Open Industry Storage Management with SNIA Swordfish™

Richelle Ahlvers, Chair, SSM TWG, SMI
Storage Technology Enabling, Intel
Agenda

- What is Swordfish?
- Swordfish Overview
  - Swordfish hierarchy, layout
  - Application to block storage, NVMe
  - Storage fabric management
- Example interaction (how can Swordfish be used)
- Swordfish ecosystem overview
  - Specs, schema, documentation
  - Mockups
  - Tools and conformance testing

snia.org/swordfish
What are Redfish and Swordfish?

DMTF Redfish™ covers server, data center, fabric management

- REST API with JSON payloads; choice of CSDL, JSON and YAML schema for development

SNIA Swordfish™: Storage Management Specification with REST Based API extends DMTF’s Redfish Specification

Swordfish adds storage management to all of these use cases, plus storage fabric management

- Covers block, file, and object storage
- Extend traditional storage domain coverage to include converged environments (servers, storage and fabric together)
- Provides the option for implementation to utilize Class of Service (intent or service level) based provisioning, management, and monitoring
- NVMe / NVMe-oF devices (through an Alliance partnership with NVM Express® and DMTF)
- Storage Fabric Management: An alliance partnership with OFA, DMTF is expanding support in RF/SF for fabrics and storage fabrics management
Building on the Redfish Hierarchy for Swordfish Advanced Storage

Redfish services for account management, events, logs, tasks, session / certificates, etc.

- Service Root: 
  - /redfish/v1
- Collection Resource: 
  - /Storage
    - Storage:<id>
- Singleton Resource: 
  - Volumes
  - Volumes:<id>
- Subordinate Resource Collections: 
  - StoragePool
    - StoragePools:<id>
  - Controllers
    - Controllers:<id>
  - Drives
    - Drives:<id>
  - /Chassis
    - Chassis:<id>
- /Registries
  - /redfish/v1/Registries/AdvertisedFeatures
    - SwordfishNVMeDrive, v1.0.0

Swordfish schema (Controllers, Volumes, StoragePools, etc) attach to storage.

- SwordfishNVMeDrive, v1.0.0

- Features Registry contains the published supported Features.

- Drives contained in chassis, managed in storage pools.

- Redfish

- Swordfish

- Chassis Collection
  - /Chassis
  - Chassis:<id>
  - Drives
  - Drives:<id>
  - Drive Instance

- /EventService
  - /EventService
  - /AccountService
  - /AccountService
  - /SessionService
  - /SessionService

- /StoragePools
  - StoragePools:<id>
  - StoragePool Collection

- //redfish/v1/Registries/AdvertisedFeatures
  - SwordfishNVMeDrive, v1.0.0
Swordfish and NVMe: Common Usage

NVMe Device Usage:
Storage == Subsystem
StorageController == NVMe Controllers (IO, Admin, Discovery)
Volume == Namespace
StoragePool == Endurance Group / NVM Set
Chassis / Drive == Physical Entity Information
Redfish/Swordfish Hierarchy: Adding Fabrics

NVMe oF:
Adds Fabric and access rights
Redfish/Swordfish Hierarchy: Extending Fabric Management
Using Swordfish: Get Volume Capacity Information

- Traverse the Service Root to find the selected volume and get its Capacity information:
  1. Read the Service Root
  2. Find the link to the Storage Collection
  3. Get Storage Collection; Pick a Storage Instance
  4. Get the Storage Instance; Read the link to the Volumes Collection
  5. Read the link to the Volume Collection; Pick desired volume
  6. Get Volume; Look at Capacity Information
Swordfish Volume Capacity Step 1: Read the Service Root
(Step 2: Find the link to the Storage Collection)

```
GET /redfish/v1/ HTTP/1.1

HTTP/1.1 200 OK
{
    "@odata.context": "/redfish/v1/$metadata#ServiceRoot.ServiceRoot",
    "@odata.id": "/redfish/v1/",
    "@odata.type": ":ServiceRoot.v1_9_0.ServiceRoot",
    "Id": "RootService",
...
    "Storage": {"@odata.id": "/redfish/v1/Storage"},
    "Chassis": {"@odata.id": "/redfish/v1/Chassis" },
...
    "Links": {
        "Sessions": {"@odata.id": "/redfish/v1/SessionService/Sessions" }
    }
}
```
Swordfish Volume Capacity Step 3: Get Storage Collection; Pick a Storage Instance

GET /redfish/v1/Storage HTTP/1.1

HTTP/1.1 200 OK

{
   "@odata.id": "/redfish/v1/Storage",
   "@odata.type": "#StorageCollection.StorageCollection",
   "Name": "Storage Collection",
   "Members@odata.count": 4,
   "Members": [  
       { "@odata.id": "/redfish/v1/Storage/MyDevice" },
       { "@odata.id": "/redfish/v1/Storage/BackupDevice" },
       { "@odata.id": "/redfish/v1/Storage/FileSyncing" },
       { "@odata.id": "/redfish/v1/Storage/Simple1" }
    ]
}
Swordfish Volume Capacity Step 4: Get the Storage Instance; Read the link to the Volumes Collection

GET /redfish/v1/Storage/MyDevice HTTP/1.1

HTTP/1.1 200 OK
{
    "@odata.context": "'/redfish/v1/$metadata#StorageCollection.StorageCollection'",
    "@odata.id": "'/redfish/v1/Storage/MyDevice'",
    "@odata.type": "#Storage.v1_0_0.Storage",
    "Id": "MyDevice",
    "Name": "My Storage System",
    ...
    "Volumes": {
        "Members": [ { "@odata.id": "'/redfish/v1/StorageSystems/1/Volumes" } ]
    },
    "Links": { }
    ...
}
Swordfish Volume Capacity Step 5: Read the link to the Volume Collection; Pick desired volume

```
GET /redfish/v1/Storage/MyDevice/Volumes HTTP/1.1

HTTP/1.1 200 OK
{
    "@odata.id": "/redfish/v1/Storage/MyDevice/Volumes",
    "@odata.type": "#VolumeCollection.VolumeCollection",
    "Id": "Volumes",
    "Name": "Volume Collection",
    "Members@odata.count": 4,
    "Members": [
        { "@odata.id": "/redfish/v1/Storage/MyDevice/StoragePools/Pool1/AllocatedVolumes/61001234876545676100123487654567" },
        { "@odata.id": "/redfish/v1/Storage/MyDevice/StoragePools/Pool1/AllocatedVolumes/65456765456761001234876100123487" },
        { "@odata.id": "/redfish/v1/Storage/MyDevice/StoragePools/Pool1/AllocatedVolumes/3" },
        { "@odata.id": "/redfish/v1/Storage/MyDevice/StoragePools/Pool1/AllocatedVolumes/ 4" },
        { "@odata.id": "/redfish/v1/Storage/MyDevice/StoragePools/Pool1/AllocatedVolumes/ 5" },
        { "@odata.id": "/redfish/v1/Storage/MyDevice/StoragePools/Pool1/AllocatedVolumes/ " }
    ]
}
```
Swordfish Volume Capacity Step 6: Get Volume; Look at Capacity Information

```
GET /redfish/v1/Storage/MyDevice/StoragePools/Pool1/AllocatedVolumes/61001234876545676100123487654567
HTTP/1.1

HTTP/1.1 200 OK
{
  ...
  "Id": "61001234876545676100123487654567",
  ...
  "Capacity": {
    "Data": {
      "ConsumedBytes": 0,
      "AllocatedBytes": 10737418240,
      "GuaranteedBytes": 536870912,
      "ProvisionedBytes": 1099511627776
    }
  }
}
```
SNIA – developing an ecosystem to enable industry interoperability

- Swordfish Resources
  - Swordfish Specification, schema, and other documentation
  - Online reference mockups swordfishmockups.com
  - OpenSource Tools to accelerate development

- Swordfish Conformance Test Program to validate implementations

For deeper dives: Tools - Don Deel, “Expanding Development of your Swordfish Implementations Using Open Source Tools”
For CTP – Richelle Ahlvers, “Drive Adoption of Your Products with the Swordfish Conformance Test Program”
What’s In the Swordfish Bundle

- Swordfish Scalable Storage Management API Specification
  - defines a comprehensive, RESTful API for storage management that addresses block storage, file systems, object storage, and storage network infrastructure.

- Swordfish Schema and Registries Bundle
  - Contains the schemas defined for JSON resources conforming to the Redfish Specification. Schema definitions are available in CSDL (XML), json, and yaml formats.

- Swordfish Profile Bundle
  - Contains all the Swordfish profiles, defining the set of features and the corresponding detailed profiles define the required functionality to implement Swordfish.

- Swordfish Scalable Storage Management API User’s Guide
  - Provides a common repository of best practices, common tasks and education for the users of the Swordfish API.

- Swordfish Scalable Storage Management Error Handling Guide
  - Provides a summary of the preferred handling of errors and error messages in a Swordfish implementation.

- Swordfish NVMe Model Overview and Mapping Guide
  - Defines the model to manage NVMe and NVMe-oF storage systems with Redfish and Swordfish, and provides the detailed mapping information between the NVMe, NVMe-oF specifications and the Redfish and Swordfish specifications.

- Swordfish Property Guide
  - Provides a listing of the properties used in the Swordfish schema.
Where to Find More Info..

SNIA Swordfish™
- Swordfish Standards
  - Schemas, Specs, Mockups, User and Practical Guide`s, …
    https://www.snia.org/swordfish
- Swordfish Specification Forum
  - Ask and answer questions about Swordfish
    http://swordfishforum.com/
- Scalable Storage Management (SSM) TWG
  - Technical Work Group that defines Swordfish
  - Influence the next generation of the Swordfish standard
  - Join SNIA & participate: https://www.snia.org/member_com/join-SNIA
- Join the SNIA Storage Management Initiative
  - Unifies the storage industry to develop and standardize interoperable storage management technologies
  - https://www.snia.org/forums/smi/about/join

DMTF Redfish™
- Redfish Standards
  - Specifications, whitepapers, guides,…
    https://www.dmtf.org/standards/redfish

Open Fabric Management Framework
- OFMF Working Group (OFMFWG)
  - Description & Links https://www.openfabrics.org/working-groups/
- OFMFWG mailing list subscription
  - https://lists.openfabrics.org/mailman/listinfo/ofmfwg
- Join the Open Fabrics Alliance
  - https://www.openfabrics.org/membership-how-to-join/

NVM Express
- Specifications https://nvmexpress.org/developers/
- Join: https://nvmexpress.org/join-nvme/
Please take a moment to rate this session.

Your feedback is important to us.