Redfish CSDL Usage

Mike Raineri

Dell Inc.

Redfish

Standard OData Annotations Used by Redfish

- Core annotations
  - OData.Description: Informative documentation
  - OData.LongDescription: Normative documentation
  - OData.Permissions: Client's ability to read/write a property
  - OData.AdditionalProperties: If implementations can add more properties
  - OData.AutoExpandReferences: Expands the navigation property reference
  - OData.AutoExpand: Service will expand entities even if not requested
- Measures annotations
  - Measures.Unit: Documents units for a value; UCUM notation used
- Capabilities annotations
  - Capabilities.InsertRestrictions: If a client can add to a collection
  - Capabilities.UpdateRestrictions: If a client can modify a resource
  - Capabilities.DeleteRestrictions: If a client can remove from a collection

Base Redfish Schemas

- RedfishExtensions_v1.xml
  - Defines Redfish specific annotations
- Resource_v1.xml
  - Defines base class for all resources
    - Resource and ResourceCollection
  - Defines common properties and types
    - Id, Name, Description, Oem
    - Links object base definition
    - Status object definition
    - Location object definition
    - Common enumerated lists

Defining Redfish Resources

- Resources are singular entities, such as Thermal
- All Resources inherit from Resource.v1_0_0.Resource
- Id is used as the key property
- Name is mandatory, but Description is optional
- Many resources define a Links property that inherits from the Resource's definition of Links
  - Links is an object that only contains navigation properties
  - The links are references to related resources
- A generic Oem object is made available for implementations to extend upon the schema as needed

Defining Redfish Resource Collections

- Resource Collections contain a set of resources of the same type, such as the Chassis Collection containing the different Chassis Resources
- All Resource Collections inherit from Resource.v1_0_0.ResourceCollection
- Name is used as the key property
- Description is optional
- All Resource Collections define a Members property as an array of the Resource type they list

Redfish Schema Versioning

- All Resources have a specific naming convention with their namespaces
  - The first namespace is “unversioned” and contains no properties
  - Subsequent namespaces are versioned, and inherit from each other
- Namespaces are named as:
  - Unversioned: *ResourceName*
  - Versioned: *ResourceName.vX_Y_Z*
    - X is the major version
    - Y is the minor version
    - Z is the errata version
- Adding a property will generate a new namespace with a new minor version
- Correcting an existing file will generate a new namespace with a new errata version

Relationship to JSON Schema

- CSDL schemas are used to generate JSON schemas via scripts
- Different number of files
  - CSDL: one file per resource type (Manager_v1.xml)
  - JSON: one file per version per resource type (Manager.json, Manager.v1_0_0.json, Manager.v1_0_1.json, etc)
- CSDL and JSON schemas are intended to be functionally equivalent

Thank you for watching!

- Redfish Standards
  - Schemas, Specs, Mockups, White Papers, FAQ, Educational Material & more
  - http://www.dmtf.org/standards/redfish
- Redfish Developer Hub
  - Redfish Interactive Explorer, Hosted Schema at Namespace & other links
  - http://redfish.dmtf.org
- SPMF (WG that defines Redfish)
  - Companies involved, Upcoming Schedules & Future work, Charter, Information on joining.
  - http://www.dmtf.org/standards/spmf