



Centralized Composable HPC Management with the OpenFabrics Management Framework

Michael Aguilar

Phil Cayton (Intel), Christian Pinto (IBM), Russ Herrell (HPE)

IPDPS/COMPSYS23

St. Petersburg, Florida, USA

May 19, 2023



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND2023-03887A

Contributors to the OFMF

The goal of the OFMF is to enable interoperability through common interfaces to enable client Managers to efficiently connect workloads with resources in a complex heterogenous ecosystem, without having to worry about the underlying network technology.



What are Composable Disaggregated HPC Systems?



Advantages of Composability over Current HPC Architectures

- Mitigate Resource Overprovisioning
- Reduce Energy Consumption and cooling costs
 - 4% of the World's Energy Consumption Is input into Datacenters
(<https://www.energy.gov/eere/buildings/data-centers-and-servers>)
- Localized Provisioning where resources are needed



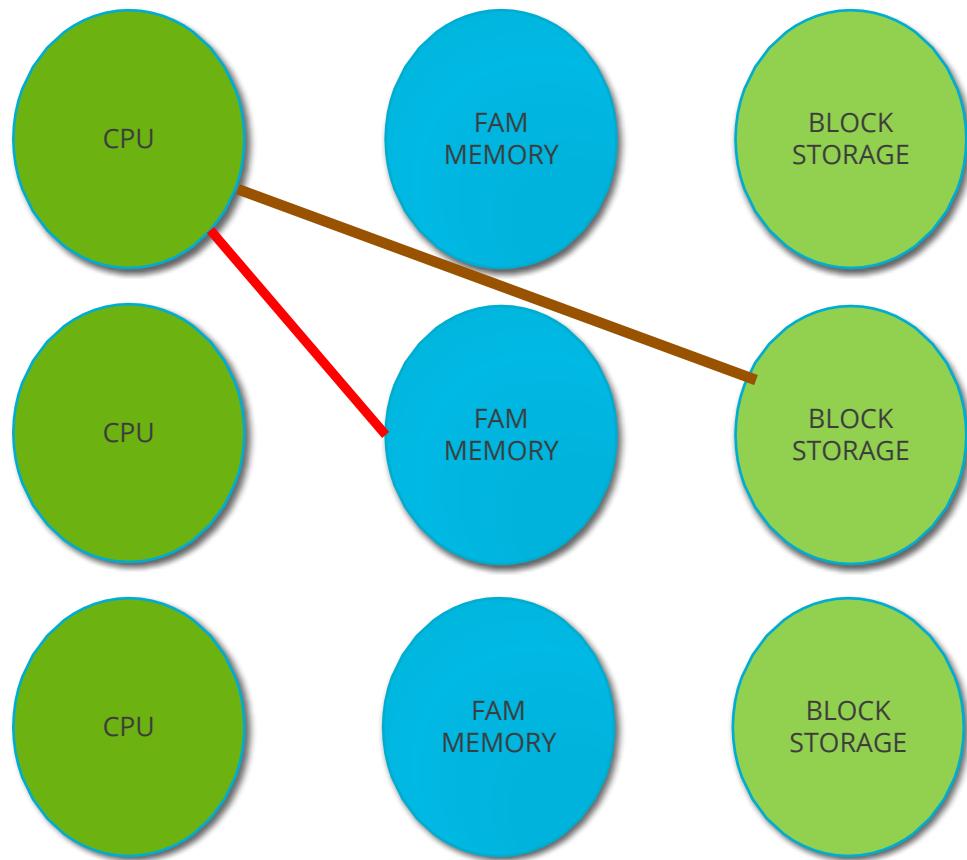
...



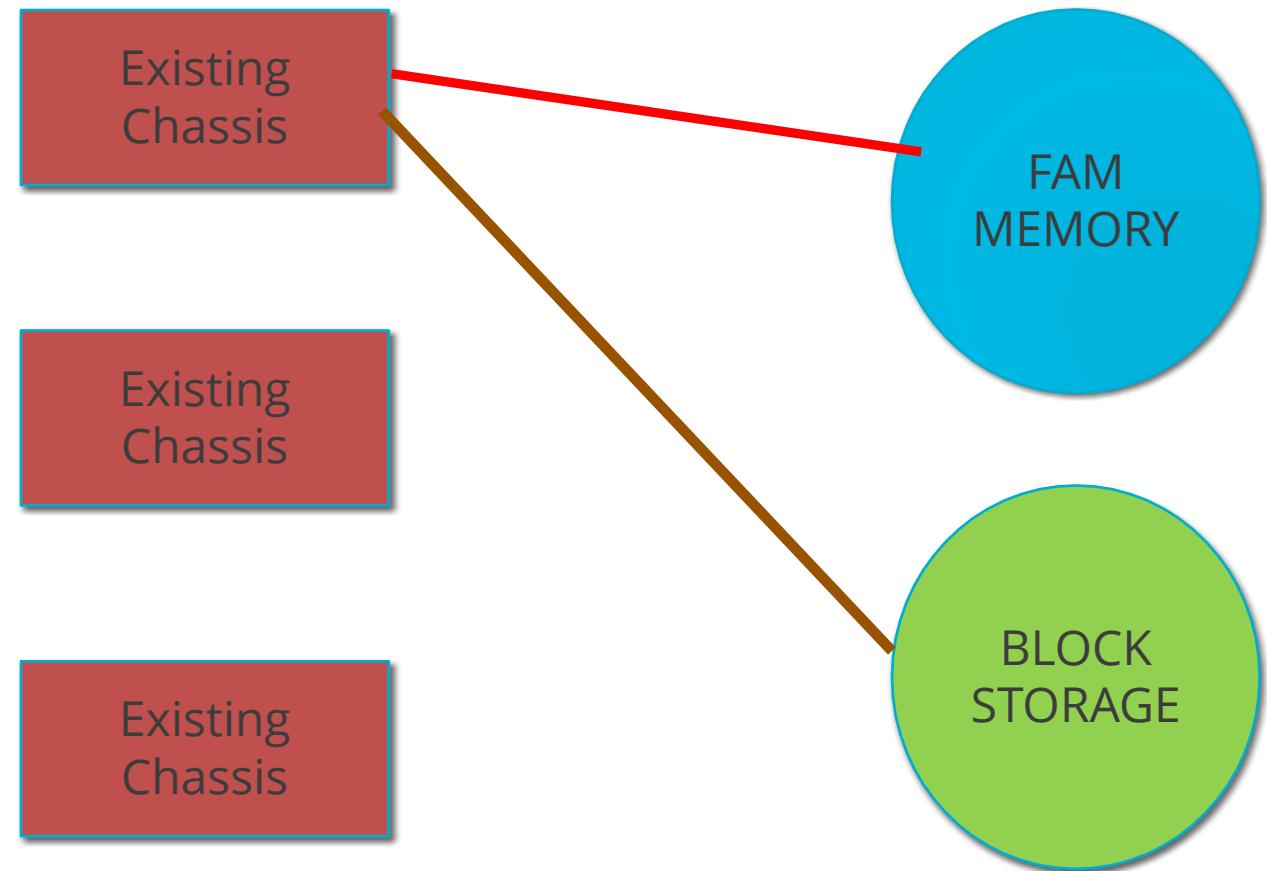
What are Composable Disaggregated HPC Systems?

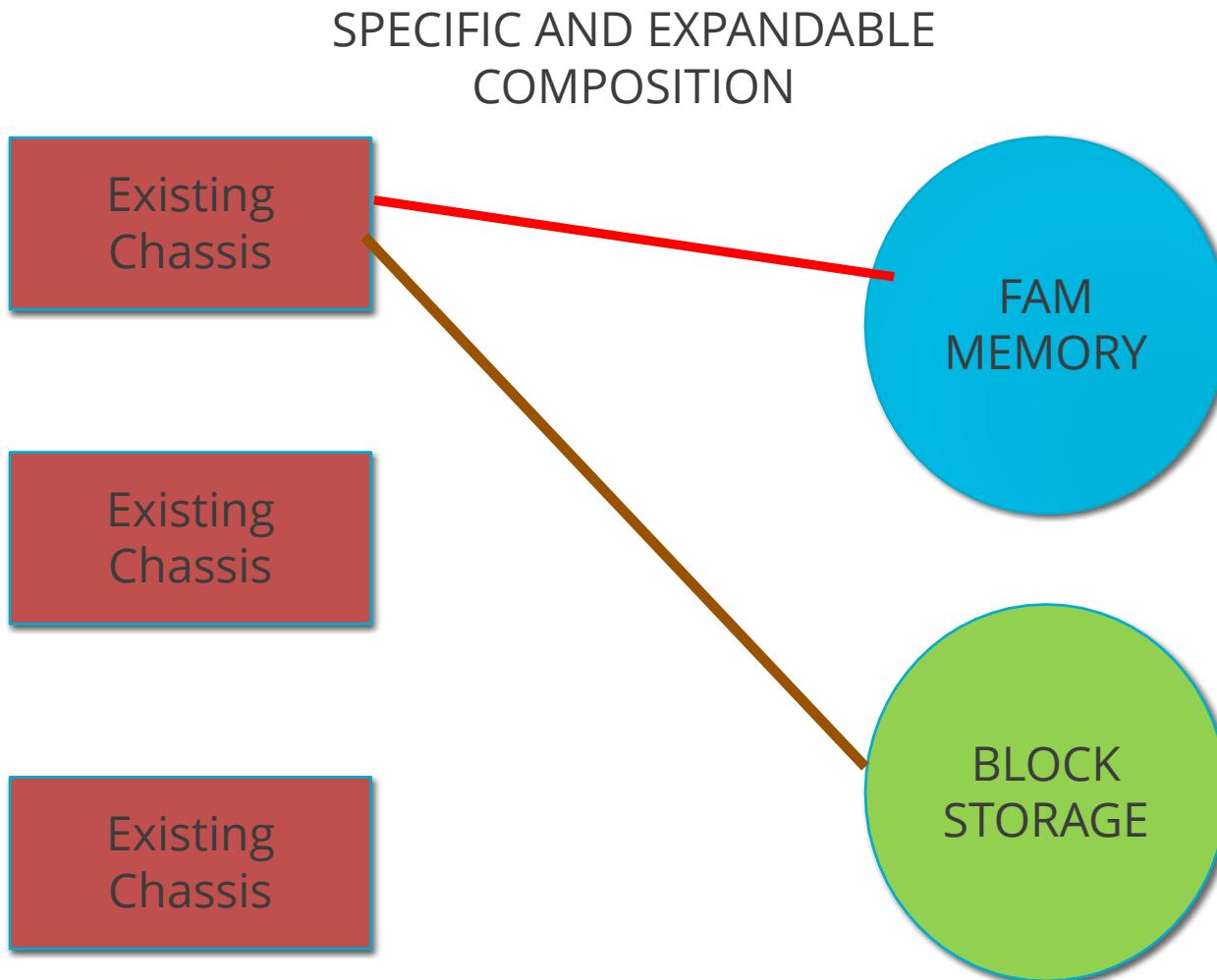


SPECIFIC OR CONSTRAINED COMPOSITION



EXPANDABLE COMPOSITION



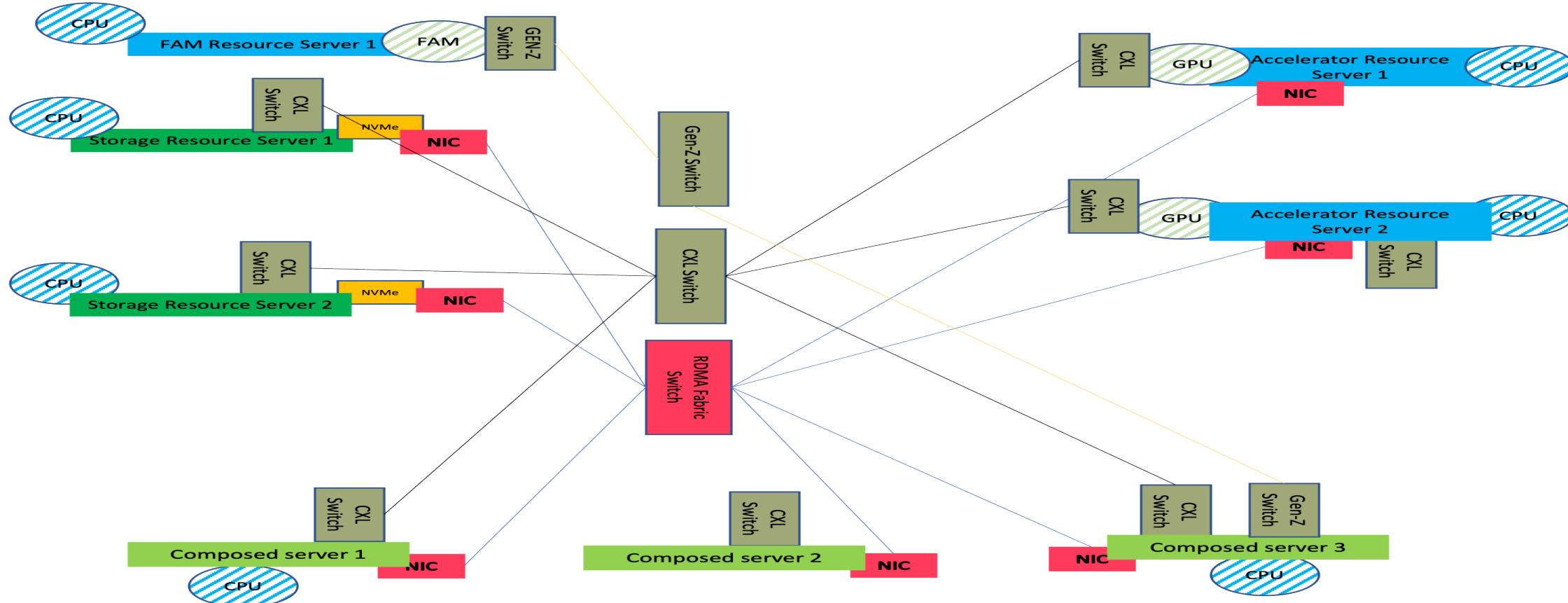


- Pools can be used to augment memory with direct-addressable devices and block devices
 - ccNUMA for the FAM memory
 - NVMeoF for the Block storage

What are Composable Disaggregated HPC Systems?



Homogeneous HPC Systems become Heterogeneous HPC Systems

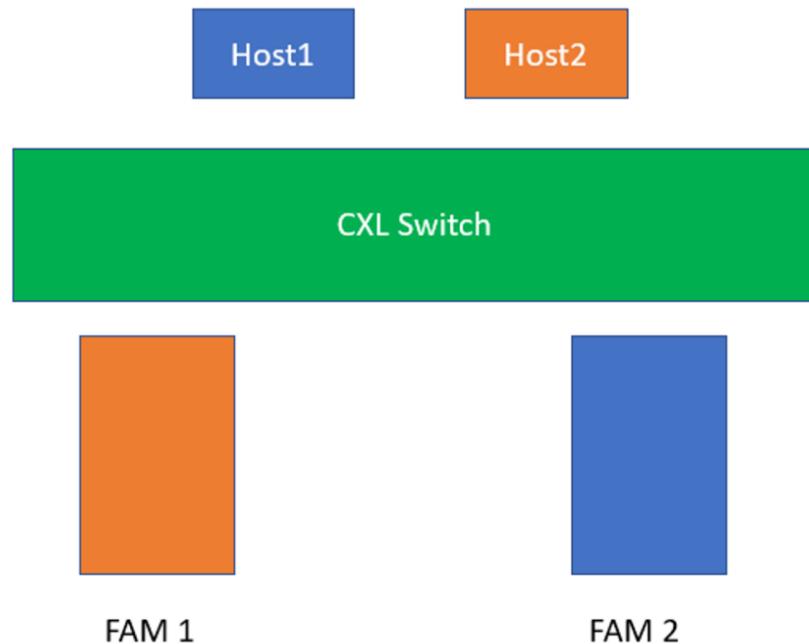


What are Composable Disaggregated HPC Systems?

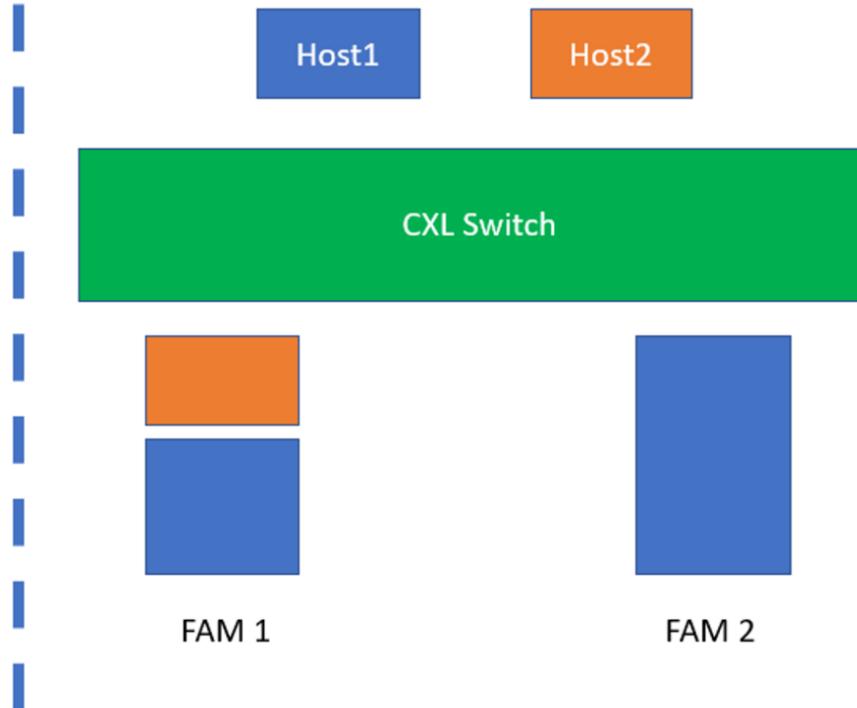


CDI HPC Nodes and Fabric Attached Memory

Single logical device case
(color coding denotes binding)



Multi logical device case



OpenFabrics Management Framework for Composable Distributed Systems



We need a centralized control infrastructure to manage our disaggregated compositions and decompositions

We want:

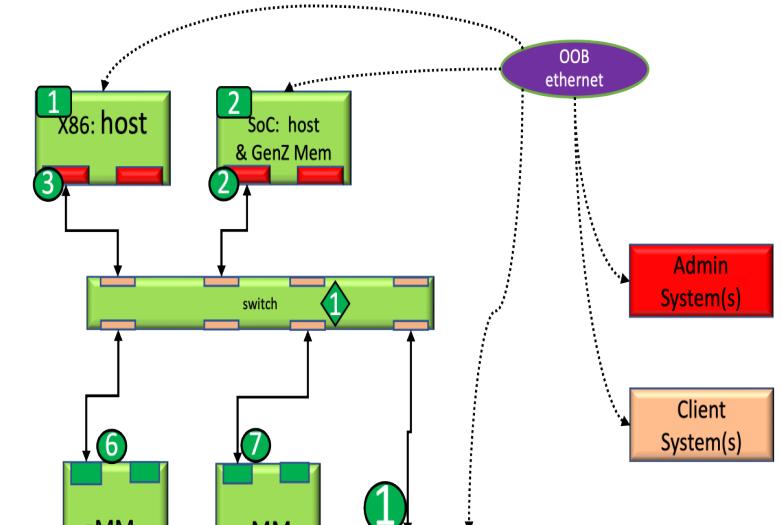
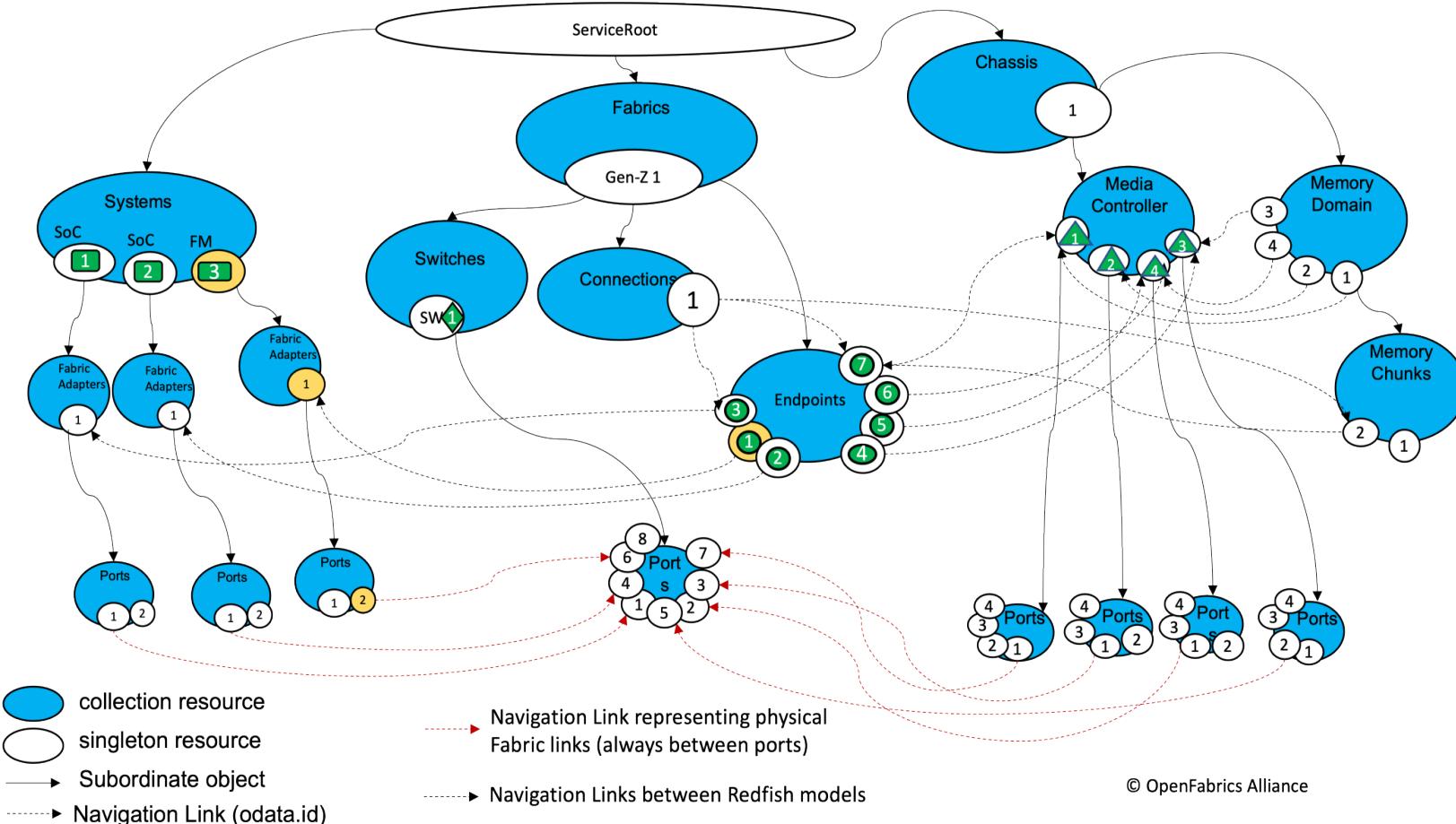
- To be able to control Composable Disaggregated Infrastructure (CDI) in an HPC Architecture
- Redfish Representation of a Composable Disaggregated Infrastructure Components. Redfish provides us with structures that we can use to store and read component information.
- Swordfish Representation of Storage Pools, Volumes, and Endpoint Groups
- A centralized service that can provide current up-to-date information on CDI compositions and cluster state information
- A centralized service that can abstractly manage our CDI compositions

Redfish Representation of a Composable Disaggregated Infrastructure

Redfish mapping of a simple HPC system



Simple Gen-Z Linux System Redfish Tree: Physical Objects, Endpoints, and Port linkages



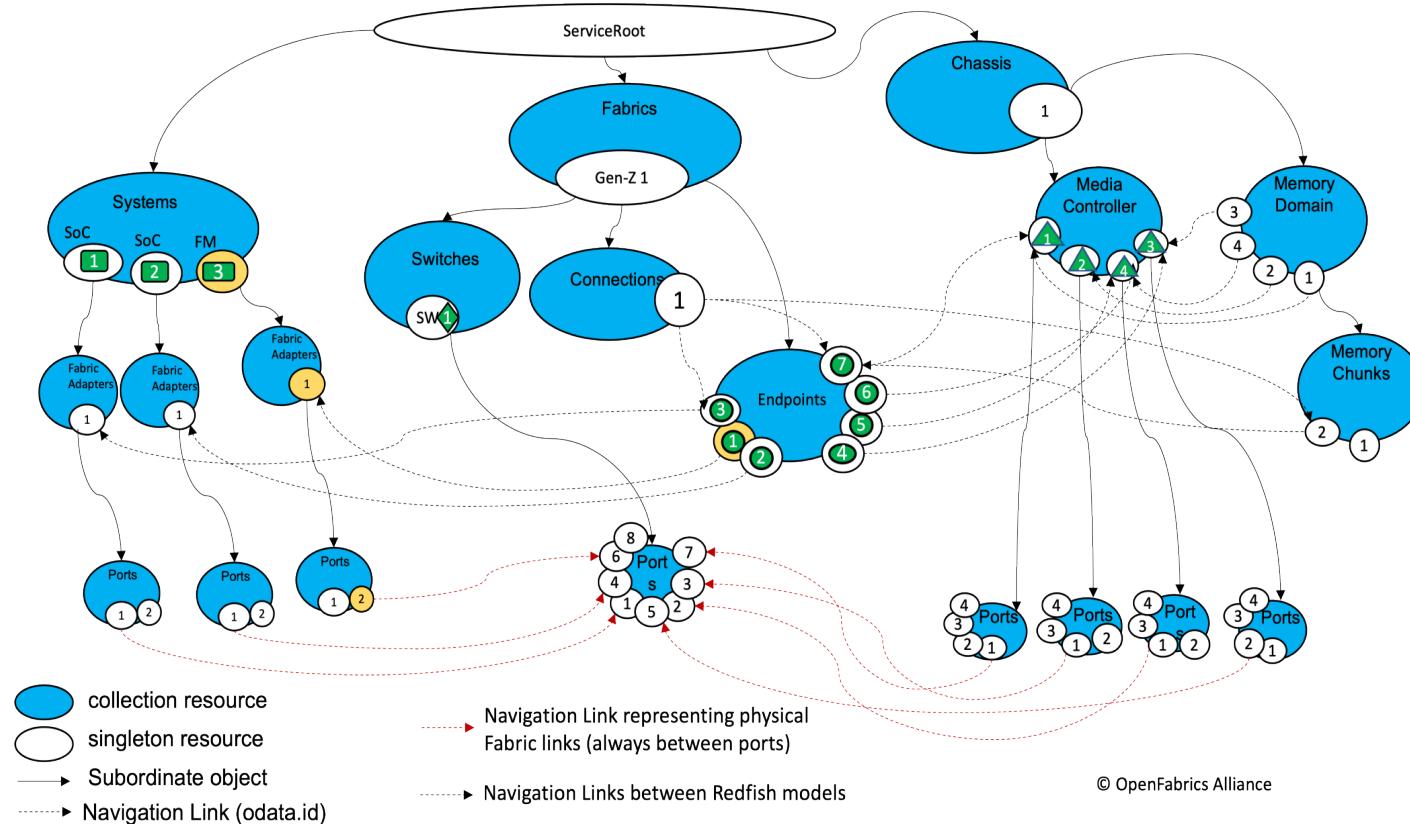
- 1 Redfish Endpoint ID
- 1 Redfish System ID
- 1 Redfish Switch ID
- ▲ Redfish Media Ctrl ID

Redfish Representation of a Composable Disaggregated Infrastructure

Redfish mapping of a simple HPC system



Simple Gen-Z Linux System Redfish Tree: Physical Objects, Endpoints, and Port linkages



```
$> curl -X GET -H "Content-Type: application/json"
http://ofmfserv:5000/redfish/v1/Fabrics
```

```
{
  "@odata.type": "#FabricCollection.FabricCollection",
  "Name": "Fabric Collection",
  "Members@odata.count": 2,
  "Members": [
    {
      "@odata.id": "/redfish/v1/Fabrics/NVMeoF"
    },
    {
      "@odata.id": "/redfish/v1/Fabrics/Ethernet"
    }
  ],
  "@odata.id": "/redfish/v1/Fabrics"
}(Swordfish)
```

```
curl -X POST -H "Content-Type: application/json" -d @fabric_connection.json
http://ofmfserv:5000/redfish/v1/Fabrics/CXL
```

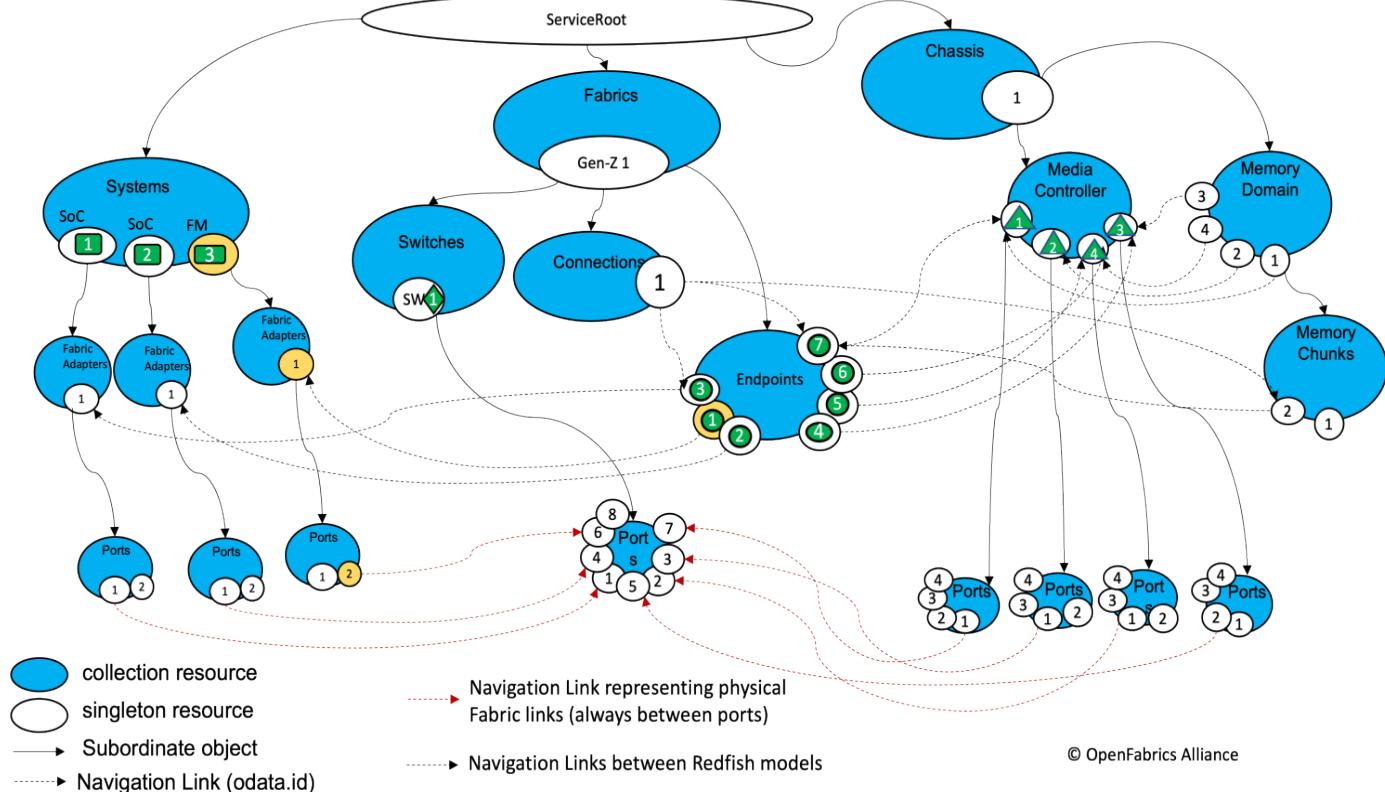
Warning: Couldn't read data from file "fabric_connection.json", this makes an empty POST.

```
{
  "@odata.id": "/redfish/v1/Fabrics/CXL",
  "@odata.type": "#Fabric.v1_3_CXL.Fabric",
  "Id": "CXL",
  "Name": "Fabric"
```

Redfish Representation of a Composable Disaggregated Infrastructure

Redfish mapping of a simple HPC system

Simple Gen-Z Linux System Redfish Tree: Physical Objects, Endpoints, and Port linkages



```
curl -X GET -H "Content-Type: application/json"  
http://ofmfserv:5000/redfish/v1/Fabrics/NVMeoF/Endpoints/Initiator1
```

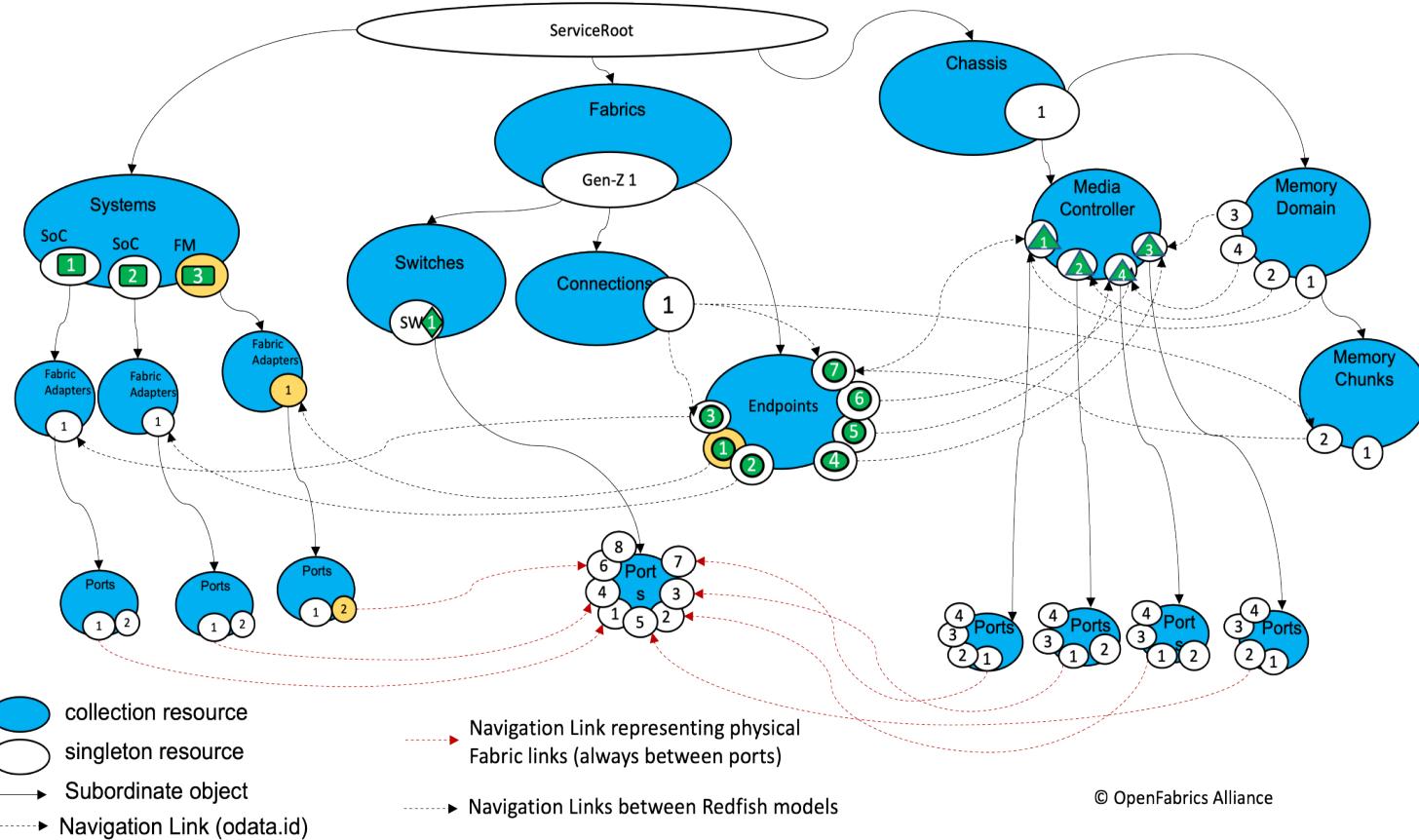
```
{  
  "@odata.type": "#Endpoint.v1_7_0.Endpoint",  
  "Id": "Initiator1",  
  "Name": "NVMe-oF Initiator (Host)",  
  "EndpointProtocol": "NVMeOverFabrics",  
  "Identifiers": [  
    {  
      "DurableName": "host.corp.com:nvme:nvm-subsys-sn-4635",  
      "DurableNameFormat": "NQN"  
    }  
  ],  
  "ConnectedEntities": [  
    {  
      "EntityType": "NetworkController",  
      "EntityRole": "Initiator"  
    }  
  ],  
  "IPTTransportDetails": [  
    {  
      "TransportProtocol": "Ethernet",  
      "IPv4Address": {  
        "Address": "10.3.5.205"  
      },  
      "Port": 13244  
    }  
  ],  
  "Links": {  
    "Connections": [  
      {  
        "@odata.id": "/redfish/v1/Fabrics/NVMeoF/Connections/1"  
      }  
    ]  
  }  
}
```

Redfish Representation of a Composable Disaggregated Infrastructure

Redfish mapping of a simple HPC system



Simple Gen-Z Linux System Redfish Tree: Physical Objects, Endpoints, and Port linkages



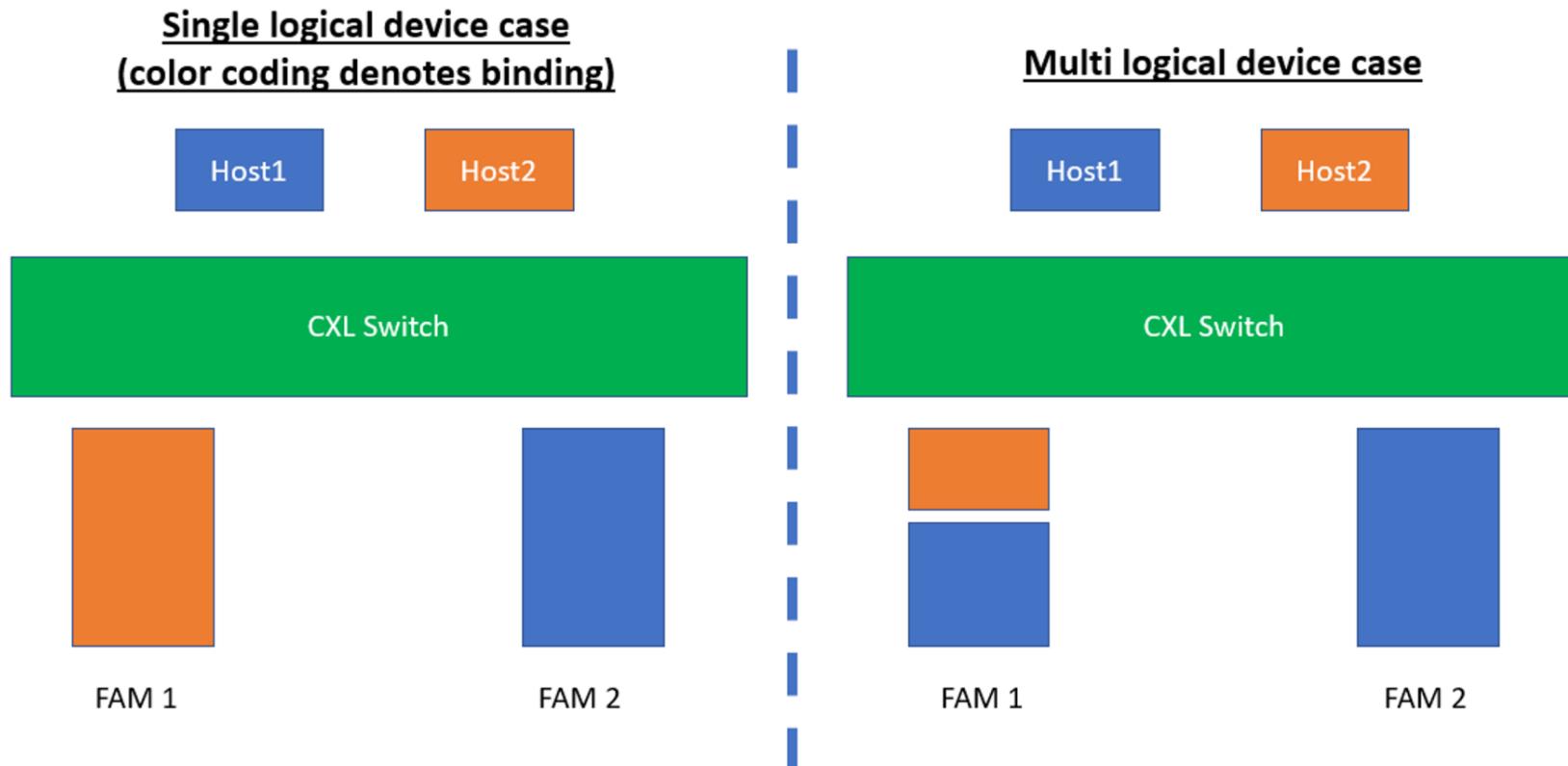
```

curl -X GET -H "Content-Type: application/json"
http://ofmfserv:5000/redfish/v1/Fabrics/NVMeoF/Connections/1
{
  "@odata.type": "#Connection.v1_0_0.Connection",
  "@Redfish.ReleaseStatus": "WorkInProgress",
  "Id": "1",
  "Name": "Host Connection 1",
  "Description": "Connection info for host 1",
  "ConnectionType": "Storage",
  "VolumeInfo": [
    {
      "AccessCapabilities": [
        "Read",
        "Write"
      ],
      "Volume": {
        "@odata.id": "/redfish/v1/Storage/IPAttachedDrive1/Volumes/SimpleNamespace"
      }
    },
    {
      "AccessCapabilities": [
        "Read",
        "Write"
      ],
      "Volume": {
        "@odata.id": "/redfish/v1/Storage/IPAttachedDrive2/Volumes/SimpleNamespace"
      }
    }
  ],
  "@odata.id": "/redfish/v1/Fabrics/NVMeoF/Connections/1"
}
  
```

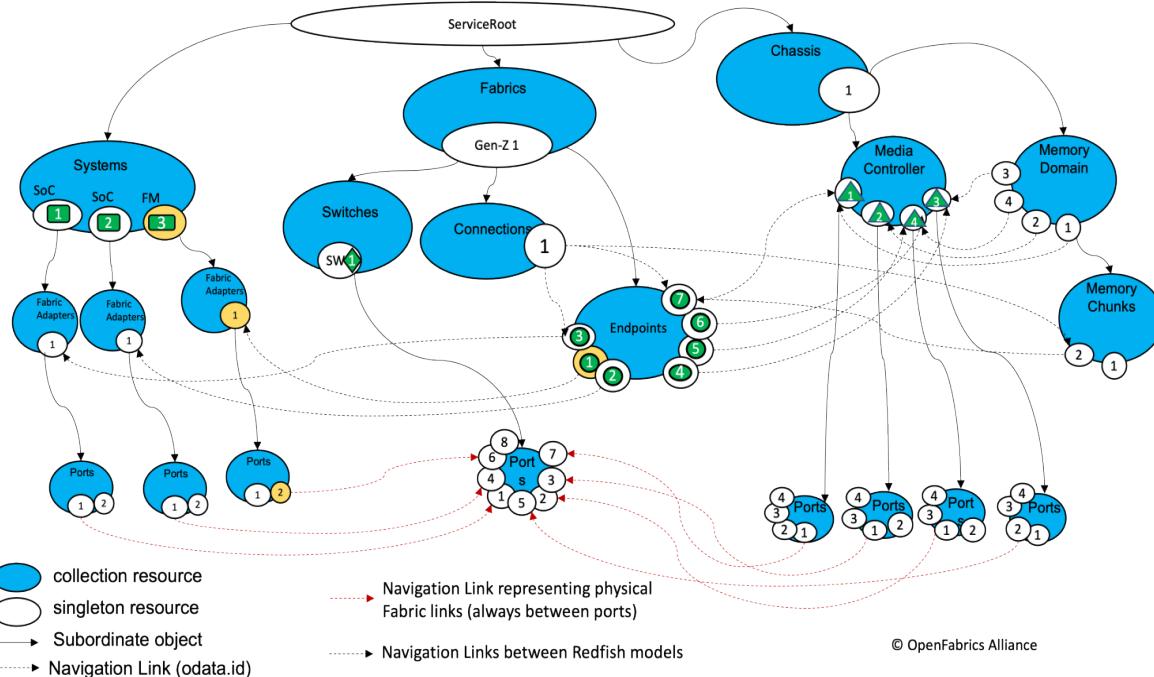
Redfish Representation of a Composable Disaggregated Infrastructure CXL-3.0



CDI HPC Nodes and Fabric Attached Memory



Simple Gen-Z Linux System Redfish Tree: Physical Objects, Endpoints, and Port linkages



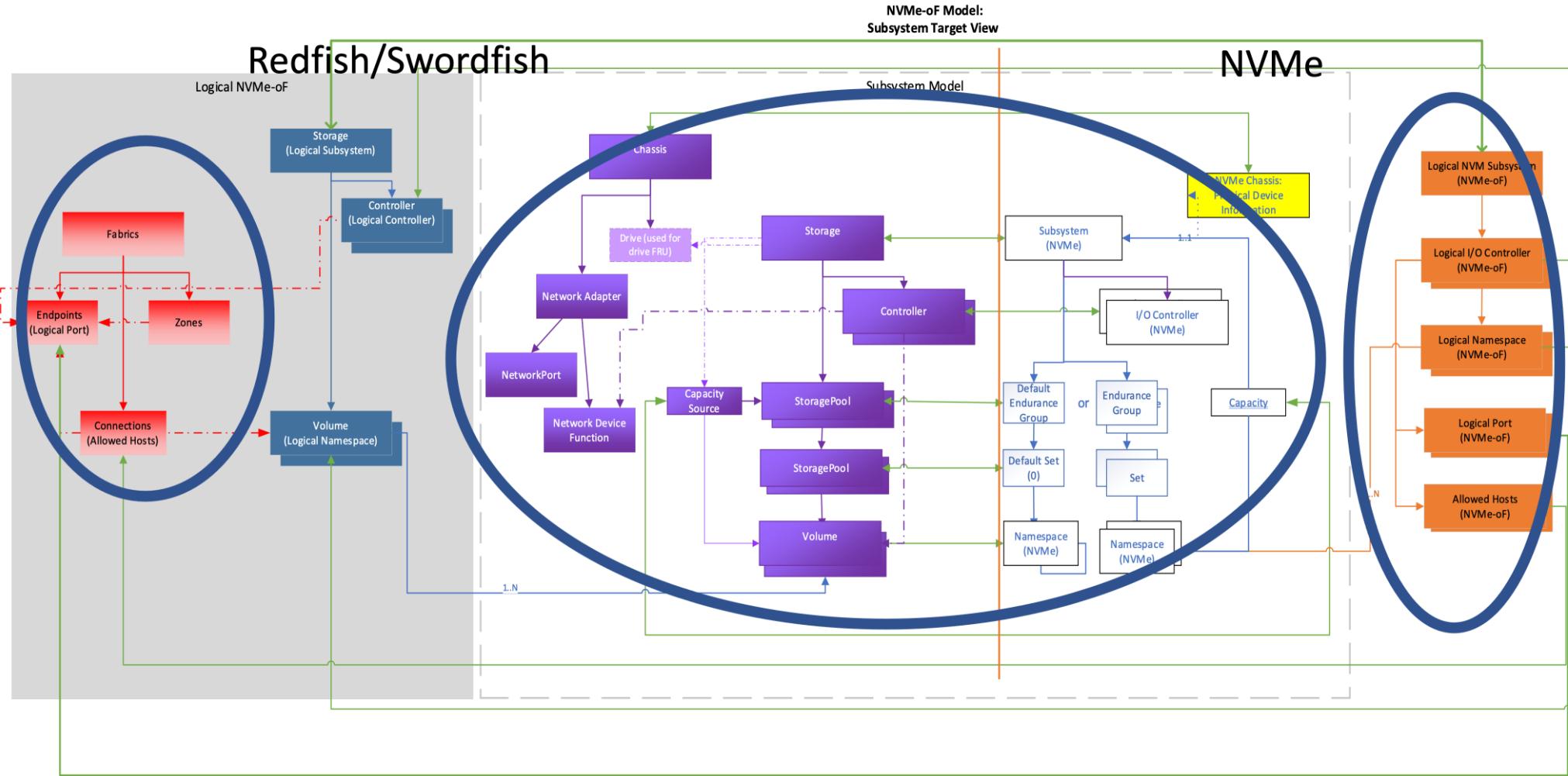
```

curl -X GET -H "Content-Type: application/json"
http://ofmfserv:5000//redfish/v1/Storage/IPAttachedDrive2/Volumes/SimpleNamespace
{
  "@odata.type": "#Volume.v1_8_0.Volume",
  "Id": "1",
  "Name": "Namespace 1",
  "LogicalUnitNumber": 1,
  "Description": "A Namespace is a quantity of non-volatile memory that may be formatted into logical blocks. When formatted, a namespace of size n is a collection of logical blocks with logical block addresses from 0 to (n-1). NVMe systems can support multiple namespaces.",
  "Status": {
    "State": "Enabled",
    "Health": "OK"
  },
  "Identifiers": [
    {
      "DurableNameFormat": "NQN",
      "DurableName": "nqn.2014-08.org.nvmeexpress:uuid:6c5fe566-10e6-4fb6-aad4-8b4159029384"
    }
  ],
  "Capacity": {
    "Data": {
      "ConsumedBytes": 0,
      "AllocatedBytes": 10737418240
    }
  },
  "NVMeNamespaceProperties": {
    "NamespaceId": "0x011",
    "NamespaceFeatures": {
      "SupportsThinProvisioning": false,
      "SupportsAtomicTransactionSize": false,
      "SupportsDeallocatedOrUnwrittenLBError": false,
      "SupportsNGUIDReuse": false,
      "
    }
  }
}

```

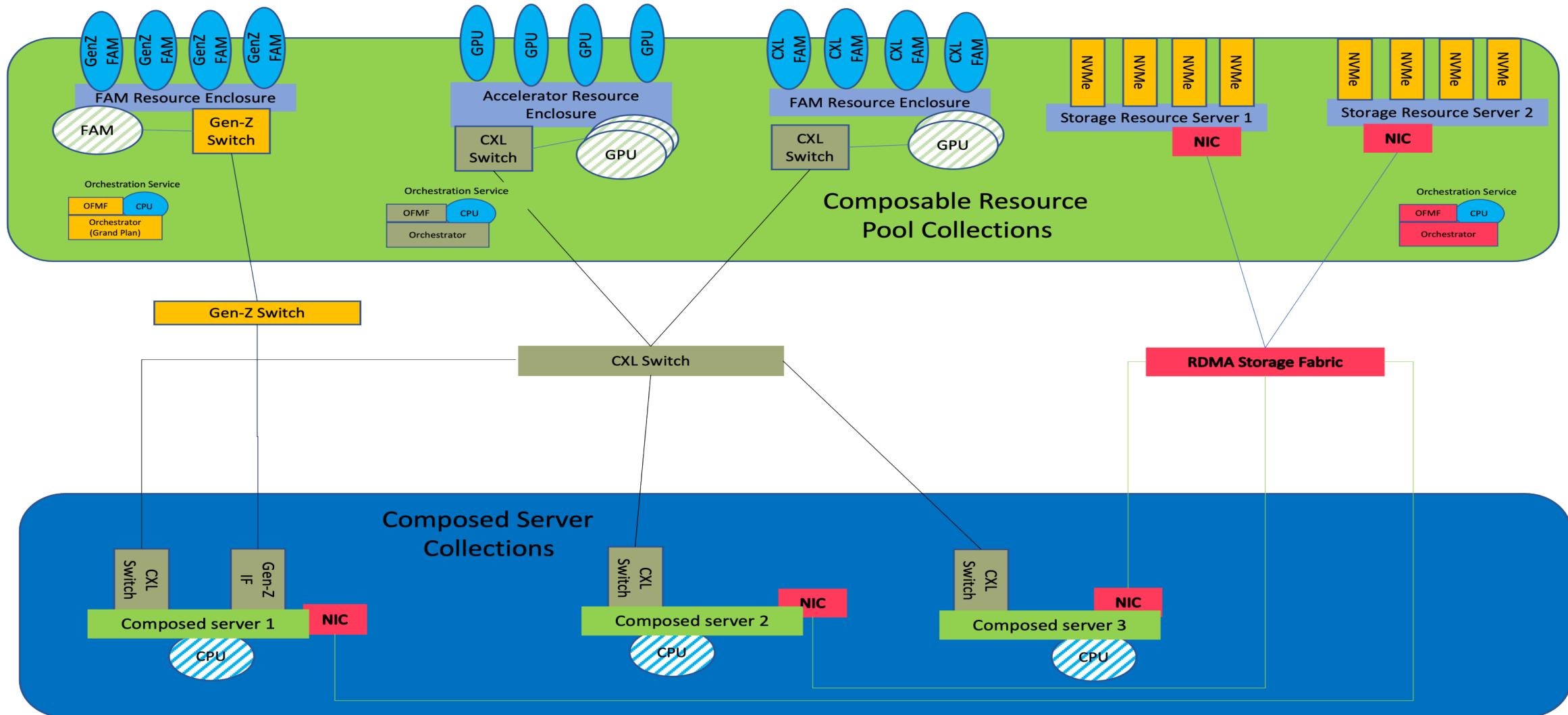
Redfish Representation of a Composable Disaggregated Infrastructure

Redfish Representation of NVMe



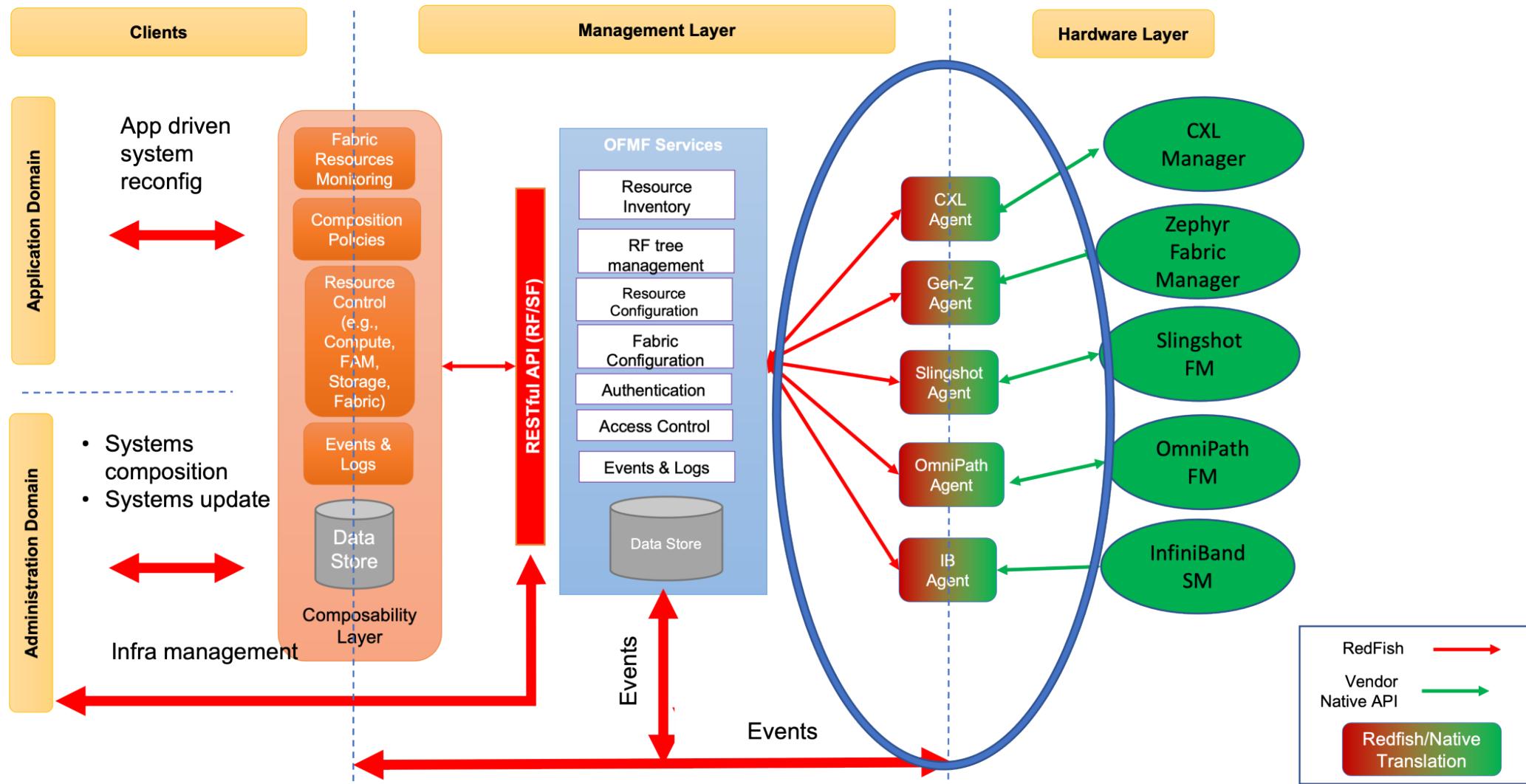
Composable Disaggregated Infrastructure (CDI) in an HPC Architecture

Composable Disaggregated HPC system controlled by the OFMF

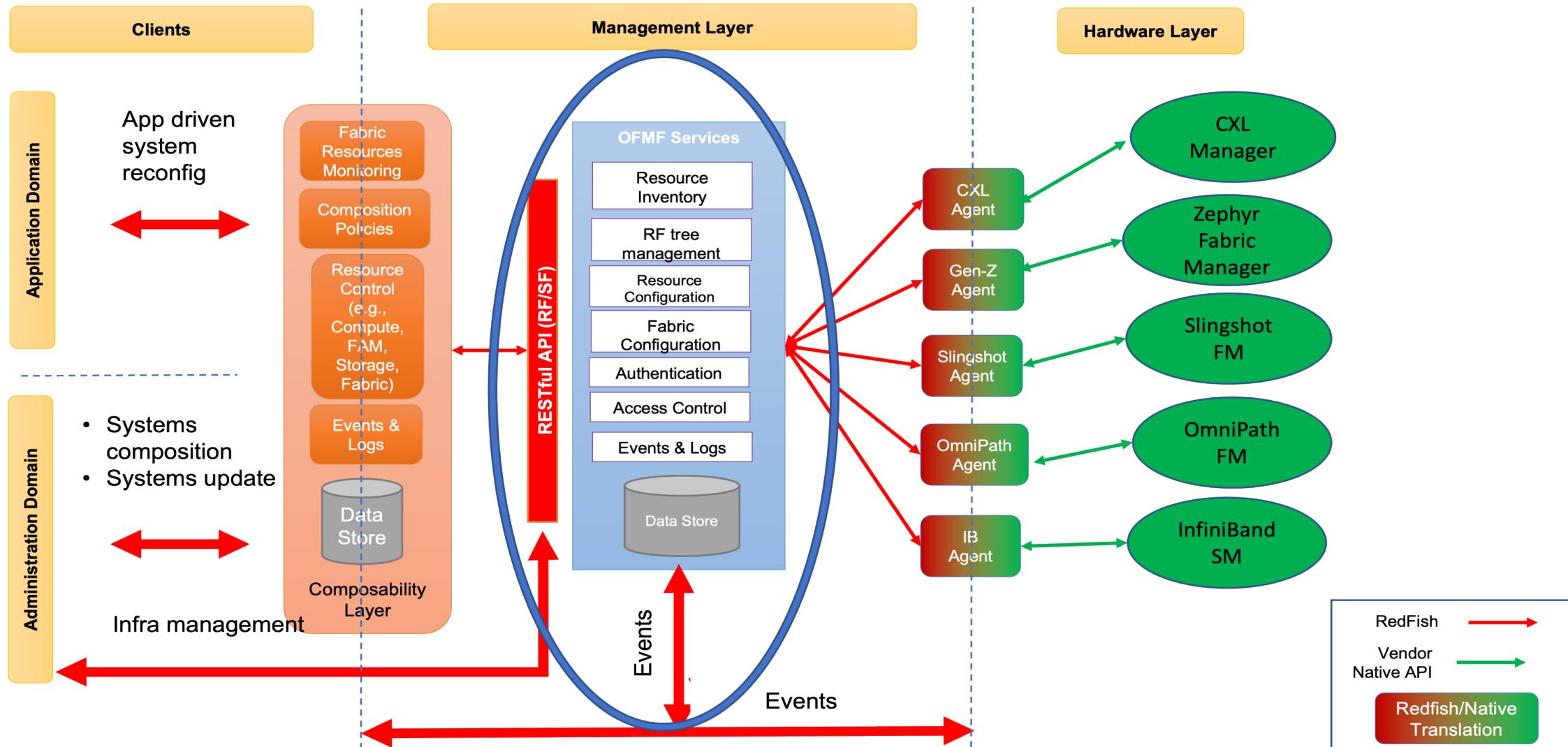


OFMF Architecture

HardWare Fabric Agents interacting with the OFMF

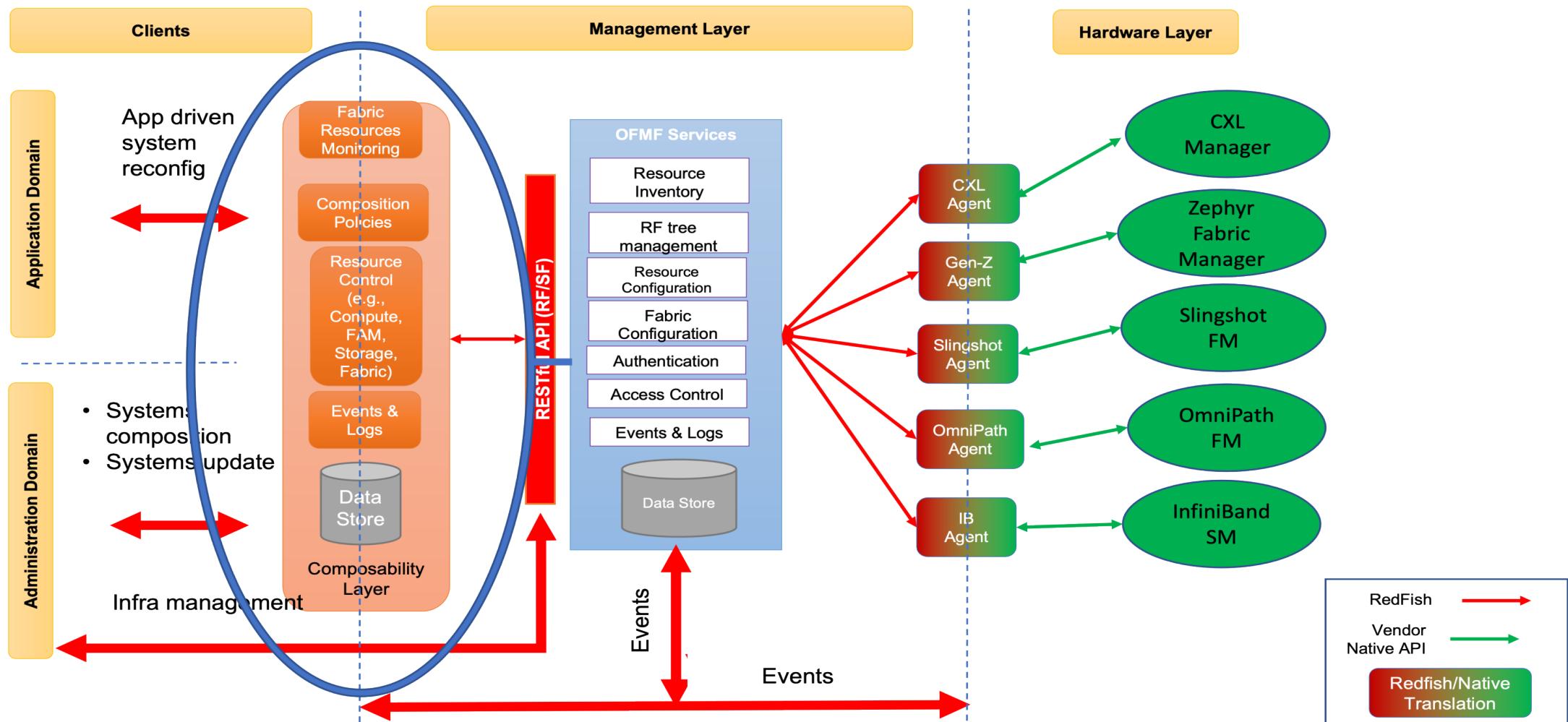


OFMF Architecture---The OFMF components



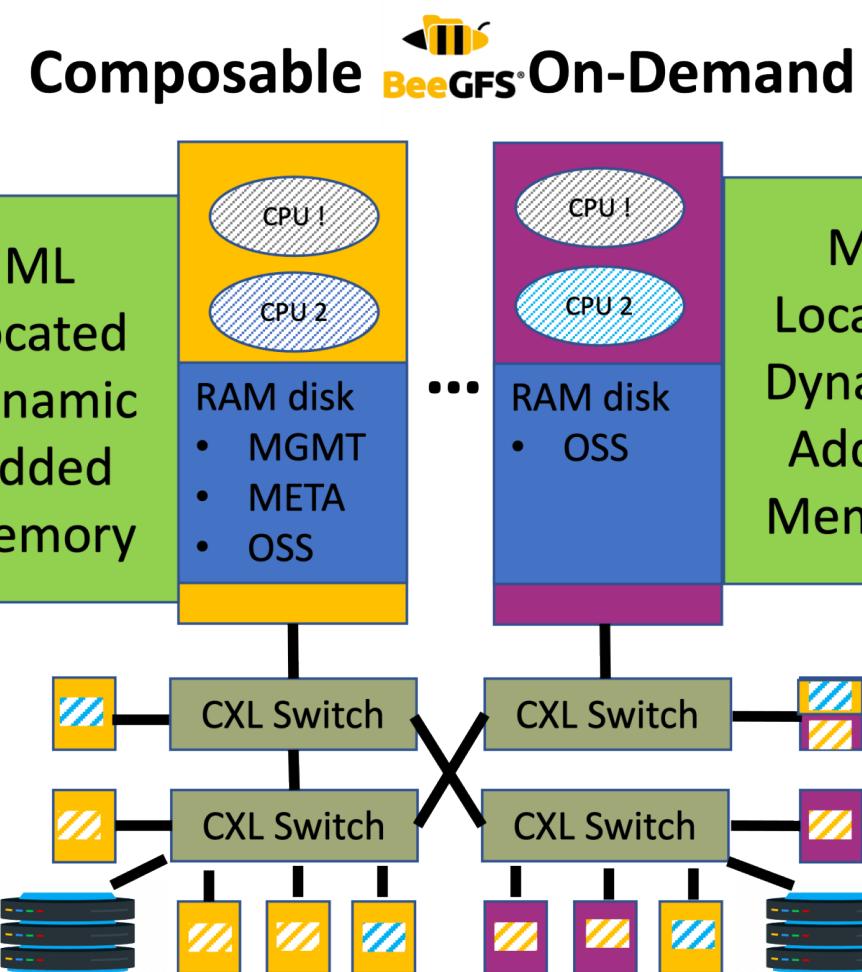
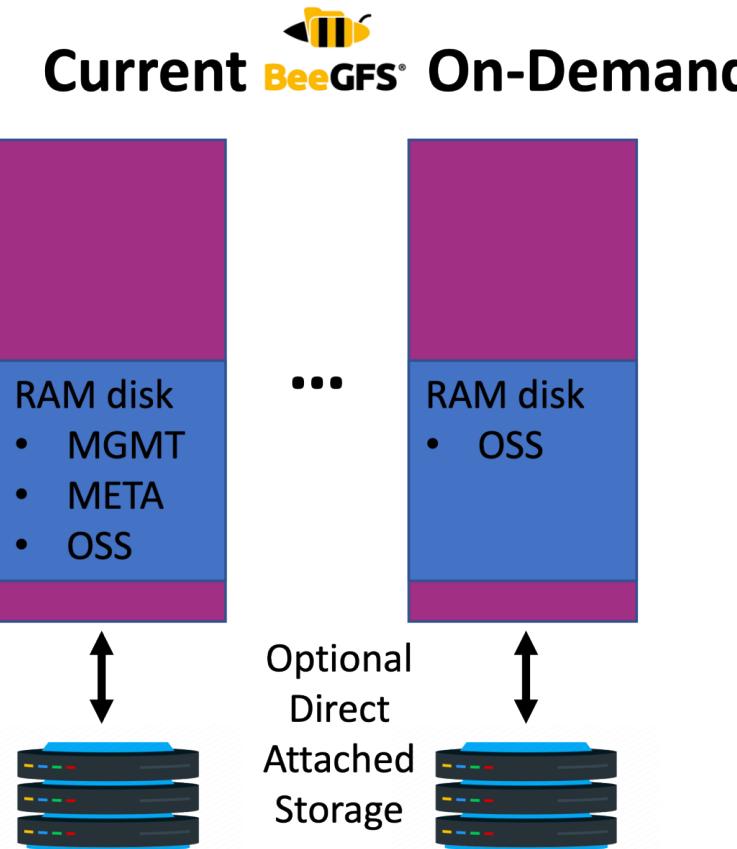
OFMF Architecture

Composable Infrastructure interacting with the OFMF

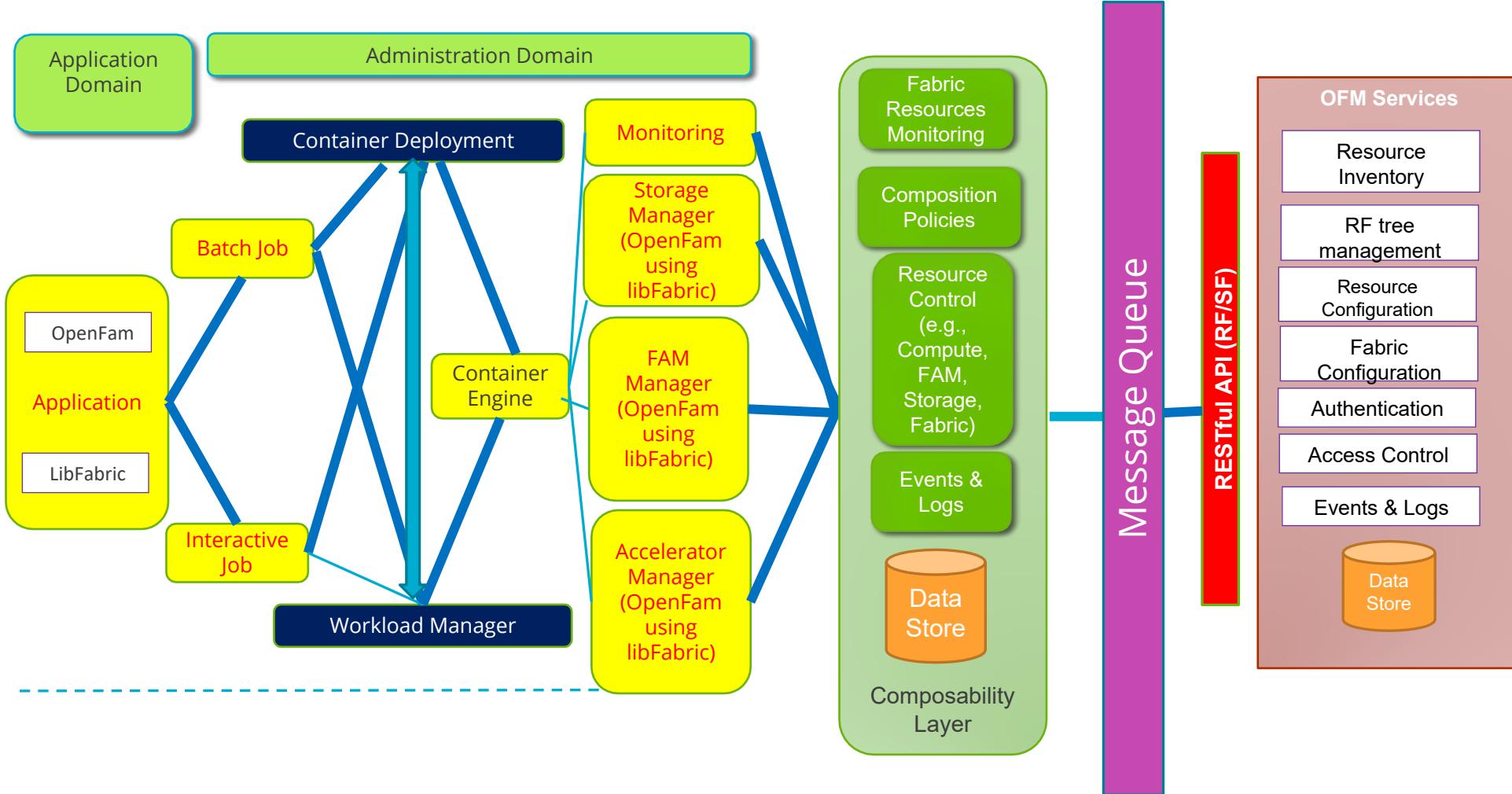


Examples of CDI HPC Use-Cases

Composable Filesystem on a composable Disaggregated Infrastructure

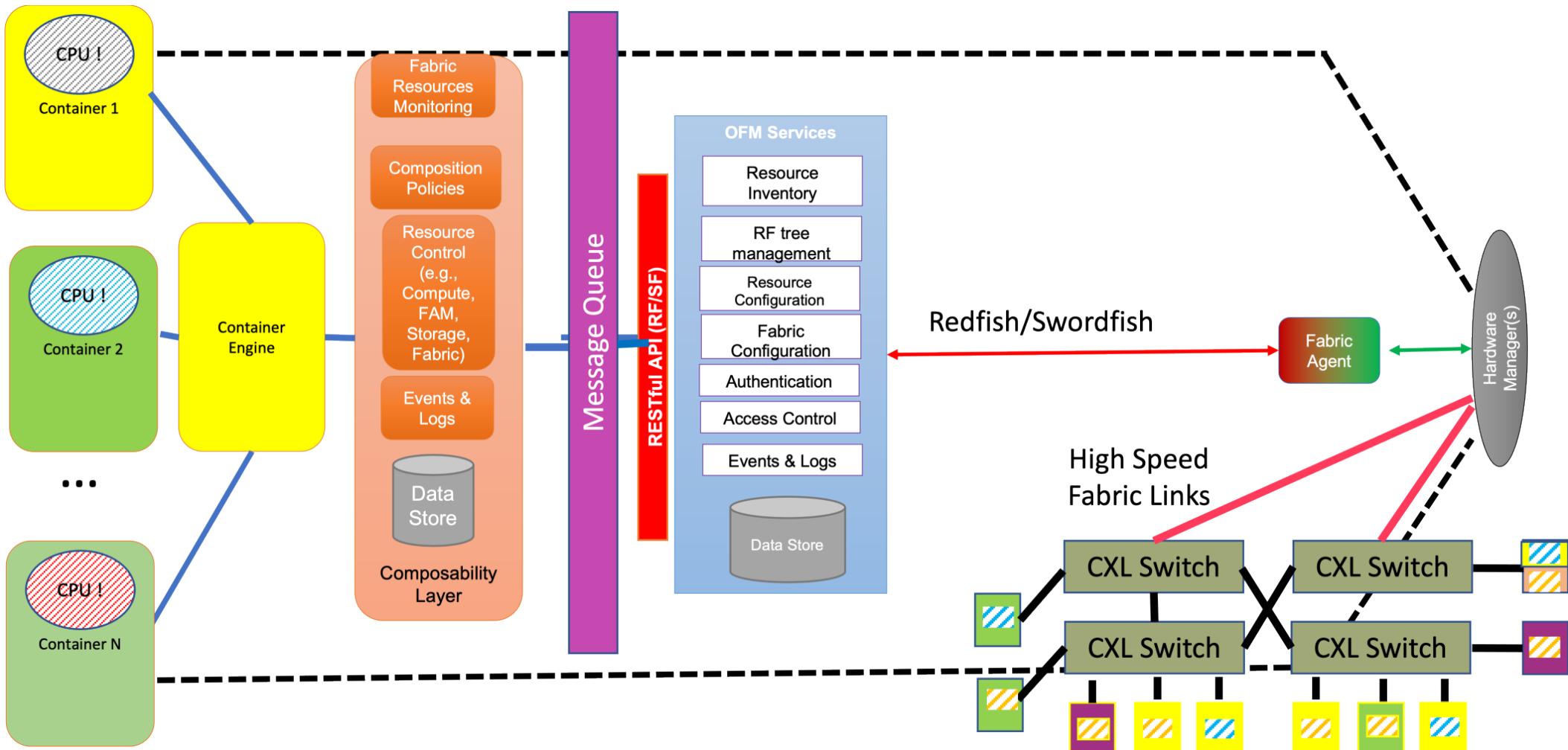


Container/Workload Manager Container Composition

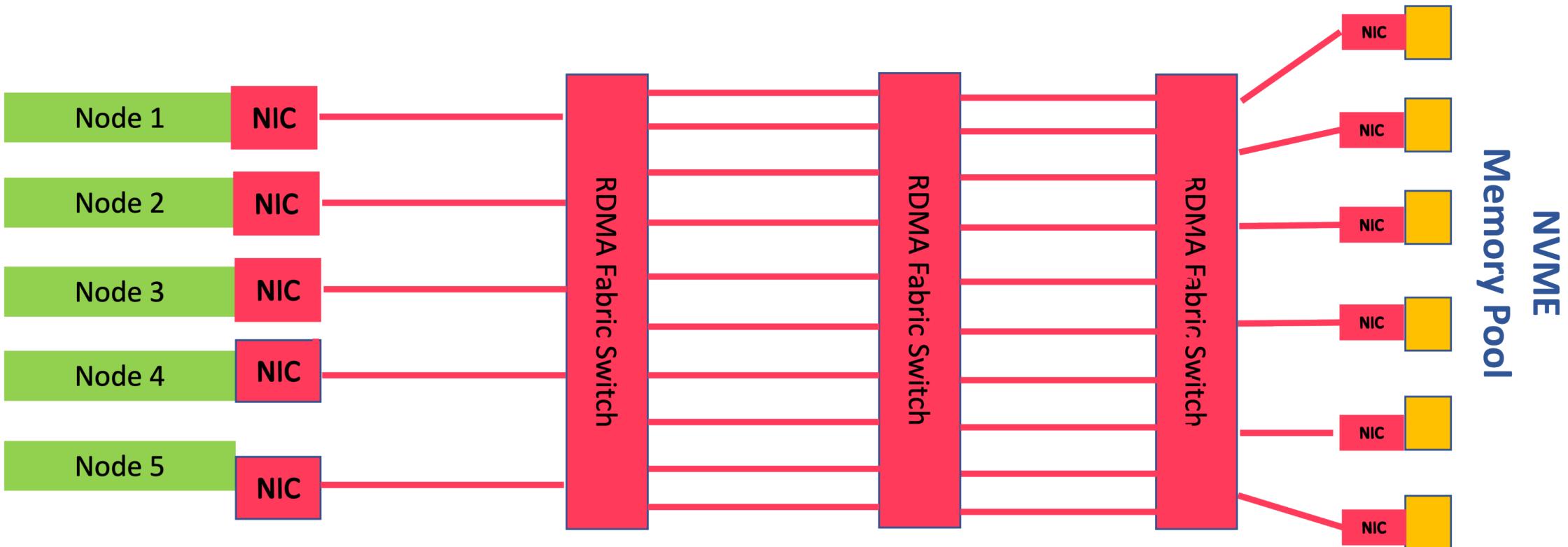


Examples of CDI HPC Use-Cases

Container Engine interacting with the OFMF



Example of NVMeoF over and RDMA fabric



What's Next for OFMF Development



We are adding Events as a way to provide notifications of changes to the HPC systems

- Events happen when a Hardware Agent provides details about network fabrics that are detected, hardware changes, etc. The events get propagated to the OFMF and to clients.

Redfish Mock-ups for CXL, GenZ, RDMA, Slingshot

OMF Redfish Tree clean-up and Stranded Resource notifications

Further development of the Reference Fabric Agent framework

Reference Composability Manager framework

Reference Fabric Attached Memory framework

Reference Monitoring framework

OS SUPPORT FOR DYNAMIC ADDITION AND DELETION OF RESOURCES

Fish name-Sunfish?

Evangelization of the OFMF to the industry

Questions?

