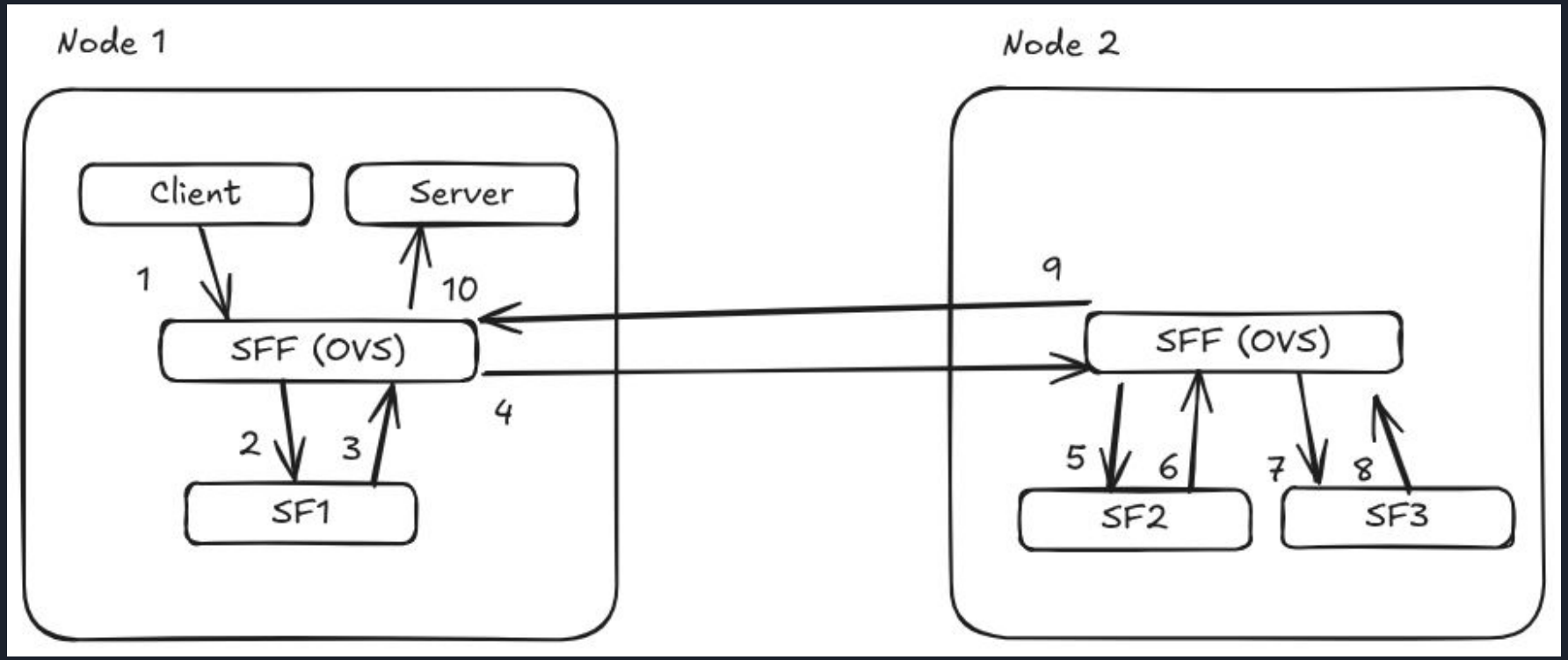
What is Service Function Chaining (SFC)?

- Logical sequencing of Service Functions (e.g., FW → DPI → LB → Router).
- Classifiers match on packet headers to be redirected to chains
- Centralized control plane defines service paths; data plane enforces them
- Can span multiple nodes or domains
- Service Functions (SFs) may be aware of the SFC
- Allows for load balancing across multiple instances of the same network service.
- Packet headers may be modified as chain is traversed

SFC Example



SFC Example - Multiple Nodes / Service Function Forwarders (SFFs)



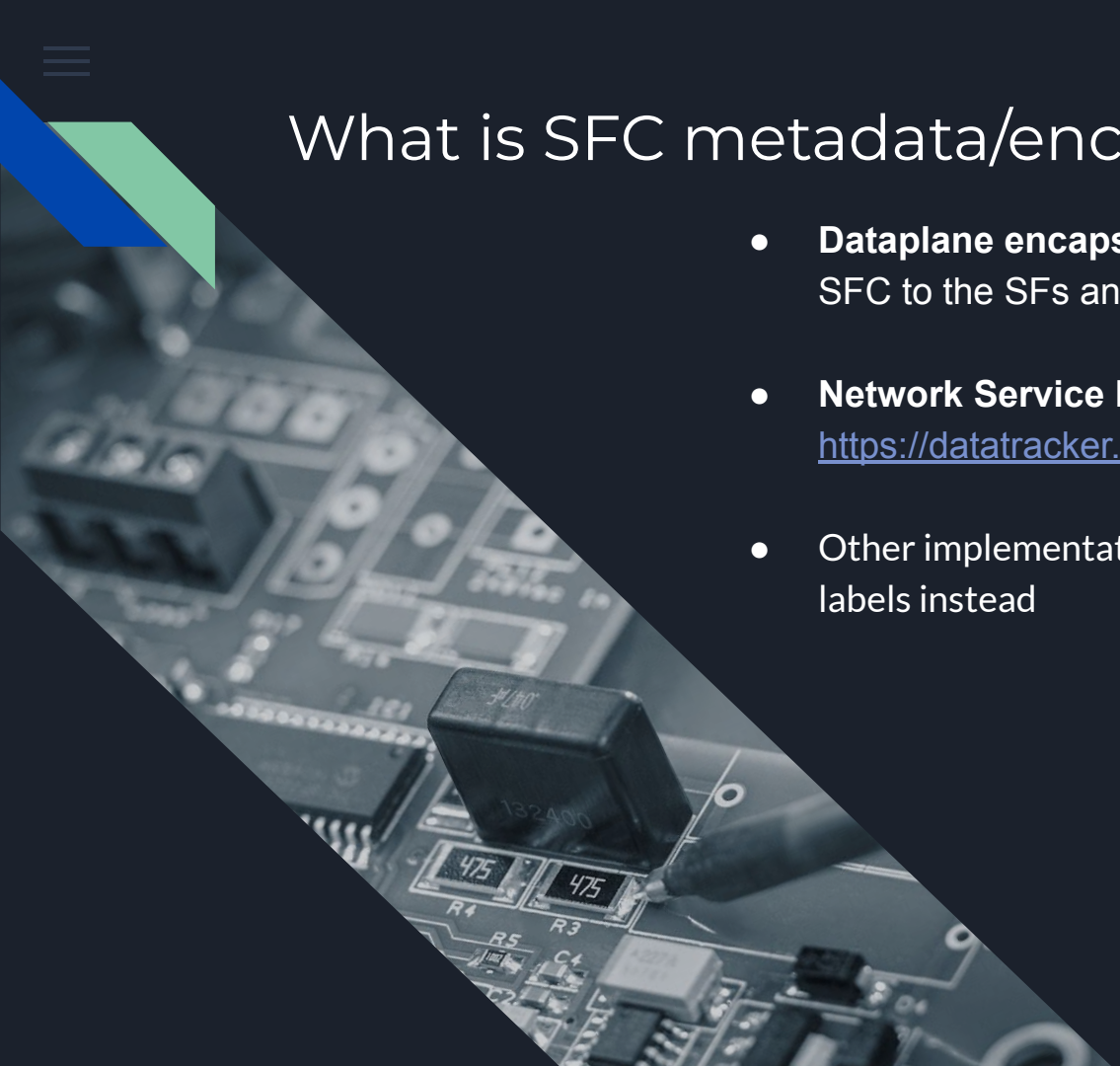
Why does SFC matter?

- **Dynamic service insertion**, scaling, and orchestration.
- Overcomes limitations of the OpenFlow dataplane
- Enables users to insert their own 3rd party network function
- Per-flow/app/tenant network pipelines

What is SFC Symmetry?

- **Reverse/Reply path** for SFC traffic should follow the same reverse order of SFs
- If symmetrical Chain A->B->C, should have reverse path of C->B->A
- Symmetry can be achieved by implicitly creating an inverse classifier and a second reverse SFC, when a classifier and SFC are created
- Symmetry can also be achieved using stateful tracking like conntrack to restore the reverse path

What is SFC metadata/encapsulation?

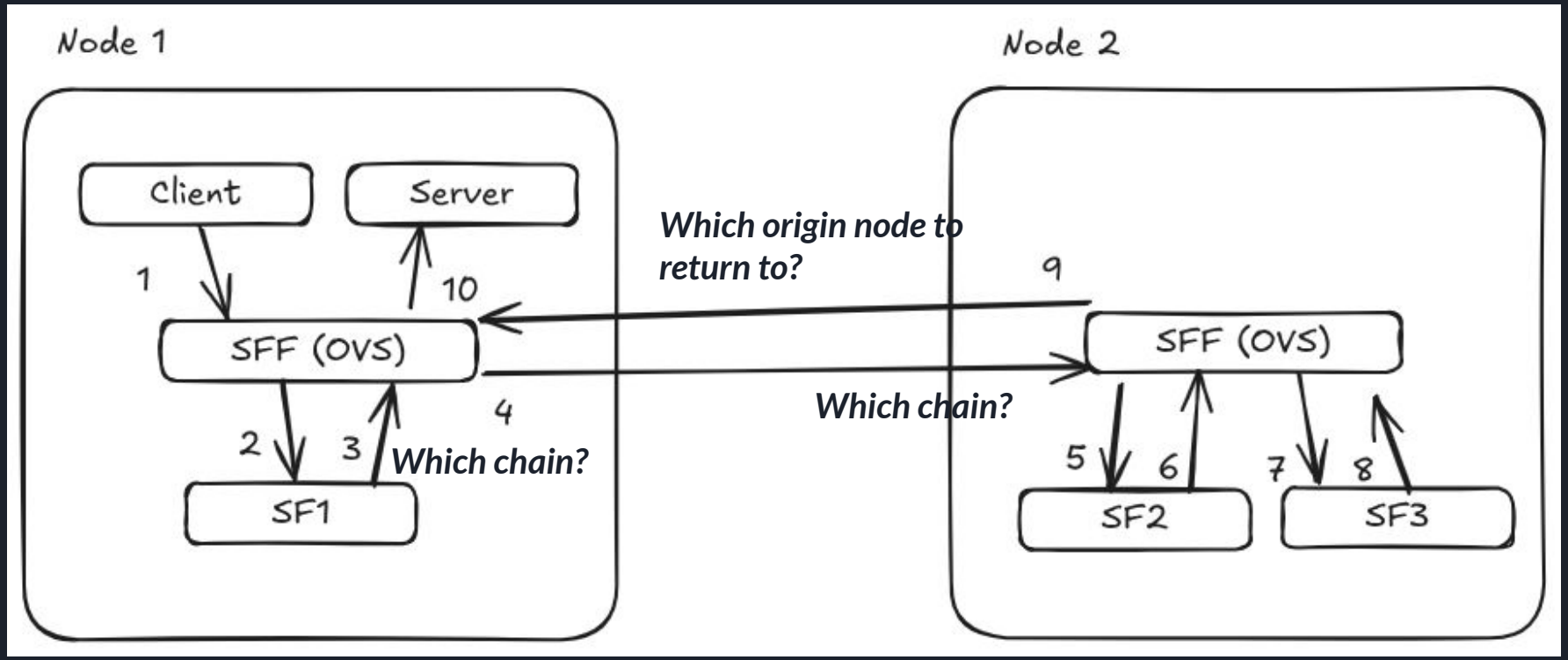


- **Dataplane encapsulation** — carries information about the SFC to the SFs and SFFs.
- **Network Service Header (NSH)** — https://datatracker.ietf.org/doc/html/rfc8300
- Other implementations like **OpenStack** ml2+OVS use **MPLS** labels instead

Why is metadata necessary?

- **Which chain (service path)** the packet belongs to — the *Service Path Identifier* (SPI).
- **Where in the chain** it currently is — the *Service Index* (SI).
- **Who is sending through the chain** — Tenant ID, specific flow information, original node context
- Without metadata SFF's are considered to be proxying for non-SFC aware SFs

SFC Example - Without Metadata



How can we adapt SFC to OVN?

- Initially only support SFC-unaware SFs (no NSH)
- Use Geneve to pass metadata between nodes
- Chain can span multiple nodes without packet modification
- Chain must be contained to a node for packet modification support
- An SF may only be part of a single chain
- Allows for load balancing across multiple instances of the same network service
- Supports HW Offload

Extending the Network Function OVN Design

- **Network_Function_Group** — Introduce new **mode** value “**load-balance**”. Allows multiple SFs of the same type to be load balanced on.
- **ACL** — Introduce new reference to a new table **Service_Function_Chain**
- New **Service_Function_Chain table** that holds references to NFGs

See https://mail.openvswitch.org/pipermail/ovs-dev/2025-June/424080.html for more the full discussion.

```
"Service_Function_Chain": {
 "columns": {
  "name": {"type": "string"},
  "groups": {
   "type": {
    "key": {"type": "uuid", "refTable": "Network_Function_Group"},
    "min": 1, "max": "unlimited"
   }
  },
  "mode": {
   "type": {
    "key": {"type": "string",
          "enum": ["set", ["local", "immutable-global"]]}
   }
  },
  "id": {
   "type": {"key": {"type": "integer", "minInteger": 0, "maxInteger": 65535}}
  },
  "external_ids": {
   "type": {"key": "string", "value": "string",
         "min": 0, "max": "unlimited"}
  }
 },
 "isRoot": true
}
```

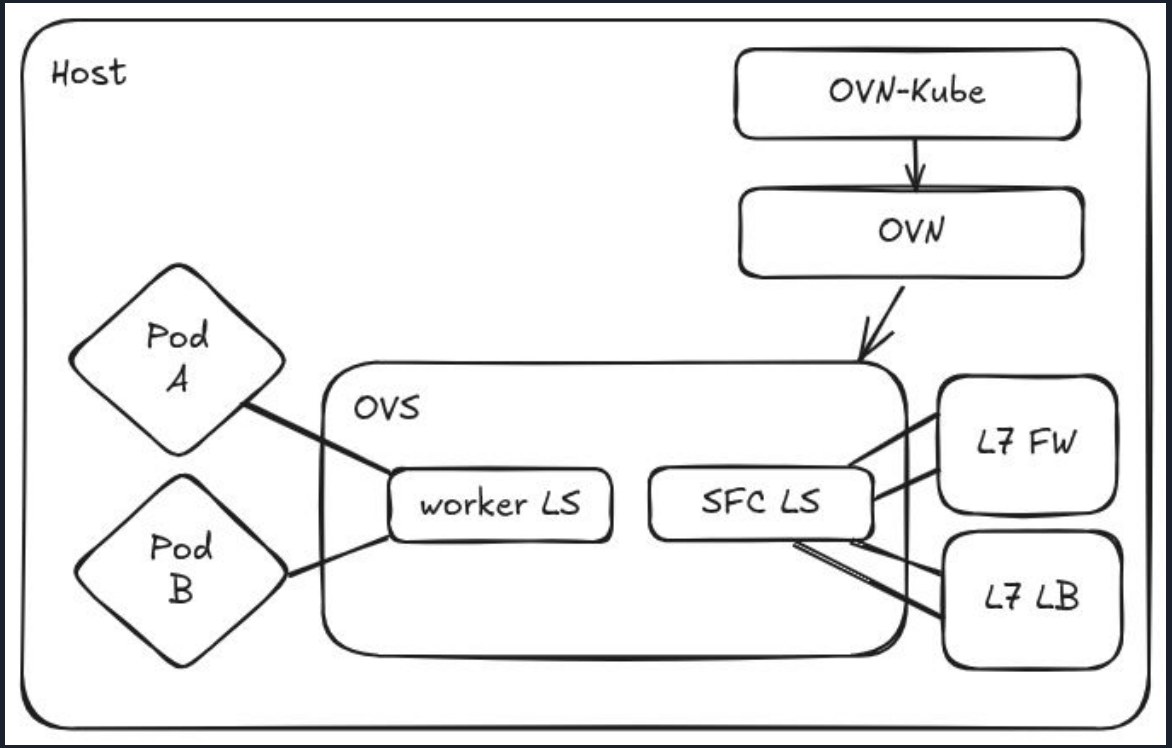
Service_Function_Chain Details

- Mode **Local** will not use conntrack, and will allow for packet modification within a node. OVN will check that all NFs that are part of the SFC are bound to the local chassis. Symmetry is not implicit, and the CMS needs to explicitly create a reverse classifier and chain.

- Mode **Immutable-Global** will use conntrack, imply symmetry through conntrack, allow for the SFC to cross nodes, but will not support packet modification. A SFC may still be only on the local node, and use this mode.

- In the future, a new mode may be added if/when NSH is added to allow for **Mutable-Global** mode.

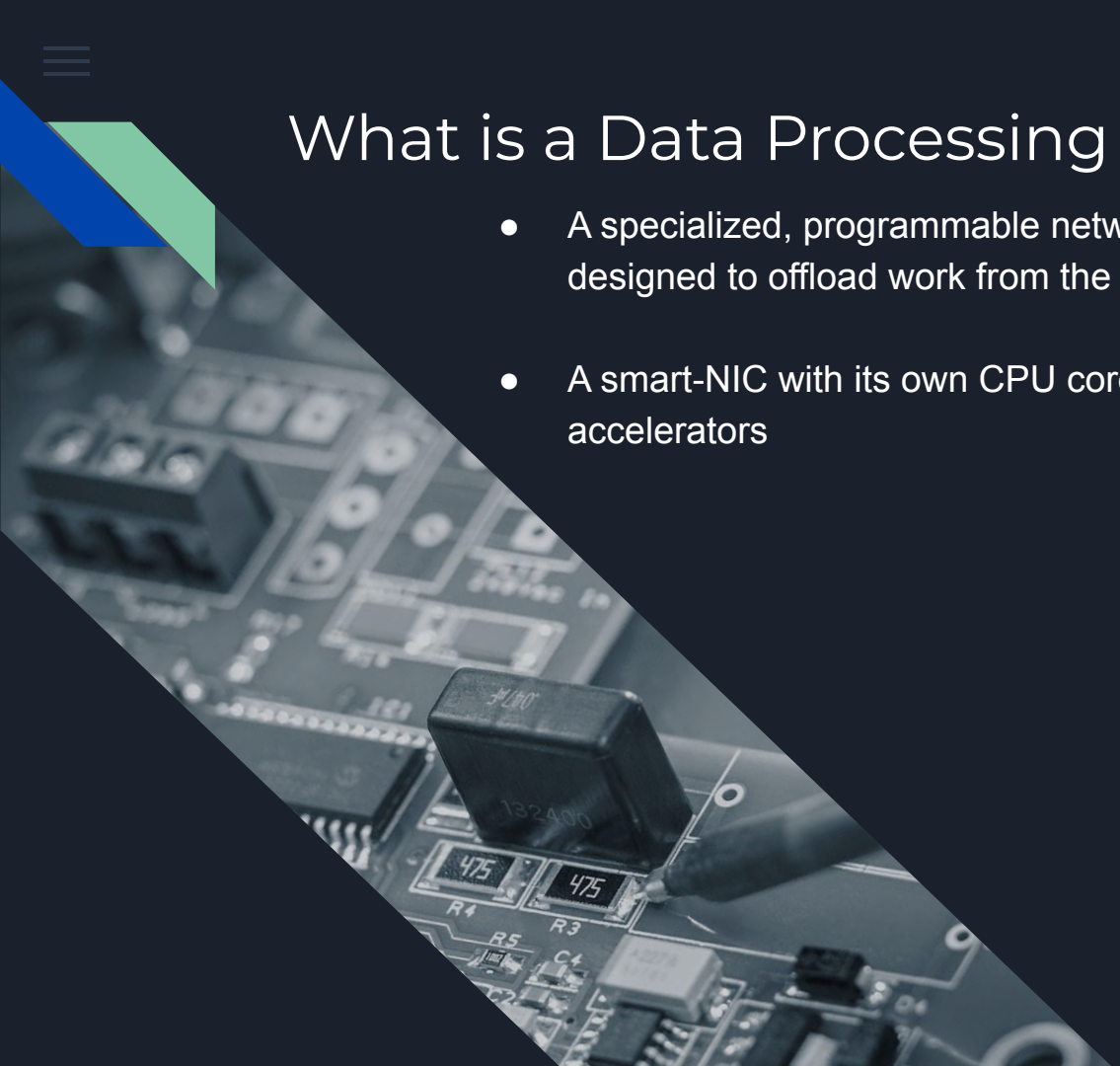
SFC in OVN-Kubernetes



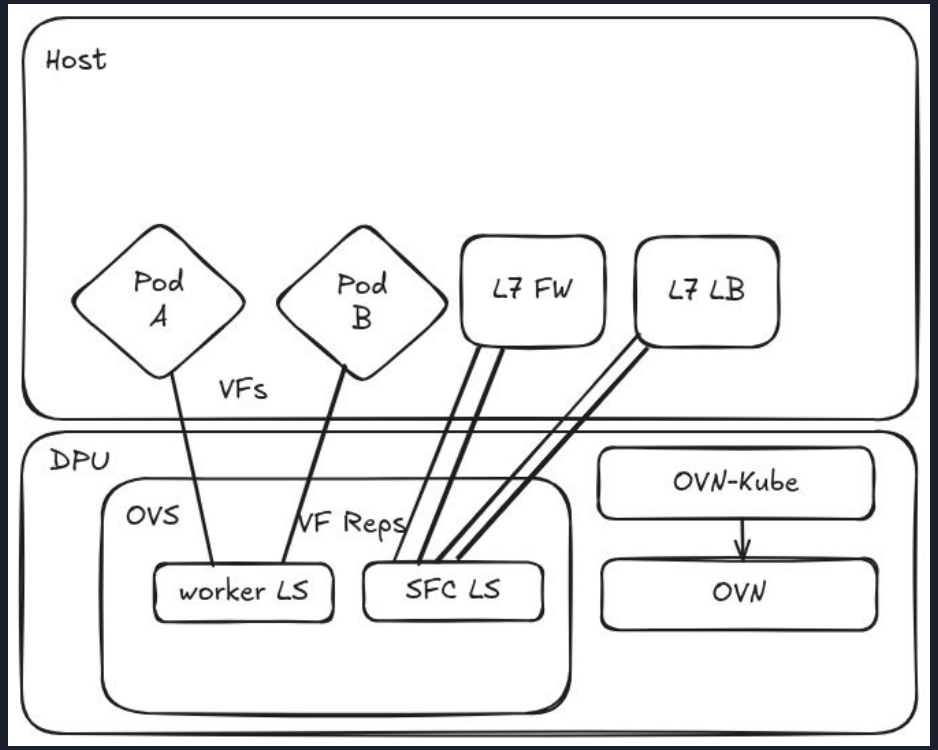
What is a Data Processing Unit (DPU)?

- A specialized, programmable networking + security + storage accelerator designed to offload work from the host CPU

- A smart-NIC with its own CPU cores, memory, OS, and hardware accelerators



OVN-Kubernetes with a DPU

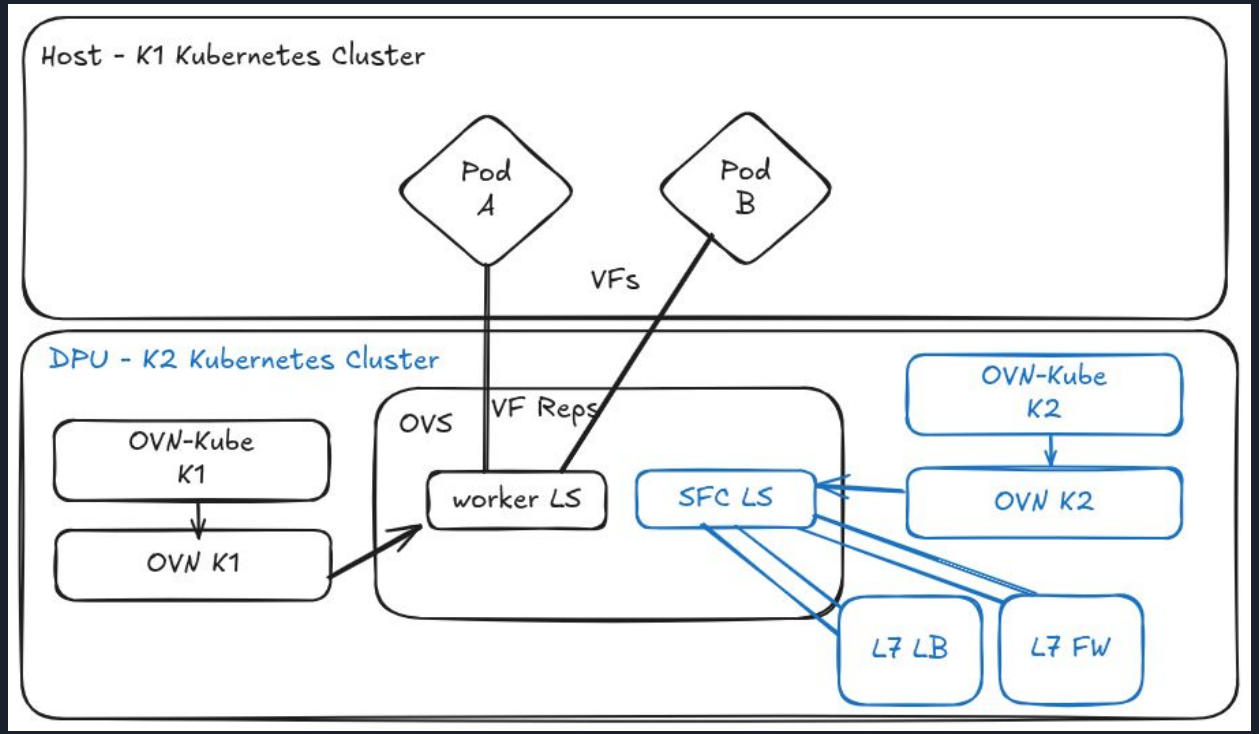


DPU Advantages

- Provides Hardware Offload, maximum throughput
- Reduces resource consumption on Host
- Moves the SDN control-plane into the DPU, improves security

What if we were to also install Kubernetes on the DPU and use OVN-Kubernetes/OVN there?

Two Kubernetes Clusters DPU Model



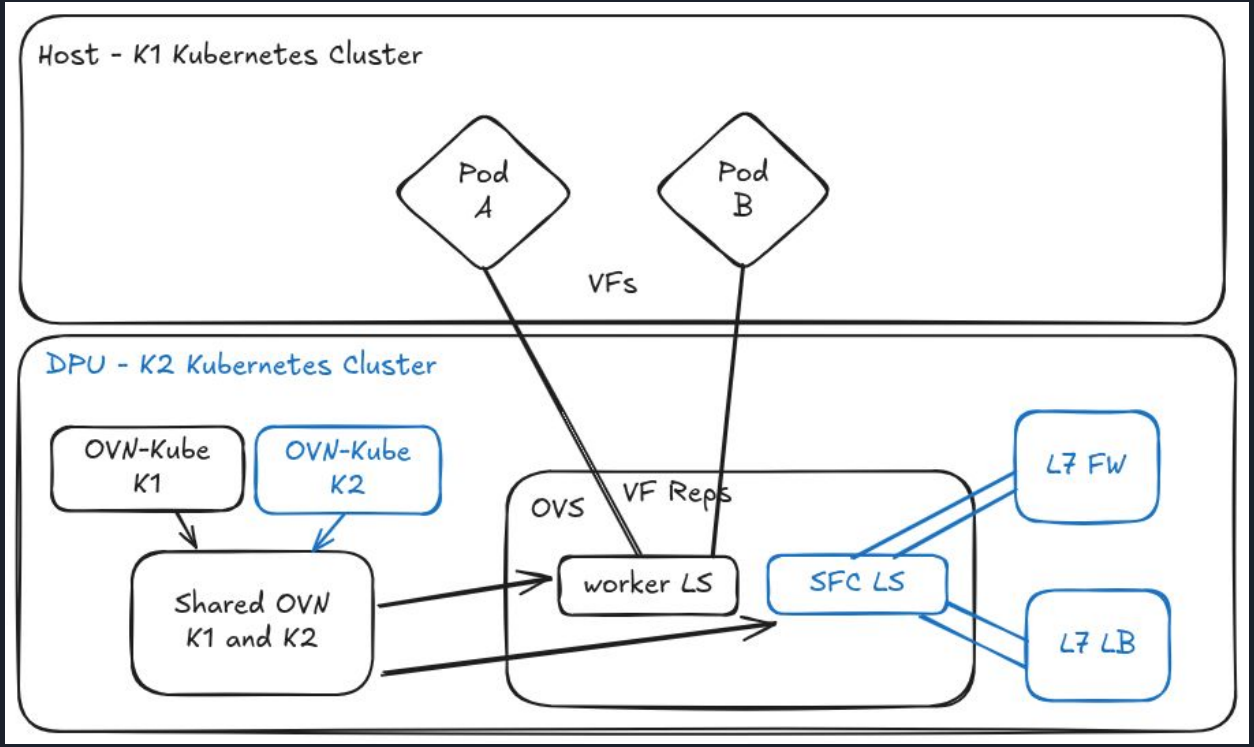
How can I get K2 to classify K1 traffic into the DPU chain?

2 Cluster DPU Advantages

- As a DPU admin, I can now enforce network security at the edge with SFC intercepting packets from the Host SDN!

What if we were combine OVNs into a single "Shared OVN"?

Two Kubernetes Clusters DPU Shared OVN



2 Cluster Shared OVN Advantages

- **DPU K2** cluster now is able to have **Admin RBAC** and modify all entities within the Host cluster, allowing it to inject ACLs as SFC classifiers.
- **K1** cluster is only allowed via **RBAC to modify its own cluster contents**, and **not** the ACLs configured by K2.
- Is OVN RBAC secure enough?
- Does running a shared OVN model actually reduce resource consumption?

Next Steps

- Need to work with OVN community to drive this forward and expand on the Network Function Insertion work. **Need your help**!
- Targeting single cluster SFC with OVN-Kubernetes first, non-shared two cluster second.
- Will do perf/scale testing to see whether shared OVN is worth it.

Thank you!