

SDC 18

September 24-27, 2018
Santa Clara, CA

www.storagedeveloper.org



Swimming With SNIA Swordfish™

Richelle Ahlvers

Principal Storage Management Architect, Broadcom Inc.

SNIA Scalable Storage Management (SSM) Technical Work Group Chair

Disclaimer

- The information in this presentation represents a snapshot of work in progress within SNIA
- This information is subject to change without notice.
- For additional information, see the SNIA website: www.snia.org/swordfish

Abstract

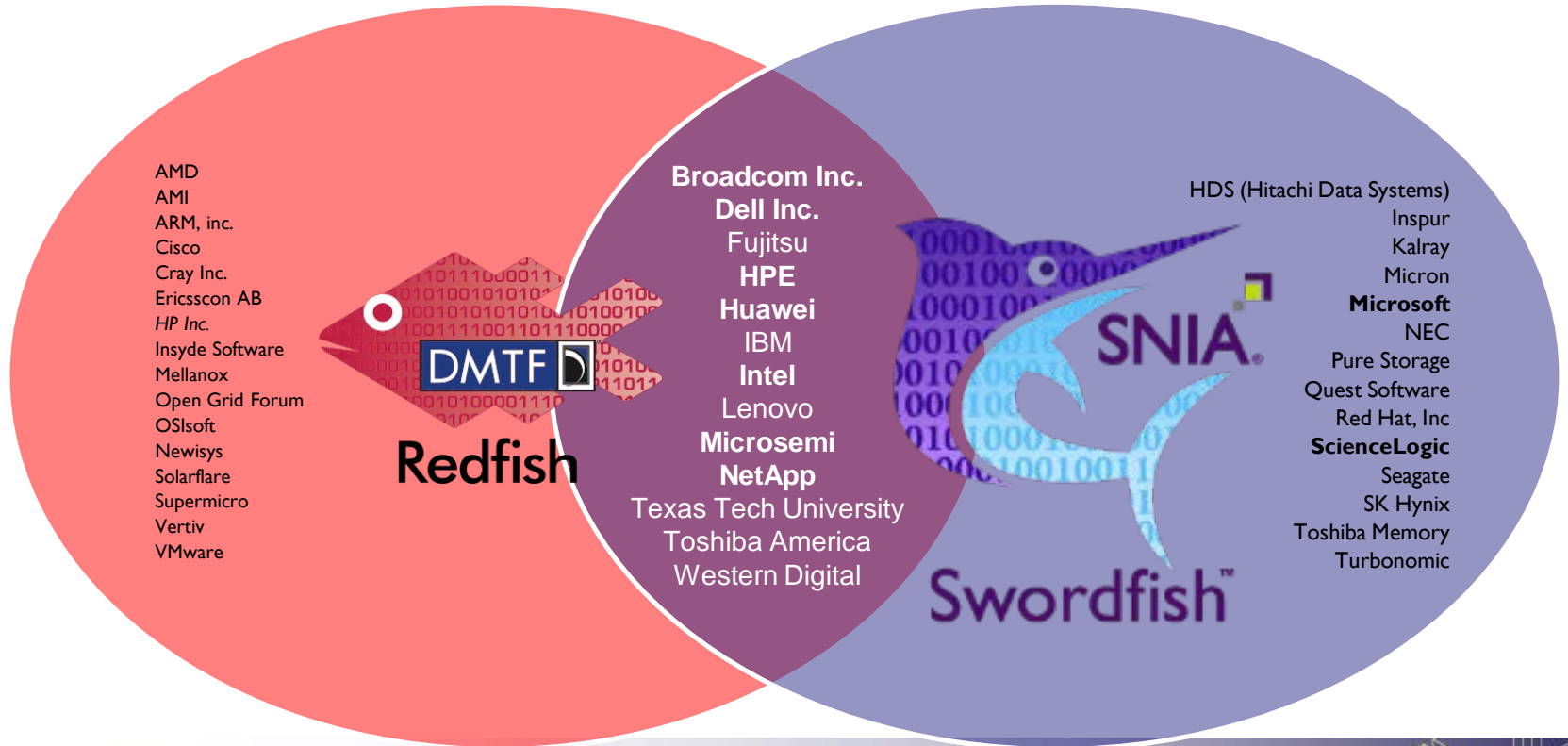


- Swordfish™ is an extension of the DMTF Redfish specification developed by the Storage Networking Industry Association (SNIA) to provide a unified approach for the management of storage equipment and services in converged, hyper-converged, hyperscale and cloud infrastructure environments, making it easier for IT administrators and DevOps to integrate scalable solutions into their data centers.
- This session will present an overview of the SNIA Swordfish specification, and will show how Swordfish takes and extends the Redfish specification to deliver the Swordfish storage model. It will also cover the drivers for the SNIA Swordfish approach, as well as providing a comprehensive overview of the functionality included in the Swordfish specification. It will also present an overview of common storage management use cases easily addressed by Swordfish.

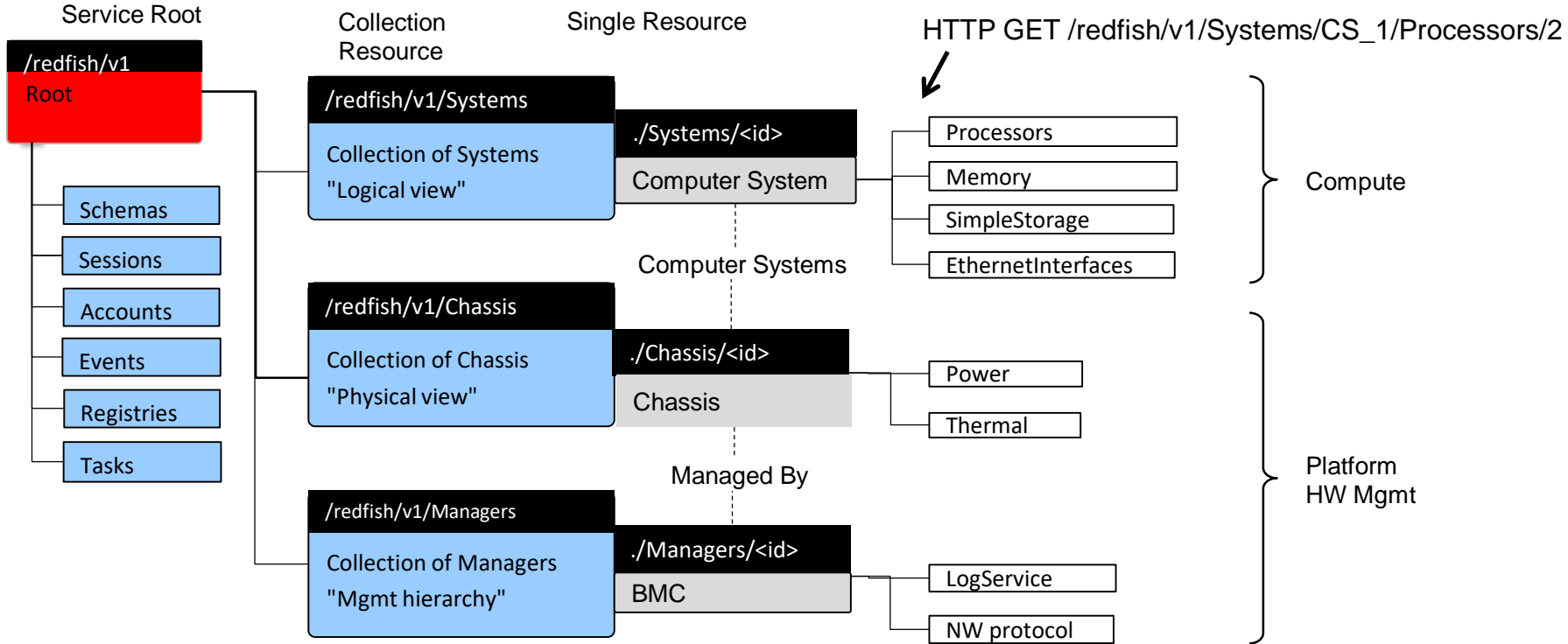
The SNIA Swordfish™ Approach

- Develop the management model from the point-of-view of what a client needs to accomplish and to provide information that the client needs.
- Utilize Class of Service (intent or service level) based provisioning, management, and monitoring
- Cover block, file and object storage
- Extend traditional storage domain coverage to include converged environments (covering servers, storage and fabric together)
- Implement the Swordfish API as an **extension** of the Redfish API
- Build using DMTF's Redfish technologies
 - RESTful interface over HTTPS in JSON format based on OData v4

Who is Developing Redfish and Swordfish*?

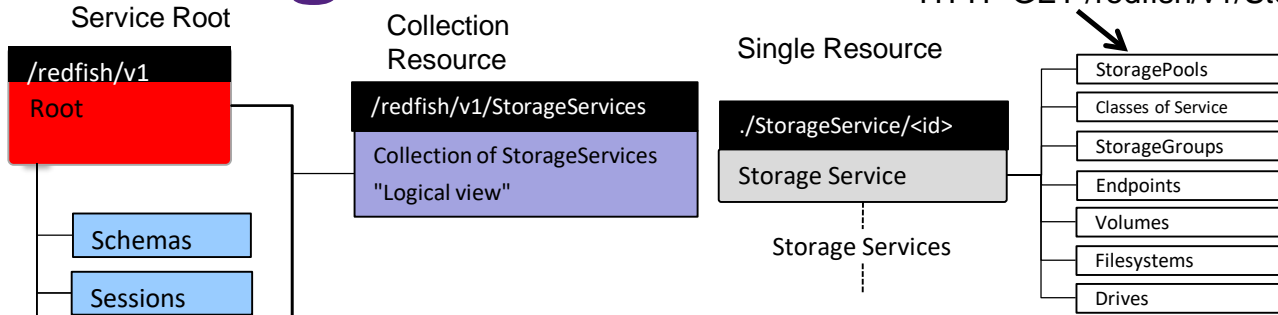


Starting with Redfish: Basic Compute

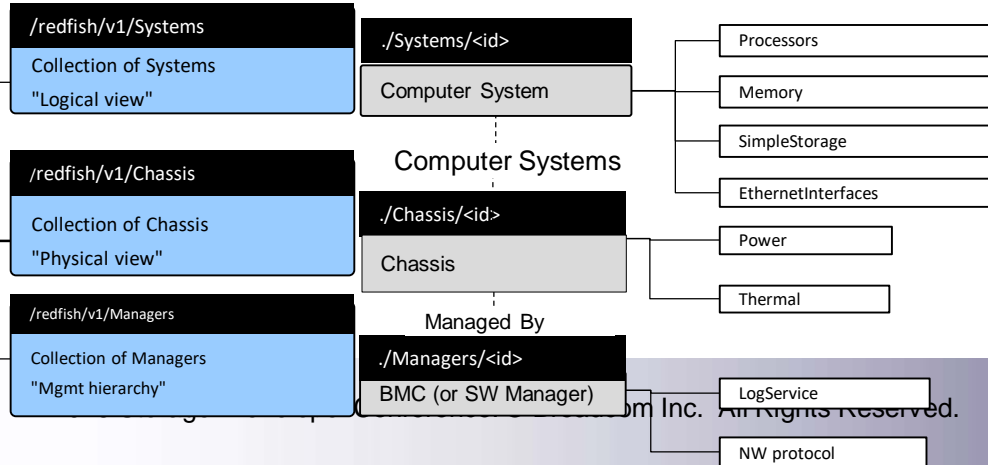


Adding Swordfish

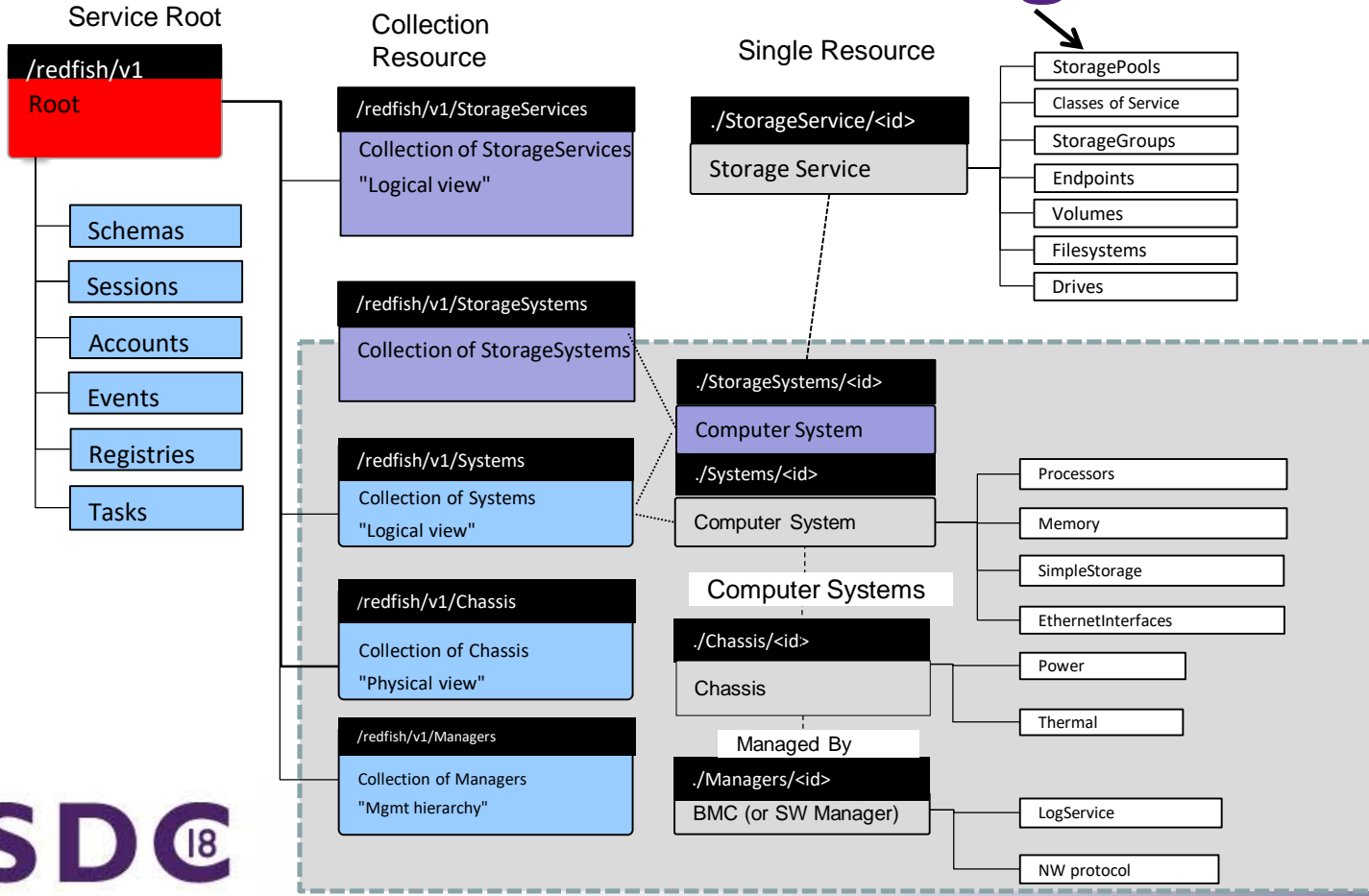
HTTP GET /redfish/v1/StorageServices/SS_1/StoragePools/2



(Two ways to host Storage Services...)



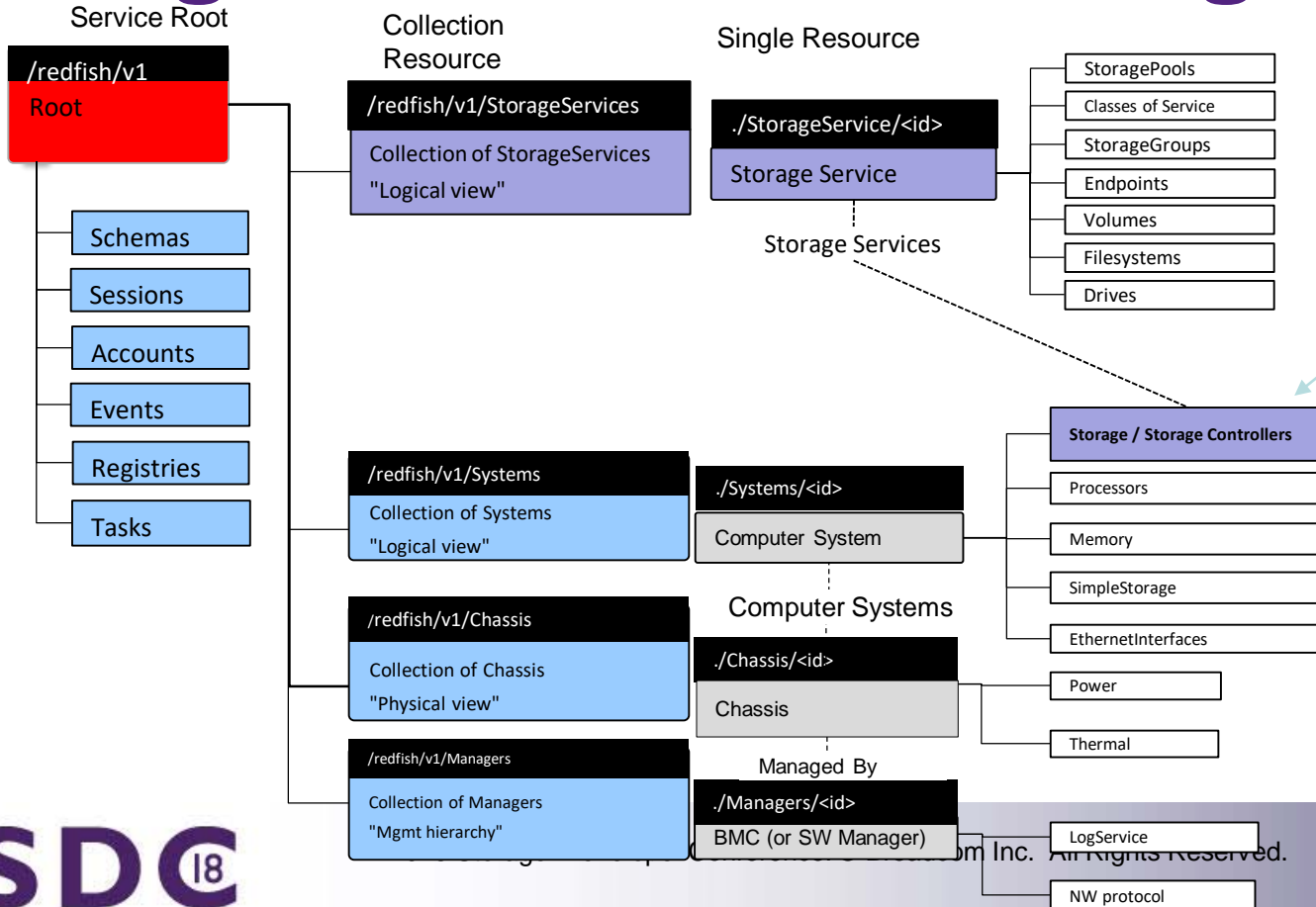
Hosted Service Configuration



A StorageSystem *is* an instance of a ComputerSystem (with a small set of additional properties defined).



Integrated Service Configuration



The Storage/StorageController in ComputerSystem hosts the StorageServices. It is capable of creating StorageServices and additional Swordfish functionality.



Swordfish Capabilities

- Block storage
 - Provisioning with **class of service** control
 - Resource provisioning from disk, volume, pool, and persistent memory
 - Volume Mapping and Masking
 - Local and Remote Replication
 - Capacity and health metrics
 - Performance metrics
- File system storage
 - Adds File System and File Share
 - Leverages all other concepts – provisioning with class of service, replication, ...
- Solution level connectivity
 - Fabric connect, host connect
 - Endpoint abstraction
- Additional content
 - Object drive storage

Primary Swordfish Elements

- ClassOfService
 - A choice of utility or warranty offered to customers by a service. Defined by selecting from available LinesOfService.
- StorageService:
 - Represents a service that provides ClassOfService based provisioning, management, and monitoring for logical storage and associated resources
- StoragePools
 - Storage capacity that can be used to produce volumes or other storage pools with a specified class of service.
- Volume
 - Block addressable storage that is conformant to a ClassOfService.
- StorageGroup
 - A set of volumes that are managed as a group; e.g., with the same consistency requirements.
- Filesystem
 - File-addressable storage that is conformant to a ClassOfService.
- Fileshare
 - A shared set of files with a common directory structure that is exported for use by remote systems.



Using Swordfish: Get Volume Capacity Information

Traverse the Service Root to find the selected volume and get its Capacity information:

- Read the Service Root resource
- Read the link to the Storage Service Collection
- Pick a Storage Service
- Read the link to the Volume Collection
- Pick desired Volume:
 - Collect the Capacity information
 - Look at the consumed vs allocated capacity information

Note: Data for this example pulled from swordfishmockups.com

Swordfish Volume Capacity Step 1: Read the Service Root

(Step 2: Read the link to the Storage Service 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_0_0.ServiceRoot",
  "Id": "RootService",
  ...
  "StorageSystems": {"@odata.id": "/redfish/v1/StorageSystems" },
  "StorageServices": {"@odata.id": "/redfish/v1/StorageServices" },
  "Chassis": {"@odata.id": "/redfish/v1/Chassis" },
  ...
  "Links": {
    "Sessions": {"@odata.id": "/redfish/v1/SessionService/Sessions" }
  },
}
```

Swordfish Volume Capacity Step 3: Pick a Storage Service

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

```
HTTP/1.1 200 OK
```

```
{
  "@odata.context":
"/redfish/v1/$metadata#StorageServiceCollection.StorageServiceCollection",
  "@odata.id": "/redfish/v1/Systems/Simple/StorageServices",
  "@odata.type": "#StorageServiceCollection.v1_0_0.StorageServiceCollection",
  "Name": "Storage Service Collection",
  "Members@odata.count": 4,
  "Members": [
    { "@odata.id": "/redfish/v1/StorageServices/1" },
    { "@odata.id": "/redfish/v1/StorageServices/2" },
    { "@odata.id": "/redfish/v1/StorageServices/FileService" },
    { "@odata.id": "/redfish/v1/StorageServices/Simple1" }
  ]
}
```

Swordfish Volume Capacity Step 4: Read the link to the Volume Collection

```
GET /redfish/v1/StorageServices/1 HTTP/1.1
```

```
HTTP/1.1 200 OK
```

```
{
  "@odata.context": "/redfish/v1/$metadata#StorageService.StorageService",
  "@odata.id": "/redfish/v1/StorageServices/1",
  "@odata.type": "#StorageService.v1_0_0.StorageService",
  "Id": "1",
  "Name": "My Storage Service",
  ...
  "Volumes": {
    "Members": [ { "@odata.id": "/redfish/v1/StorageServices/1/Volumes" } ]
  },
  "StorageSubsystems": { "@odata.id": "/redfish/v1/StorageServices/1/StorageSubsystems"
},
  "Links": { }
  ...
}
```

Swordfish Volume Capacity Step 5: Pick Desired Volume

```
GET /redfish/v1/StorageServices/1/Volumes HTTP/1.1
```

```
HTTP/1.1 200 OK
```

```
{  
  ...  
  "Name": "Volumes",  
  "Members@odata.count": 6,  
  "Members": [  
    { "@odata.id":  
      "/redfish/v1/StorageServices/1/Volumes/61001234876545676100123487654567" },  
    { "@odata.id":  
      "/redfish/v1/StorageServices/1/Volumes/65456765456761001234876100123487" },  
    { "@odata.id": "/redfish/v1/StorageServices/1/Volumes/3" },  
    { "@odata.id": "/redfish/v1/StorageServices/1/Volumes/4" },  
    { "@odata.id": "/redfish/v1/StorageServices/1/Volumes/5" },  
    { "@odata.id": "/redfish/v1/StorageServices/1/Volumes/6" }  
  ]  
}
```


Swordfish Volume Capacity Step 5a and 5b: Look at Capacity Information

```
GET
/redfish/v1/StorageServices/1/Volumes/6100123
4876545676100123487654567 HTTP/1.1
```

```
HTTP/1.1 200 OK
```

```
{
...
  "Id": "61001234876545676100123487654567",
...
}
```

```
"Capacity": {
  "Data": {
    "ConsumedBytes": 0,
    "AllocatedBytes": 10737418240,
    "GuaranteedBytes": 536870912,
    "ProvisionedBytes": 1099511627776
  },
  "Metadata": {
    ...
  },
  "Snapshot": {
    ...
  }
}
```

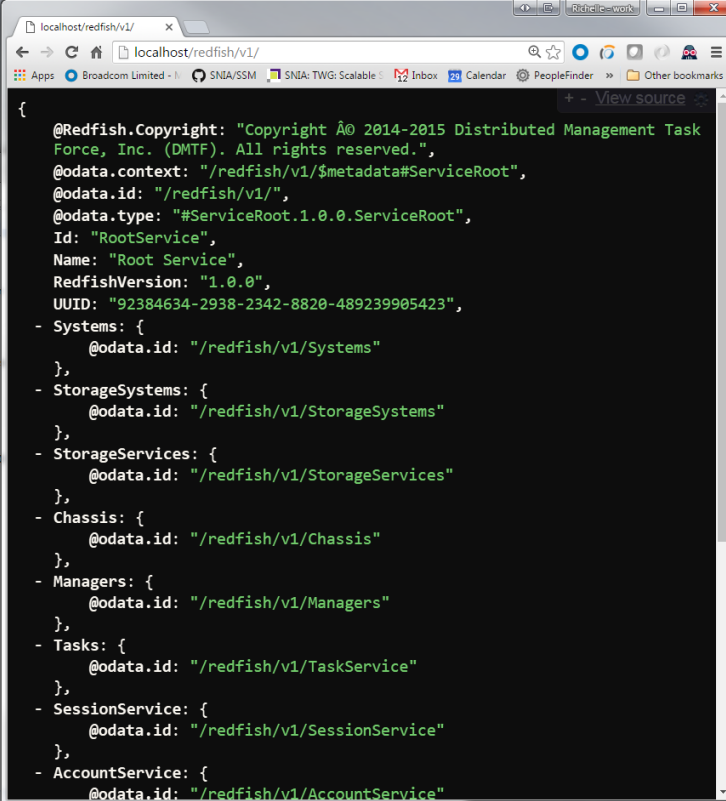
See example Swordfish configurations

- As a work tool, the Technical Work Group (TWG) works with “mockups” (snapshots of a state in time) of different types of systems
- Published at [http://swordfishmockups.com](http://swordfishmockups.com(/redfish/v1/)) (/redfish/v1/)

Note: Mockups are representations of implementations, not normative

Overview of Swordfish

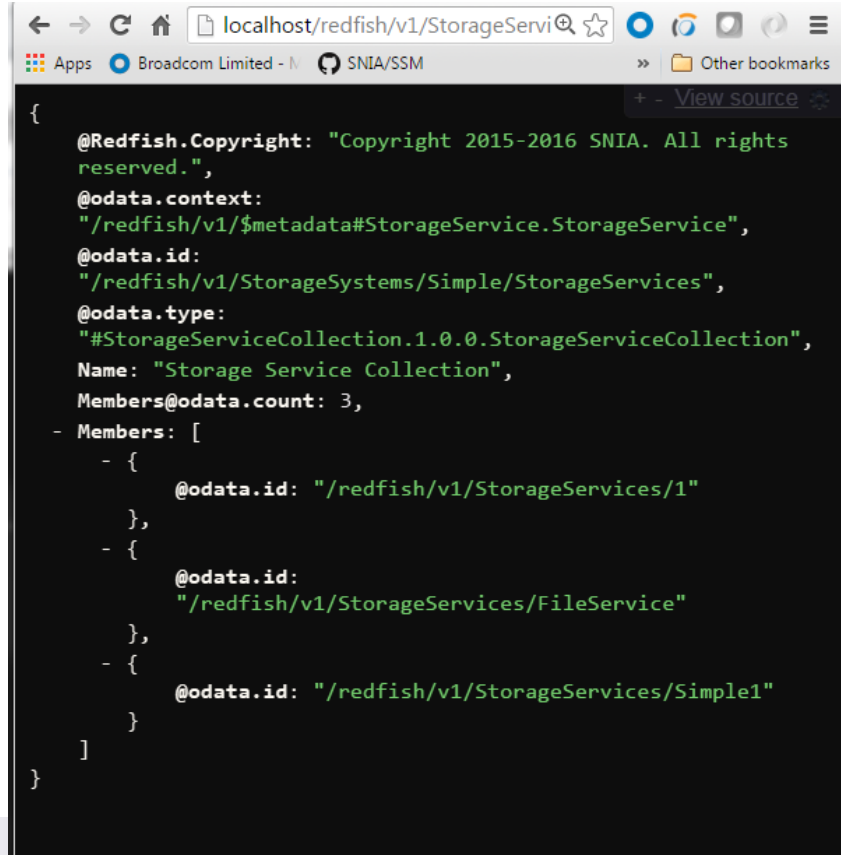
- Explore the Swordfish data model to see potential / typical implementation
- Navigate through the model to learn about and see various resources
- SNIA mockups show examples of block storage systems
 - Simple: A small external array
 - Complex: all of the elements in the block storage model, with remote replication
- Also show ISC vs HSC configurations
- .. and an example of a file server with multiple file shares



```
{
  @Redfish.Copyright: "Copyright Â© 2014-2015 Distributed Management Task
  Force, Inc. (DMTF). All rights reserved.",
  @odata.context: "/redfish/v1/$metadata#ServiceRoot",
  @odata.id: "/redfish/v1/",
  @odata.type: "#ServiceRoot.1.0.0.ServiceRoot",
  Id: "RootService",
  Name: "Root Service",
  RedfishVersion: "1.0.0",
  UUID: "92384634-2938-2342-8820-489239905423",
  - Systems: {
    @odata.id: "/redfish/v1/Systems"
  },
  - StorageSystems: {
    @odata.id: "/redfish/v1/StorageSystems"
  },
  - StorageServices: {
    @odata.id: "/redfish/v1/StorageServices"
  },
  - Chassis: {
    @odata.id: "/redfish/v1/Chassis"
  },
  - Managers: {
    @odata.id: "/redfish/v1/Managers"
  },
  - Tasks: {
    @odata.id: "/redfish/v1/TaskService"
  },
  - SessionService: {
    @odata.id: "/redfish/v1/SessionService"
  },
  - AccountService: {
    @odata.id: "/redfish/v1/AccountService"
  }
}
```

Navigating through the Mockups...

- Select the [.../redfish/v1/StorageServices](#) link to see the “Collection” of Storage Services
- Click the [.../StorageServices/Simple](#) link to see the details of the Simple mockup
 - [.../StorageServices/1](#) to see the details of the complex storage service mockup
 - [.../StorageServices/FileService](#) to see the filesystem mockup
 - [.../StorageServices/ISC](#) to see the ISC mockup (look for links to the hosting system)



```
{
  @Redfish.Copyright: "Copyright 2015-2016 SNIA. All rights reserved.",
  @odata.context: "/redfish/v1/$metadata#StorageService.StorageService",
  @odata.id: "/redfish/v1/StorageSystems/Simple/StorageServices",
  @odata.type: "#StorageServiceCollection.1.0.0.StorageServiceCollection",
  Name: "Storage Service Collection",
  Members@odata.count: 3,
  - Members: [
    - {
      @odata.id: "/redfish/v1/StorageServices/1"
    },
    - {
      @odata.id: "/redfish/v1/StorageServices/FileService"
    },
    - {
      @odata.id: "/redfish/v1/StorageServices/Simple1"
    }
  ]
}
```

What's in a Storage Service? (Block)

- Available Classes Of Service
 - Lines of Service that are used to compose the Classes of Service
- Volumes
- Pools
- Groups
- Endpoints
- ...
- Pointer to related resources (system, chassis,..)



```
localhost/redfish/v1/Storage x  
localhost/redfish/v1/StorageServ  
Apps Broadcom Limited - S  
SNIA/SSM Other bookmarks  
+ - View source  
{  
  @Redfish.Copyright: "Copyright 2014-2016 SNIA. All rights reserved.",  
  @odata.context: "/redfish/v1/$metadata#StorageService.StorageService",  
  @odata.id: "/redfish/v1/StorageServices/1",  
  @odata.type: "#StorageService.1.0.0.StorageService",  
  Id: "1",  
  Name: "My Storage Service",  
  Description: "Description of storage",  
  + Status: {...},  
  + ClassesOfService: [...],  
  - Drives: {  
    @odata.id: "/redfish/v1/Chassis/StorageEnclosure1/Drives"  
  },  
  + InitiatorEndpointGroups: [...],  
  + TargetEndpointGroups: [...],  
  + Endpoints: {...},  
  + StorageGroups: [...],  
  - StoragePools: {  
    @odata.id: "/redfish/v1/StorageServices/1/StoragePools"  
  },  
  - Volumes: {  
    @odata.id: "/redfish/v1/StorageServices/1/Volumes"  
  },  
  - Links: {  
    - Enclosures: {  
      @odata.id: "/redfish/v1/Chassis/1"  
    },  
    - HostingSystem: {  
      @odata.id: "/redfish/v1/StorageSystems/Complex"  
    },  
    - DataProtectionLoSCapabilities: {  
      @odata.id: "/redfish/v1/StorageServices/1/DataProtectionLoSCapabilities"  
    },  
    - DataSecurityLoSCapabilities: {  
      @odata.id: "/redfish/v1/StorageServices/1/DataSecurityLoSCapabilities"  
    }  
  }  
}
```

What's in a Storage Service? (File)

Same structure:

- Available Classes Of Service
- *File systems*
- Pools
- Groups
- Endpoints
- ...
- Pointer to related resources (system, chassis, **block service** or drives)



```
{
  @Redfish.Copyright: "Copyright 2014-2016 SNIA. All rights reserved.",
  @odata.context: "/redfish/v1/$metadata#StorageService.StorageService",
  @odata.id: "/redfish/v1/StorageServices/FileService",
  @odata.type: "#StorageService.1.0.0.StorageService",
  Id: "1",
  Name: "My Storage Service",
  Description: "Description of storage",
  - Status: {
    State: "Enabled",
    Health: "OK"
  },
  + ClassesOfService: [...],
  - FileSystems: {
    @odata.id: "/redfish/v1/StorageServices/FileService/FileSystems"
  },
  - StorageServiceCapabilities: {
    @odata.id: "/redfish/v1/StorageServices/FileService/StorageServiceCapabilities"
  },
  + StorageGroups: [...],
  + StoragePools: {...},
  - Links: {
    - Enclosures: {
      @odata.id: "/redfish/v1/Chassis/1"
    },
    - HostingSystem: {
      @odata.id: "/redfish/v1/StorageSystems/FileServer"
    }
  },
  Oem: { }
}
```

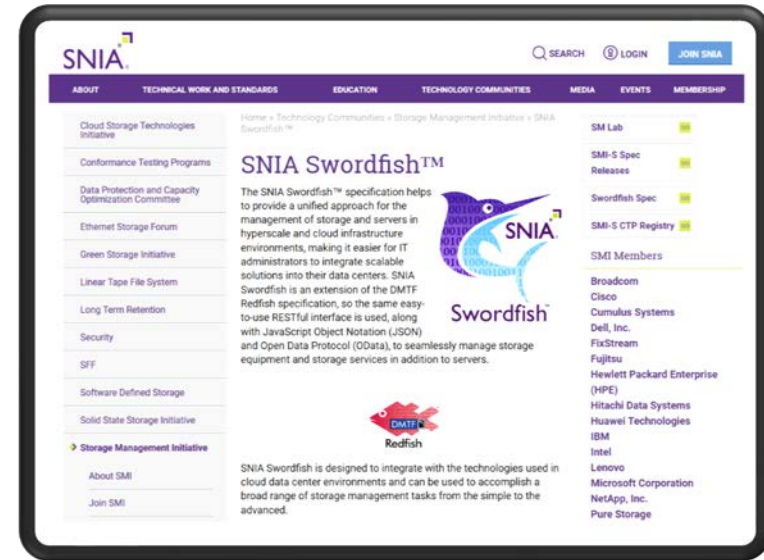
Swordfish Specs and Technical Content... In 2018

- v1.0.6 Technical Position Release in (WIP in Feb 2018, TP in May 2018)
 - Introduction of two StorageSystem models:
 - Hosted Service Configuration and Integrated Service Configuration
 - Schema updates, Spec section additions, User's guide updates: new use cases for on-demand replicas
- Q4 2018 Releases:
 - Updated Swordfish mockups: swordfishmockups.com
 - v1.0.7 Swordfish WIP Release:
 - Enhanced Class of Service Capabilities for Spare Capacity Management, Rebuild Management, Volume types
 - White Paper for Spare Management
 - WIP Profile Development: Basic Swordfish Support
- Future Functionality
 - Storage-specific security roles
 - Enhanced profiles for SNIA Alliance partner organizations
 - Functionality alignment across DMTF, NVMeExpress/NVMe-MI and SNIA
 - Object Storage

Swordfish Info: www.snia.org/swordfish



- Resources
 - Specifications
 - User's Guide
 - GitHub for Swordfish Tools
 - Practical Guide
 - Other Documentation
- Swordfish Mockups Site
 - ISC and HSC configurations
 - Block vs file configurations
 - Small and large configurations
- Education/Community
 - Whitepapers, Presentations
 - YouTube shorts & Webinars
- Participate
 - Join SNIA and the SSM TWG
 - Implement



Open Source Tools and Infrastructure Development



- Available: <http://github.com/sniaswordfish>
 - Swordfish Emulator Extensions
 - Extends the Redfish emulator – adds all Swordfish schema (behave like dynamic objects)
 - Basic Swordfish Web client
 - Discover, display and edit Swordfish services
 - DataDog and Power BI Client Sample Dashboards
 - Sample implementations show integration concepts with sample code:
 - PowerBI: Point-in-time dashboard; Datadog: Data trending dashboard

Documentation and Supporting Materials



- Online Practical Guide
 - [SNIA Swordfish Practical Guide](#)
- Swordfish School:
 - [Swordfish School Playlist](#) (YouTube)
- Swordfish API Specification
- Webcasts

How to Participate: Shaping the Standard



- Find pointers to the latest technical content:
 - <http://snia.org/swordfish>
 - <http://www.snia.org/publicreview#swordfish>
- Join the SSM TWG
 - By Joining the SNIA and SSM TWG, you can shape the standard: <https://members.snia.org/apps/org/workgroup/ssmtwg>
- Through the SNIA feedback portal, providing feedback on “Work In Progress”
 - As the group produces “Works In Progress”, you can provide feedback at <http://www.snia.org/feedback>



Q&A