# Redfish Release 2020.3

DMTF Redfish Forum

September 2020

Redfish Release 2020.3

- Redfish Specification v1.11.1
  - Errata release
- Redfish Host Interface Specification v1.3.0
  - Adds bootstrap credential method
- Redfish Schema Bundle 2020.3
  - DSP8010 contains all released Redfish schemas
- Redfish Message Registry Bundle 2020.3
  - DSP8011 contains all released Redfish registries
  - NEW Job and LogService message registries
  - Updated Privilege registry
  - Updated Base message registry to include argument descriptions for all messages
- 2 NEW Schemas, 37 updated schemas
  - Addition of Connection and StorageController schema
  - Support for NVMe-over-Fabric
  - Support for collecting and retrieving DiagnosticData ("crash dumps")
  - Migration from NetworkPort to Port schema for consistency throughout model
  - Added support for per-system remote console and virtual media instances, expanded details of serial console
  - See release notes in DSP8010 for errata details
- Download all published material at: http://www.dmtf.org/standards/redfish

www.dmtf.org
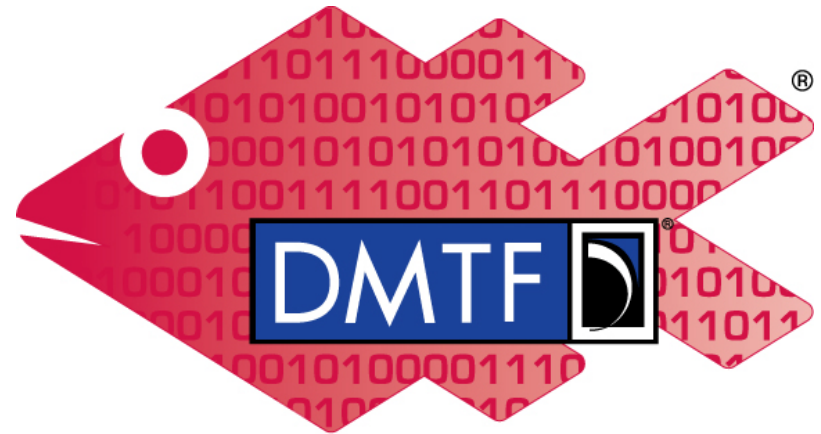
NEW Host Interface Specification Updates

- New credential bootstrapping feature added to the Redfish Host Interface Specification v1.3

- Added a set of IPMI commands to allow host software or the operating system to generate credentials for itself without requiring a-priori knowledge about the Redfish service

  - Previous UEFI variable mechanism for transporting host credentials has been deprecated in favor of this approach

- Additions to the *HostInterface* resource to control whether the credential bootstrapping feature is enabled, and control the role assigned to the bootstrap account

NEW Connection schema

- Added *Connection* to the fabric model
- A *Connection* resource is used to express the types of resources an endpoint can access when connected to another endpoint within a fabric
  - For example, if a storage initiator connects to a storage target, which volumes can the initiator access
- *Connection* maps *Endpoint* resources to other resources, and specifies the permissions for the resource

NEW StorageController schema

- Created *StorageController* to address scalability concerns
  - The Storage schema currently contains an object array for controllers
    - This property works for traditional HBAs or RAID adapters that have a physical controller, where the *Storage* resource contains only one or two controllers
  - Added a link in *Storage* to a *StorageController* resource collection
- A *StorageController* resource collection allows for cases where a single storage subsystem will have a more dynamic set of controllers
  - For NVMe-oF, controllers are created and retired as NVMe-oF hosts connect or disconnect from NVMe-oF targets

NEW NVMe-oF modeling using Redfish

- Additions made to the data model to support NVMe-oF
  - *Storage* added to *ServiceRoot* for showing externally accessible storage subsystems
  - *Connection* added to *Fabric* to allow mapping of *Endpoints* to other resources, such as *Volumes*
  - *Drives* added to *Chassis* to show drives within an enclosure
- *Endpoints* are used to represent NVMe-oF hosts and interfaces on an NVMe-oF subsystem
  - An *Initiator Endpoint* represents a host, and contains the host's NQN
  - A *Target Endpoint* represents an interface to the subsystem, and contains the subsystem's NQN as well as a reference to the *NetworkDeviceFunction* representing the physical interface

NEW NVMe-oF Modeling in Redfish

www.dmtf.org
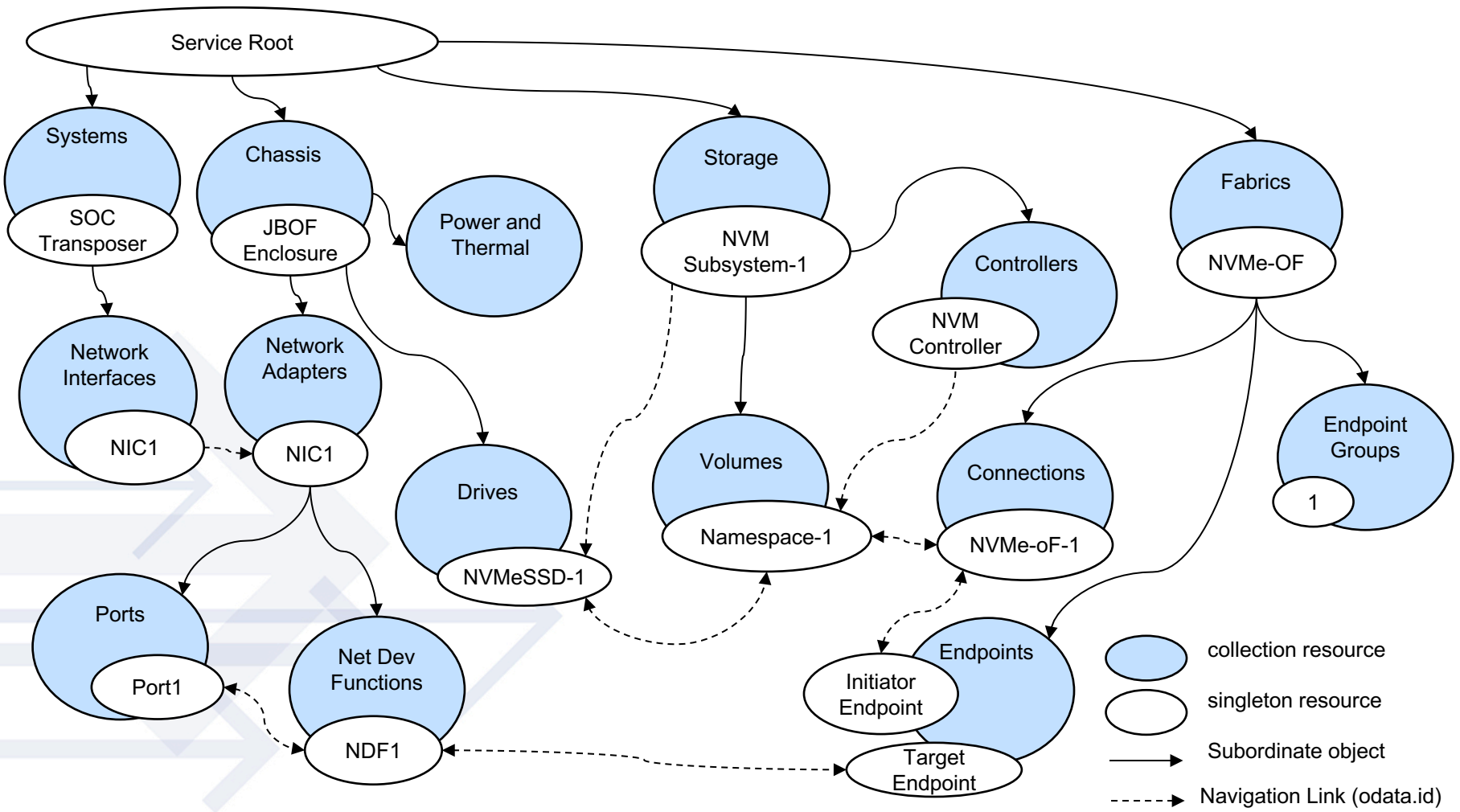
Port schema updates, migration from NetworkPort

- Large overlap between *Port* and *NetworkPort*
  - Both schemas filled essentially the same purpose
  - Standardize on *Port* going forward
  - Properties added to *Port* that were present only in *NetworkPort*
- References to *NetworkPort* deprecated and references to *Port* added
  - *NetworkAdapter*
  - *NetworkInterface*
  - *NetworkDeviceFunction*
- Autospeed negotiation for network adapters now explicitly supported
  - *AutoSpeedNegotiationEnabled* - allows turning negotiation on/off
  - *ConfiguredNetworkLinks* - set of lane speed/count possibilities to which autospeed negotiation is restricted

NEW Diagnostic Data support in LogService / LogEntry

- Diagnostic data can be collected at any time and is stored in a file to troubleshoot problems that have occurred

- The data may consist of a "crash dump", application core, network configuration, system inventory configuration, journal log, etc.

- *LogService CollectDiagnosticData* action collects the diagnostic data
  - Parameters *DiagnosticDataType* and *OEMDiagnosticDataType*

- *LogEntry* adds support for referencing a diagnostic data file
  - *AdditionalDataSizeBytes* - Size of the additional data file
  - *AdditionalDataURI* - URI of the additional data file
  - *DiagnosticDataType* - Type of diagnostic data
    - Types include: *Manager, OEM, OS,* and *PreOs*
  - *OEMDiagnosticDataType* – OEM-defined diagnostic data type
    - Used when *DiagnosticDataType* is *OEM*

NEW InfiniBand support

- Added *InfiniBand* to *NetDevFuncType* in *NetworkDeviceFunction* to be able to identify network devices that support InfiniBand

- Added *InfiniBand* property to *NetworkDeviceFunction* for containing InfiniBand specific properties
  - Port GUID
  - Node GUID
  - System GUID
  - MTU capabilities and configuration

NEW *LocationIndicatorActive* property

- Deprecates and replaces *IndicatorLED* property throughout the model
  - *IndicatorLED* improperly exposed hardware implementation details
    - Redfish attempts to model usage semantics rather than physical characteristics
  - This caused interoperability issues discovered during Redfish Plugfests
    - Some vendors use "On" as the active state, others use "Blinking"
    - Some vendors use "Off" as the inactive state, other use "On"

- *LocationIndicatorActive* property is a simple Boolean
  - True if the locator function is active, False if it is not in use
  - Added to *Assembly, Circuit, Chassis, ComputerSystem, Drive, Memory, Outlet, PCIeSlots, Port, Processor, Switch*

- For compatibility with existing clients, services should:
  - Implement both properties in ComputerSystem, Chassis, Drive, Switch instances where *IndicatorLED* already exists
    - Reflect changes (from user PATCH operations) in both properties
  - Replace *IndicatorLED* support in Circuit, Outlet as these are new schemas

Remote console and Virtual Media updates

- Remote console and virtual media properties moved to *ComputerSystem*
  - Allows clients to locate functionality in multi-node
- New *VirtualMediaConfiguration* object, *VirtualMediaCollection* link
  - Configuration object provides connection information per system
  - No changes to *VirtualMedia* resource collection definition
- *SerialConsole*
  - Support allows for multiple console protocols
    - SSH, Telnet, IPM-over-LAN Serial-over-LAN supported
  - *SharedWithManagerCLI* indicates console also allows access to Manager
  - *ConsoleEntryCommand* reports the command sequence client issues to access the host console once connected using the given protocol
  - *HotKeySequenceDisplay* provides text to client software to display to user showing the keyboard "hotkey" sequence used to exit the host console

- **AddressPool v1.1.0**
  - Added *Ethernet* property
- **Assembly v1.3.0**
  - Added *LocationIndicatorActive* and *Location* properties
- **Chassis v1.14.0**
  - Added *LocationIndicatorActive* and *Drives* properties
  - Deprecated *IndicatorLED* property in favor of *LocationIndicatorActive*
- **Circuit v1.1.0**
  - Added *LocationIndicatorActive* property
  - Deprecated *IndicatorLED* property in favor of *LocationIndicatorActive*
- **ComputerSystem v1.13.0**
  - Added *LocationIndicatorActive, BootProgress, SerialConsole, GraphicalConsole, VirtualMediaConfig, VirtualMedia, PowerOnDelaySeconds, PowerOffDelaySeconds,* and *PowerCycleDelaySeconds properties*
  - Deprecated *IndicatorLED* property in favor of *LocationIndicatorActive*

- Drive v1.11.0
  - Added *LocationIndicatorActive* property
  - Deprecated *IndicatorLED* property in favor of *LocationIndicatorActive*
- Endpoint v1.5.0
  - Added *Manager and Connections* properties
- EndpointGroup v1.3.0
  - Initial transition from SNIA contribution to DMTF
  - Deprecated *AccessState* in favor of *AccessState* in Connection. Deprecated *Endpoints* in favor of *Endpoints* in Links. Deprecated *Client* and *Server* in favor of *Initiator* and *Target*. Added *Connections* to *Links*
  - Added *Connections* property
- Event v1.6.0
  - Added *SpecificEventExistsInGroup* property

Redfish Schema Minor Revisions (3 of 7)

- EventDestination v1.9.0
  - Added *Certificates*, *VerifyCertificate, SyslogFilters, OEMProtocol,* and *OEMSubscriptionType* properties
  - Added syslog-related values to *Protocol* and *SubscriptionType*
- Fabric v1.2.0
  - Added *EndpointGroups and Connections* properties
- HostInterface v1.3.0
  - Added *CredentialBootstrapping* property and *CredentialBootstrappingRole* to *Links*.
  - Deprecated *KernelAuthRoleId*, *KernelAuthEnabled*, *FirmwareAuthRoleId*, *FirmwareAuthEnabled*, *KernelAuthRole*, and *FirmwareAuthRole* in favor of new credential bootstrapping properties
- LogEntry v1.7.0
  - Added *AdditionalDataSizeBytes, AdditionalDataURI, DiagnosticDataType, OEMDiagnosticDataType* properties
- LogService v1.2.0
  - Added *SyslogFilters* property and *CollectDiagnosticData* action

www.dmtf.org

Redfish Schema Minor Revisions (4 of 7)

- Manager v1.10.0
  - Added *TimeZoneName* property
  - Deprecated *VirtualMedia* and *SerialConsole* properties in favor of those added to ComputerSystem
- Memory v1.10.0
  - Added *LocationIndicatorActive property*
- MemoryChunks v1.4.0
  - Added *DisplayName property*
- MemoryMetrics v1.4.0
  - Added *CorrectableECCErrorCount* and *UncorrectableECCErrorCount* properties to *CurrentPeriod* and *LifeTime*
- MetricDefinition v1.1.0
  - Added *OEMCalculationAlgorithm* property and OEM *value to CalculationAlgorithm*
- NetworkAdapter v1.5.0
  - Deprecated *NetworkPorts* property in favor of *Ports* to leverage the common Port schema

www.dmtf.org

Redfish Schema Minor Revisions (5 of 7)

- NetworkDeviceFunction v1.5.0
  - Added *Infiniband* property and *MTUSizeMaximum* to *Ethernet*
  - Deprecated *AssignablePhysicalPorts* and *PhysicalPortAssignment* properties in favor of AssignablePhysicalNetworkPorts and PhysicalNetworkPortAssignment to leverage the common Port schema

- NetworkInterface v1.3.0
  - Deprecated *NetworkPorts* property in favor of *Ports* to leverage the common Port schema

- NetworkPort v1.3.0
  - Added *Starting* and *Training* values to LinkStatus

- Outlet v1.1.0
  - Added *LocationIndicatorActive* property
  - Deprecated *IndicatorLED* property in favor of *LocationIndicatorActive*

- PCIeDevice v1.5.0
  - Added *UUID* property

- ## PCIeSlots v1.4.0
  - Added *LocationIndicatorActive* property
- ## Port v1.3.0
  - Added *LocationIndicatorActive, MaxFrameSize, LinkConfiguration, FibreChannel,* and *Ethernet* properties
- ## Processor v1.10.0
  - Added *LocationIndicatorActive, BaseSpeedMHz, SpeedLimitMHz,* and *SpeedLocked* properties
  - Added *ProtectedIdentificationNumber* to *ProcessorId*
- ## Protocol
  - Added *Ethernet* and *Infiniband* values
- ## Resource v1.10.0
  - Added *NGUID* to *DurableName*
- ## ServiceRoot v1.9.0
  - Added *Storage* property

- Session v1.3.0
  - Added *ClientOriginIPAddress* property
- SimpleStorage v1.3.0
  - Added *Storage* to *Links*
- Storage v1.9.0
  - Added *Controllers* and *Identifiers* properties
  - Added *SimpleStorage* and *StorageServices* to *Links*
- Switch v1.4.0
  - Added *LocationIndicatorActive, CurrentBandwidthGbps,* and *MaxBandwidthGbps* properties. Added *PCIeDevice* to *Links*
  - Deprecated *IndicatorLED* property in favor of *LocationIndicatorActive*
- Task v1.5.0
  - Added *SubTasks* property
- Zone v1.5.0
  - Added *AddEndpoint* and *RemoveEndpoint* actions

Errata Schema Revisions (1 of 5)

- ## AccountService v1.x.x
  - Clarified that *MinPasswordLength* and *MaxPasswordLength* do not apply to external accounts
  - Fixed writable permission for *MinPasswordLength* and *MaxPasswordLength*
  - Clarified that *AuthFailureLoggingThreshold* applies per account

- ## AttributeRegistry v1.x.x
  - Clarified the evaluation order for *Dependencies* containing multiple logic expressions.

- ## Chassis v1.x.x
  - Added recommendation for using *@Redfish.AllowableValues* for *ResetType* for compatibility with older clients

Errata Schema Revisions (2 of 5)

- ## ComputerSystem v1.x.x
  - Removed restriction in the *UefiTargetBootSourceOverride* to allow for *Continuous* to be used with *UefiTarget*
  - Clarified that *ProcessorySummary* only contains processors that execute system code
  - Added recommendation for using @*Redfish.AllowableValues* for *ResetType* for compatibility with older clients

- ## EthernetInterface v1.x.x
  - Clarified several descriptions to state the difference between IP addresses that are currently in use versus addresses that are requested to be assigned

- ## Event v1.x.x
  - Clarified that *Severity* and *MessageSeverity* in responses can be service-defined and not come from a message registry

Errata Schema Revisions (3 of 5)

- ## IPAddresses v1.x.x
  - Clarified the description of the IPv4 *Gateway* property to show that there should only be one gateway per interface if multiple addresses are allowed

- ## Job v1.x.x
  - Clarified that steps of jobs cannot have their own steps

- ## LogEntry v1.x.x
  - Corrected the description for *SensorNumber* to show that it only applies for entries of type SEL

- ## Manager v1.x.x
  - Added recommendation for using @*Redfish.AllowableValues* for *ResetType* for compatibility with older clients

Errata Schema Revisions (4 of 5)

- ManagerNetworkProtocol v1.x.x
  - Clarified the SNMP *AuthenticationProtocol* property does not apply when *EncryptionProtocol* contains the value *Account* and instead is determined based on the settings of individual accounts.
  - Clarified usage of *VirtualMedia* and *KVMIP* properties that they apply to all systems managed by the manager

- Message, MessageRegistry v1.x.x
  - Clarified that *Severity* and *MessageSeverity* in responses can be service-defined and not come from a message registry

- MetricDefinition v1.0.5
  - Clarified *CalculationParameters* to show it should be used when *ImplementationType* is *Synthesized* or *Calculated*

- MetricReportDefinition v1.x.x
  - Corrected permissions for *ReportActions*, *ReportUpdates*, *MetricId*, and properties in *Wildcards* to be writable
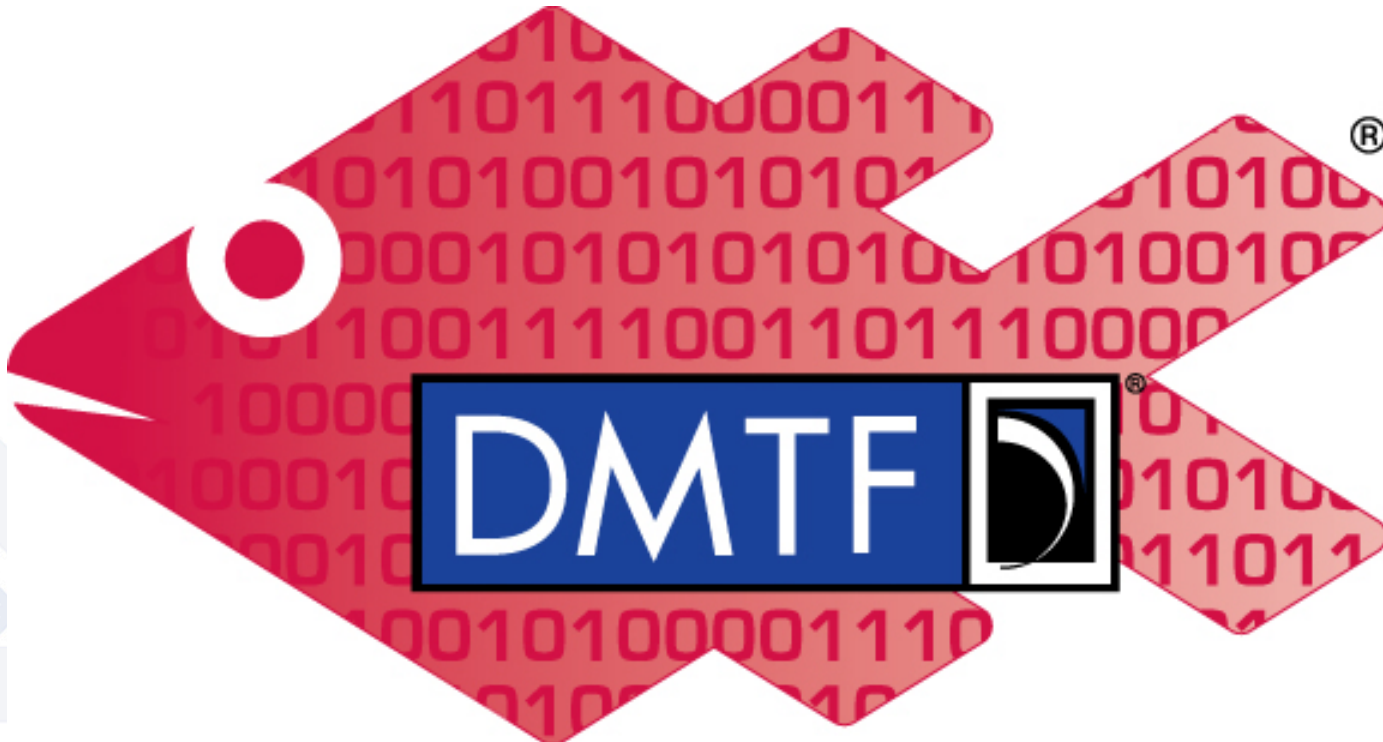
- OperatingConfig v1.0.1
  - Clarified the description for BaseSpeedMHz to show it's the nominal speed
- Port v1.x.x
  - Clarified the descriptions for *Starting* and *Training*
- Settings v1.x.x
  - Clarified usage of the *SupportedApplyTimes* property to state that having a single member is valid for conveying a non-configurable apply time
- UpdateService v1.x.x
  - Clarified the descriptions of *FirmwareInventory* and *SoftwareInventory* to explain the types of members to find in their respective collections
  - Clarified that *TransferProtocol* in *SimpleUpdate* is ignored if the scheme is provided in the URI
- (various) v1.x.x
  - Corrected permissions and update capabilities

Schema Guide, Property Guide, and Schema Supplement

- Guide documents are updated for each schema (DSP8010) release
  - Description text for properties taken directly from schema files
  - DMTF documentation generator creates these guide documents
- Redfish Resource and Schema Guide (DSP2046)
  - Presents schema definition contents in a more human-readable format
  - Includes example payloads for each schema
- Redfish Property Guide (DSP2053)
  - Provides an alphabetical list of all properties defined in Redfish schema
  - Useful for schema writers to locate existing definitions or properties
  - Helps avoid re-defining property names already in use
- Redfish Schema Supplement (DSP0268)
  - Presents normative ("LongDescription") schema definition contents
  - Useful for Redfish Service developers to ensure conformance

www.dmtf.org