Redfish Console and Virtual Media property enhancements

Redfish Forum Work In Progress
May 2020
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 DMTF website.
Providing Feedback

- Feedback to the DMTF Redfish Forum is encouraged
  - Submit items using the DMTF feedback portal
  - https://www.dmtf.org/standards/feedback
- Questions and comments can be posted on the Redfish User Forum
  - https://www.redfishforum.com
Issues

• Graphical, Serial Console and Virtual Media should be properties of a ComputerSystem, not a Manager
  • Requirement for any Redfish service to manage more than one server
  • Current design only allows for one instance per service

• SerialConsole support lacks details fully describe how functionality
  • Need to facilitate automation and utilities that launch console sessions
  • Details include:
    • Port information
    • Login information
    • Command line integration / automation strings
GraphicalConsole Proposal

- Add existing *GraphicalConsole* object to *ComputerSystem*
  - No changes to definition required
  - Direct association with a *ComputerSystem* instance solves issue
  - Deprecate usage of *GraphicalConsole* in Manager

```
"GraphicalConsole": {
    "ServiceEnabled": true,
    "MaxConcurrentSessions": 2,
    "ConnectTypesSupported": [
        "KVMIP"
    ],
},
```
VirtualMedia Proposal

- Add VirtualMediaCollection to ComputerSystem
  - No changes to existing definition required
  - Add link property VirtualMedia
  - Direct association with a ComputerSystem instance solves issue
  - May deprecate usage of VirtualMedia in Manager, but expect to see duplication to provide compatibility with existing tools and implementations

```json
"VirtualMedia": {
  "@odata.id": "/redfish/v1/Systems/1Z458TJ434/VirtualMedia"
},
```
SerialConsole Proposal

- Add *SerialConsole* object to *ComputerSystem*
  - Requirement for any Redfish service to manage more than one server
  - Provides direct association with a *ComputerSystem* instance
- Enhance *SerialConsole* with new properties to describe functionality
  - Allow multiple connection / console types
    - Embedded objects for “SSH”, “Telnet”, “IPMI”, “OEM”
  - Connection details and enable/disable function per console type
  - Describe “shared” implementations that allow switching between a console session and a command-line interface (CLI)
  - Facilitate automation and utilities that launch console sessions
SerialConsole Proposal

- **SharedWithManagerCLI** describes two types of implementations
  - True if serial console is a function in the Manager’s command-line interface
  - False if serial console is a dedicated port on Telnet or SSH
- **ConsoleEntryCommand** provides CLI arguments for a scripted connection command
  - Example: “ConsoleEntryCommand”: “**connect console 1**” used with SSH:
    - `ssh -t administrator@<Manager-hostname:Port> connect console 1`
- **HotKeySequenceDisplay** provides text to display to user as exit instructions

```
"SerialConsole": {
  "MaxConcurrentSessions": 1,
  "SSH": {
    "SharedWithManagerCLI": true,
    "Port": 22,
    "ServiceEnabled": true,
    "ConsoleEntryCommand": "connect console 1",
    "HotKeySequenceDisplay": "Press ESC- ( to exit console"
  },
  "IPMI" {
    "Port": 623;
    "ServiceEnabled": true
    "HotKeySequenceDisplay": "Press ~. to exit console"
  }
}
```

**Dedicated Port example:**
```

```
"SSH": {
  "SharedWithManagerCLI": false,
  "Port": 9022,
  "ServiceEnabled": true
  "HotKeySequenceDisplay": "Press ~. to exit console"
}
```