The Case for Redfish

Jeff Hilland
President, DMTF
CT Manageability, Hewlett Packard Enterprise DCIG
June 2017
Disclaimer

• The information in this presentation represents a snapshot of work in progress within the DMTF.

• This information is subject to change without notice. The standard specifications remain the normative reference for all information.

• For additional information, see the Distributed Management Task Force (DMTF) website.
The Status Quo

Inefficient architecture
- Designed for 8-bit microarchitectures of the past
- Increased development cost from using multiple incompatible protocols & tools

High barrier to entry
- Protocols not human readable
- Significant expertise required to develop for legacy protocols
- Proprietary protocols and fragmentation from OEM extensions
- Lack of interoperability

Security Risks
- Not developed with security focus
- No security best practices deployed

Scaling Limitations
- Can’t describe modern systems (i.e. multi-node)
- Current specs do not address scale data centers
- As scale increases, the need to monitor and manage efficiently increases exponentially

Outdated Tools
- Layers needed to adapt to the current tool chain.
- Special Utilities, libraries & reformatting needed to meet customer needs.
- Layers on layers
- Inefficiencies in representation and number of IOs
What we did to fix the Status Quo

- Bring forth a modern standard that advances capabilities started by IPMI, SMASH, DASH, etc.
- Build a consortium of industry leaders using an existing standards body
- Strive for broad adoption of specification across industry to meet growing customer demand
- Seed ecosystem with open source
- Unprecedented level of interoperability
- Expand scope to rest of Software Defined IT Infrastructure over time
How to Re-invent IT Management

- Well defined management interface that exposes information directly to the **modern tool chain**
- **Scalable** and **extensible** data model purpose built for managing Software Defined Hybrid IT
- **Leverage** web standards, best practices & **security** methods.
- **Human readable but machine capable**

Interoperability is key
A Hybrid IT Management Solution

Design Tenets
- Leverage common Internet / Web Services standards, other standards where appropriate
- Represent modern hardware designs (standalone to scale-out, current silicon, OCP)
- Does not require a PhD to design or use.
- Separation of protocol from data model, allowing them to be revised independently

Protocol Suite
- HTTPS / SSL: Primary data transport
- SSDP from uPnP: Service Discovery
- HTTP-based alert subscription
- Leverage OData v4

REST & JSON
- Modern, standards-based
- Widely used for web services, software defined and public APIs
- Easy for IT professionals and amateurs to utilize

Data Model
- Schema-based, starting with CSDL & JSON Schema
- Prepare to add schema language definitions as market changes
- An easy to use data model that a human can just read
- Create new modeling tenants to facilitate ease of design (inheritance by copy, polymorphism by union)
Why REST, HTTP and JSON?

- **REST**: The API architecture
  - Rapidly replacing SOAP
- **HTTPS**: The Web protocol
  - Well-understood by admins
  - Known security model
  - Known network configuration
- **JSON**: Modern data format
  - Human-readable
  - Simpler than XML
  - Modern language support
- The combination of language support and ubiquity of REST, HTTP and JSON means that IT management tasks can be performed using the same skill set and tool chain as all other IT and dev/ops tasks.
Scalable Platforms Management Forum (DMTF group that defines Redfish)

Co-Chairs: Jeff Autor (HPE), Mike Raineri (Dell)

SPMF Leadership Companies

- Broadcom
- Cisco
- Dell
- Ericsson
- Hewlett Packard Enterprise
- Inspur
- Intel
- Lenovo
- Supermicro
- Vertiv
- VMware

SPMF Supporting Companies


SPMF Industry Alliance Partners & efforts

- OpenCompute Project – Collaborating on profile definition
- UEFI – Collaborating on Firmware Update and Host Interface work
- SNIA – Collaborating on Storage modeling / alignment between SNIA SSM and Redfish
- TGG – Pursuing relationship to work on Power/Cooling (existing DMTF Alliance Partner)
- IETF – working on Switch modeling (no official alliance)

www.dmtf.org
What is Redfish?

- Industry Standard Software Defined Management for Converged, Hybrid IT
  - HTTPS in JSON format based on OData v4
  - Schema-backed but human-readable
  - Equally usable by Apps, GUIs and Scripts
  - Extensible, Secure, Interoperable
- Version 1 focused on Servers
  - A secure, multi-node capable replacement for IPMI-over-LAN
  - Represent full server category: Rackmount, Blades, HPC, Racks, Future
  - Intended to meet OCP Remote Machine Management requirement
- Expand scope over time to rest of IT infrastructure
  - Additional features coming out approximately every 4 months
  - Working with SNIA to cover more advanced Storage (Swordfish)
  - Working with The Green Grid to cover Facilities (Power/Cooling)
  - Work with the IETF to cover some level of Ethernet Switching
Timeline of Redfish™ Specification

- The DMTF Redfish technology
  - Sep 2014: SPMF Formed in DMTF.
    - Released multiple work-in-progress for public feedback
  - Aug 2015: Redfish Specification with base models (v1.0)
  - May 2016: Models for BIOS, disk drives, memory, storage, volume (2016.1)
  - Aug 2016: Models for endpoint, fabric, switch, PCIe device, zone, software/firmware inventory & update (2016.2)
  - Dec 2016: Adv. communications devices (multi-function NICs), host interface (KCS replacement), privilege mapping (2016.3)
  - May 2017: Composability (2017.1)

- Alignment with other standard organizations
  - Aug 2016: SNIA releases model for network storage services (Swordfish)
  - DMTF created work registers with UEFI Forum, TGG and OCP
  - Working with IETF to create a programmatic Redfish mapping
Redfish Developer Hub: redfish.dmtf.org

- **Resources**
  - Schema Index
  - Specifications
  - GitHub for Redfish Tools
  - Registries
  - Other Documentation

- **Mockups**
  - Simple Rack-mounted Server
  - Bladed System
  - Proposed OCP Redfish Profile
  - More being added

- **Education/Community**
  - Redfish User Forum
  - Whitepapers, Presentations
  - YouTube shorts & Webinars
## Redfish Task Forces

Task forces created when there is sufficient need or interest

<table>
<thead>
<tr>
<th>Task Forces</th>
<th>Product</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host Interface</td>
<td>Interface for in-band entity to access the Redfish service</td>
<td>Complete</td>
</tr>
<tr>
<td>Firmware Update</td>
<td>Resource models for performing firmware updates</td>
<td>Complete</td>
</tr>
<tr>
<td>Storage</td>
<td>Resource models for managing local storage devices</td>
<td>Complete</td>
</tr>
<tr>
<td>Advanced Comm Device</td>
<td>Resource models for managing network ports</td>
<td>Complete</td>
</tr>
<tr>
<td>Privilege Mapping</td>
<td>Model extensions for expressing user privilege</td>
<td>Complete</td>
</tr>
<tr>
<td>Composability</td>
<td>Resource models for composing resource</td>
<td>Model released, ongoing</td>
</tr>
<tr>
<td>Network Infrastructure</td>
<td>Resource models for managing an Ethernet Switch model, derived from YANG models</td>
<td>WIP released, ongoing</td>
</tr>
<tr>
<td>Tools</td>
<td>Open source for using and developing Redfish models</td>
<td>Released 10 tools to date</td>
</tr>
<tr>
<td>Interoperability</td>
<td>Develop interoperability tools &amp; hold plugfests</td>
<td>Ongoing</td>
</tr>
</tbody>
</table>
## Tools Task Force: Redfish Tools Description

SPMF Tools TF open source tools to enable Redfish  [http://www.dmtf.org/standards/opensource](http://www.dmtf.org/standards/opensource)

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Extend</strong></td>
<td></td>
</tr>
<tr>
<td>CSDL Validator</td>
<td>Validates the CSDL conforms to Redfish requirements</td>
</tr>
<tr>
<td>CSDL-to-JSON schema convertor</td>
<td>Generates json-schema files from CSDL</td>
</tr>
<tr>
<td>Document Generator</td>
<td>Generates documentation from json-schema</td>
</tr>
<tr>
<td><strong>Working Svc</strong></td>
<td></td>
</tr>
<tr>
<td>Mockup Server</td>
<td>Exposes a mockup as a static HTTP service (GETs only)</td>
</tr>
<tr>
<td>Mockup Creator</td>
<td>Creates a mockup from a Redfish service</td>
</tr>
<tr>
<td>Profile Simulator</td>
<td>Dynamic simulator of the proposed Redfish profile for OCP</td>
</tr>
<tr>
<td>Interface Emulator</td>
<td>Dynamic simulator which can rapid</td>
</tr>
<tr>
<td><strong>Test</strong></td>
<td></td>
</tr>
<tr>
<td>Service Validator</td>
<td>Validates a Redfish service is conformant</td>
</tr>
<tr>
<td>Service Conformance Tool</td>
<td>Verifies the conformance of a Redfish service to assertions</td>
</tr>
<tr>
<td></td>
<td>extracted from the Redfish Specification</td>
</tr>
<tr>
<td><strong>Client</strong></td>
<td></td>
</tr>
<tr>
<td>Command Line (redfishtool)</td>
<td>A command line tool for interacting with a Redfish service</td>
</tr>
<tr>
<td></td>
<td>(similar to ipmitool)</td>
</tr>
<tr>
<td>Python Utility &amp; Python Library</td>
<td>A Command line tool with UI and python libraries for</td>
</tr>
<tr>
<td></td>
<td>interacting with Redfish services</td>
</tr>
</tbody>
</table>
Thank you!

Redfish

www.dmtf.org
Backup Material
How simple is Redfish?

Example Python code to retrieve serial number from a server:

```python
rawData = urllib.urlopen('http://192.168.1.135/redfish/v1/Systems/1')
jsonData = json.loads(rawData)
print( jsonData['SerialNumber'] )
```

Output is:

```
1A87CA442K
```

Three lines of code: point to the resource, get the data, print the serial number.

*Example uses Redfish ComputerSystem resource*
GET http://<ip-addr>/redfish/v1/Systems/{id}/Processors/{id}

Use the Redfish Resource Explorer (redfish.dmtf.org) to explore the resource map
Redfish Releases 2016

- **2016.1**:
  - BIOS Configuration
  - Memory (DIMM) Inventory
  - Secure Boot
  - Advanced Storage (SAS, SATA, RAID, NVMe)

- **2016.2**
  - PCIe Devices
  - Switched Fabrics
  - Persistent Memory Management
  - Installed Firmware and Firmware Update
  - ActionInfo to describe Action Parameters

- **2016.3**
  - Advanced Communications Devices (multi-function NICs)
  - Redfish Host Interface (KCS replacement)
  - Resource and Property-level Privilege Mapping

- **2017.1**
  - Composability
Retrieving “IPMI class” data
- Basic server identification and asset info
- Health state
- Temperature sensors and fans
- Power supply, power consumption and thresholds

Basic I/O infrastructure data
- Host NIC MAC address(es) for LOM devices
- Simple hard drive status / fault reporting

Perform Common Actions
- Reboot / power cycle server
- Change boot order / device
- Set power thresholds

Discovery
- Service endpoint (network-based discovery)
- System topology (rack/chassis/server/node)

Access and Notification
- Serial console access via SSH
- Alert / event notification method(s)
- Event Log access method(s)

Security
- Session-based leverages HTTPS

BMC infrastructure
- View / configure BMC network settings
- Manage local BMC user accounts

Working on more…
Network Infrastructure task force: YANG to Redfish

- Enable converged infrastructure management
  - One interface (one tool chain) to manage compute, storage and network
  - Switches have platform components in common with servers and storage
  - Network Functions Virtualization (NFV) will need common manageability for compute and networking

- DMTF wants to leverage the networking industry's expertise
  - YANG is the basis for general network industry manageability
  - Large body of existing YANG work
  - Model driven approach to network management