Introduction to Redfish

DMTF Scalable Platforms Management Forum
May 2016

www.dmtf.org

Information & Disclaimer

- The information in this presentation represents a snapshot of work in progress within the DMTF.
- This information is subject to change. The Standard Specifications remain the normative reference for all information.
- DMTF Specifications in progress are considered DMTF confidential and are being shown under existing work register agreements.
- Documents cannot be shared outside of the DMTF work group. The documents may be posted on the WG site, but not otherwise distributed without DMTF Board approval.
- Non-member comments, feedback and submissions are subject to the DMTF Patent Policy statement

What is Redfish?

- **Industry Standard RESTful API for IT Infrastructure**
  - HTTPS in JSON format based on Odata v4
  - Equally usable by Apps, GUIs and Scripts
  - Schema-backed but human-readable
- **Version 1 focused on Servers**
  - A secure, multi-node capable replacement for IPMI-over-LAN
  - Add devices over time to cover customer use cases & technology
    - PCIe Switching, Local Storage, NVDIMMs, Multifunction Adapters, Composability
  - Intended to meet OCP Remote Machine Management requirements
- **Expand scope over time to rest of IT infrastructure**
  - Working with SNIA to cover more advanced storage.
  - Plan on working with partners like the Green Grid to cover Power/Cooling.
  - Goal is to accommodate or map existing switch standards over time.

Scalable Platforms Management Forum (DMTF Group that Defines Redfish)

DMTF

Co-Chairs: Jeff Autor (HPE), Paul Vancil (Dell)

Leadership Companies

BROADCOM. DELL EMC² EMERSON. Network Power Hewlett Packard Enterprise inspur

intel lenovo. FOR THOSE WHO DO. Microsoft SUPERMICRO vmware

Supporting Companies

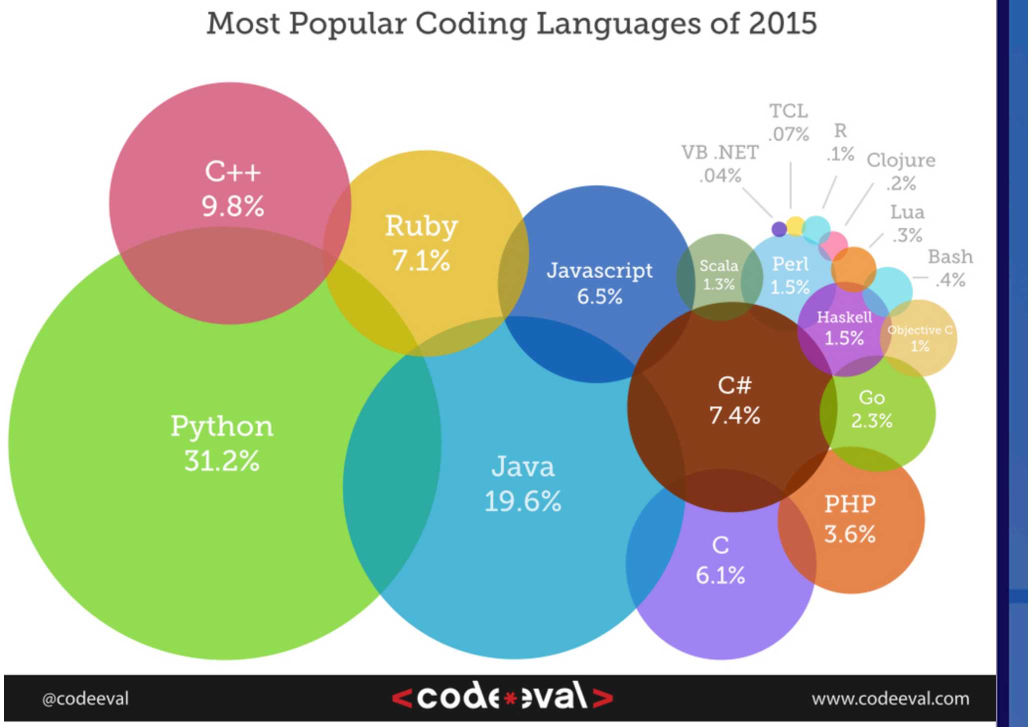
AMI, Cisco, Fujitsu, Western Digital, Huawei, IBM, Insyde Software, Mellanox, NetApp, Oracle, Microsemi, Qualcomm, Seagate

Industry Alliance Partners

- OpenCompute Project
- UEFI - Collaborating on Firmware Update and Host Interface work
- SNIA – Collaborating on Storage modeling/alignment between SSM and Redfish
- TGG – Pursuing relationship to work on Power/Cooling (existing DMTF Alliance Partner)

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



Most Popular Coding Languages of 2015

- The combination of language support and ubiquity of REST, HTTP and JSON means that systems management tasks can be performed using the same skill set and tool chain as all other IT and dev/ops tasks.

How simple is REST using JSON?

Example Python code to retrieve serial number from a server:

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

Output is:

```
1A87CA442K
```

***Example uses Redfish ComputerSystem resource,**
Authentication not shown

Redfish v1.0 Feature Set

Retrieve "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

Discovery

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

Security

- Session-based leverages HTTPS

Perform Common Actions

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

Access and Notification

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

BMC infrastructure

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

Working on more…

Redfish releases

- v1.0 Released August 2015
  - Specification and Schema files
- v1.1 Release November 2015
  - Spec 1.0.1, Schema 1.1, 1.0.1 and 1.0
- 2016.1 Release April, 2016
  - Spec 1.0.2, Schema 1.2, 1.1, 1.0.2 and 1.0
- Releases planned for Schema and Specification
  - 2016.2 - Summer 2016 (August)
  - 2016.3 - Fall 2016 (December)
- Expecting 3 releases per year.
  - Each release will have updates, errata and additions that are ready at that time
- Download specification, white paper, FAQ, schemas, Works in Progress:
  - http://www.dmtf.org/standards/redfish

Redfish 2016 Release 1

- Redfish Specification v1.0.2 (errata)
- **NEW** Schemas v1.0.0
  - AttributeRegistry, Bios, Drive, Memory, MemoryCollection, MemoryMetrics, SecureBoot, Storage, StorageCollection, Volume
- Schemas v1.1.0 or v1.2.0 (minor revs)
  - Chassis, ComputerSystem, Event, Manager, Power, Resource, SimpleStorage, Thermal
- Schemas v1.x.2 (errata)

Schema Release Bundles (posted at http://www.dmtf.org/standards/redfish)

DSP8010_2016.1: All new, minor, and errata 2016.1 schemas

Individual file contents also posted on http://redfish.dmtf.org/schemas

DSP8010_2016.0.9a: Work in Progress

New schemas under development within SPMF

Expected Redfish Open Source Efforts

Conformance Tool Client Library

- Common utility support functions
  - Discovery, Enumeration, etc.
  - Event subscription
- Typical tasks
  - Power on/off/reboot
  - Gather thermal data
- Languages under consideration
  - Python
  - Java
  - PowerShell
  - Other possibilities…

Command Line Utility

- Similar to IPMItool
- Designed for end users
- Calls Client library
- Likely written in Python

Conformance Test Suite

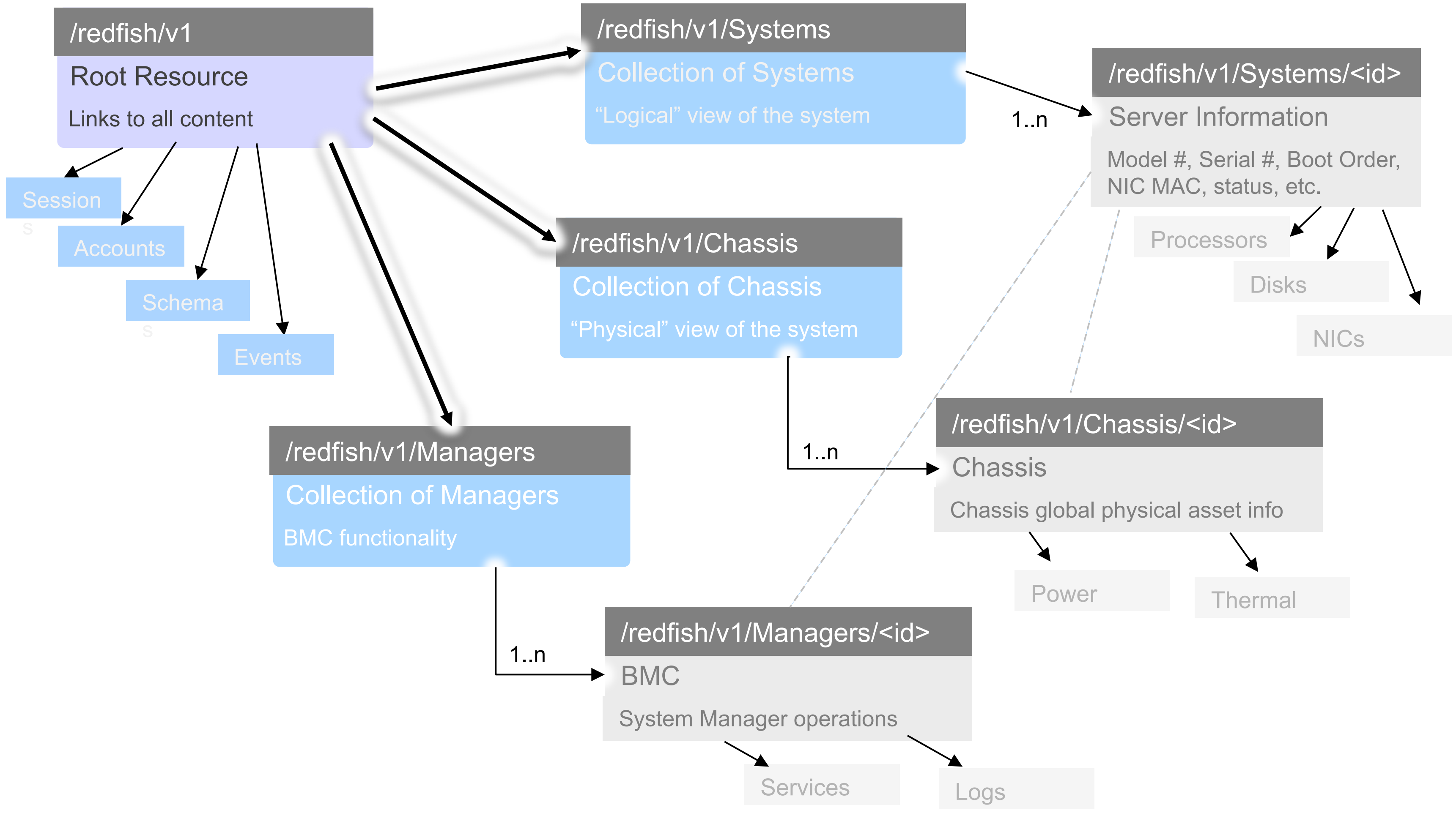
- Schema validation (JSON and CSDL)
- Payload validation
- Spec and Schema conformance
- Checklist for vendors and customers
- Avoid spec interpretation conflicts

Redfish

Introduction to the Redfish data model

- All resources linked from a Service Entry point (root)
  - Always located at URL: /redfish/v1
- Major resource types structured in ‘collections’ to allow for standalone, multi-node, or aggregated rack-level systems
  - Additional related resources fan out from members within these collections
- **ComputerSystem**: properties expected from an OS console
  - Items needed to run the “computer”
  - Roughly a logical view of a computer system as seen from the OS
- **Chassis**: properties needed to locate the unit with your hands
  - Items needed to identify, install or service the “computer”
  - Roughly a physical view of a computer system as seen by a human
- **Managers**: properties needed to perform administrative functions
  - aka: the systems management subsystem (BMC)

Resource map (highlights)



/redfish/v1

Root Resource

Links to all content

Sessions

Accounts

Schemas

Events

/redfish/v1/Systems

Collection of Systems

“Logical” view of the system

1..n

/redfish/v1/Systems/<id>

Server Information

Model #, Serial #, Boot Order, NIC MAC, status, etc.

Processors

Disks

NICs

/redfish/v1/Chassis

Collection of Chassis

“Physical” view of the system

1..n

/redfish/v1/Chassis/<id>

Chassis

Chassis global physical asset info

Power

Thermal

/redfish/v1/Managers

Collection of Managers

BMC functionality

1..n

/redfish/v1/Managers/<id>

BMC

System Manager operations

Services

Logs

More on Redfish:

- Join the SPMF
  - Help shape the standard
  - http://www.dmtf.org/join/spmf
- We want your Feedback
  - On the Standard or Works in Progress
  - http://www.dmtf.org/standards/feedback
- Redfish Standards
  - Schemas, Specs, Mockups, White Papers, FAQ, Educational Material & more
  - http://www.dmtf.org/standards/redfish
- Redfish Developer Portal
  - Redfish Interactive Explorer, Hosted Schema at Namespace & other links
  - http://redfish.dmtf.org
- SPMF (Working Group that defines Redfish)
  - Companies involved, upcoming schedules & future work, charter, information on joining.
  - http://www.dmtf.org/standards/spmf