

After the upgrade a reboot will be initiated. The connection will be lost during this time frame

Parameters

Type	Name	Description	Schema
FormData	SWImage <i>required</i>	The SW upgrade file	file

Responses

HTTP Code	Description	Schema
200	OK	No Content
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Consumes

- `multipart/form-data`

Security

Type	Name
apiKey	AdminSecurity

2.4.11. Change WWN of the Library

POST /library/wwn

Description

With this command the WWN (WWide Node Name) can be set. Format is 16 hex characters [0-9 A-F]. After successful change the Library will be rebooted.

Parameters

Type	Name	Description	Schema
Body	setConfWwn <i>required</i>	the new WWN to be set	PostLibraryWwn

PostLibraryWwn

Name	Description	Schema
wwn <i>required</i>	Length : 16	string

Responses

HTTP Code	Description	Schema
200	The WWN was successfully changed.	No Content
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	ServiceSecurity

2.5. Library Test

Library Tests

2.5.1. Slot to Slot Test

POST /test/slot2slot

Description

Performs the slot to slot test n times. The number of executions has to be defined in the count parameter. The command returns immediately after starting the test. The progress has to be polled by the /test/status request

Parameters

Type	Name	Description	Schema
Body	count <i>required</i>	the number of test iterations to perform	PostTestSlotToSlot

PostTestSlotToSlot

Name	Description	Schema
count <i>required</i>	Maximum value : 100 Example : 10	integer

Responses

HTTP Code	Description	Schema
200	The test could be started successfully	No Content
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	AdminSecurity

2.5.2. Test Status

GET /test/status

Description

Returns the status of a test which is currently executed. A test run always asynchronously, a response is sent immediately after the start. This function is used to monitor the status of the test, it returns the current test cycle and the run status.

Responses

HTTP Code	Description	Schema
200	Status of current test	TestStatus
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	AdminSecurity

Example HTTP response

Response 200

```
{
  "currentCycle" : "2",
  "testCycles" : "10",
  "status" : "TRUE",
  "running" : "FALSE"
}
```

2.5.3. System (Demo) Test

POST /test/systemtest

Description

Performs the integrated system (Demo) test n times. The number of executions has to be defined in the count parameter. The command returns immediately after starting the test. The progress has to be polled by the /test/status request

Parameters

Type	Name	Description	Schema
Body	testParameter <i>required</i>	the number of test iterations to perform and seating mode	PostTestSystemTest

PostTestSystemTest

Name	Description	Schema
count <i>required</i>	Maximum value : 100 Example : 10	integer
seating <i>optional</i>		enum (TRUE, FALSE)

Responses

HTTP Code	Description	Schema
200	The test could be started successfully	No Content
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	AdminSecurity

2.6. Drive Information

Drive Information

2.6.1. Get drive information

```
GET /drive/information
```

Description

Returns the information about drives like Product Id, Model, Serial number and other useful information.

Responses

HTTP Code	Description	Schema
200	OK	< DriveInfo > array

Security

Type	Name
apiKey	UserSecurity

Example HTTP response

Response 200

```
{
  "LogicalNumber" : "1",
  "PhysicalNumber" : "10",
  "Module" : "1",
  "Partition" : "0",
  "Generation" : "8",
  "Cartridge" : "FALSE",
  "Vendor" : "IBM",
  "Product" : "ULT3580-HH8",
  "FWRevision" : "J28F",
  "SerialNumber" : "000000005B",
  "WWNodeName" : "",
  "Interface" : "FC",
  "MFGSerialNumber" : "1013000188",
  "ErrorState" : "TRUE",
  "Power" : "FALSE",
  "Presence" : "TRUE",
  "ADTMode" : "IADT"
}
```

2.7. Drive Configuration

Drive Configuration

2.7.1. Upgrade Drive FW

```
POST /drive/fwupgrade
```


Description

The command upgrades the FW of the specified drives in the library.

Parameters

Type	Name	Description	Schema
FormData	FWImage <i>required</i>	The FW upgrade file	file
FormData	drivenumber <i>required</i>	The logical drive number(s), multiple drives possible as array (comma separated values)	< integer > array(csv)

Responses

HTTP Code	Description	Schema
200	The FW upgrade was successful	No Content
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Consumes

- `multipart/form-data`

Security

Type	Name
apiKey	AdminSecurity

2.7.2. Drive Reset

```
POST /drive/reset
```

Description

The command performs a drive reset

Parameters

Type	Name	Description	Schema
Body	DriveNumber <i>required</i>	The logical number of the drive to be reset	PostDriveReset

PostDriveReset

Name	Description	Schema
DriveNumber <i>required</i>	Minimum value : 1 Maximum value : 21 Example : 3	integer

Responses

HTTP Code	Description	Schema
200	The drive was reset successfully	No Content
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	AdminSecurity

2.8. Partition Information

Partition Information

2.8.1. Get partition (Logical Library) information

```
GET /partition/information
```

Description

The command returns the list of partitions including partition details

Responses

HTTP Code	Description	Schema
200	OK	< PartitionInfo > array
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	UserSecurity

Example HTTP response

Response 200

```
[ {
  "PartitionNumber" : "1",
  "Name" : "New Partition_1",
  "SerialNumber" : "DE00000000_LL01",
  "NumSlots" : "87",
  "NumIOSlots" : "0",
  "NumDrives" : "1",
  "LunMasterDrive" : "1",
  "LunMasterDrivePhys" : "10",
  "LunMasterDriveArr" : [ "1" ],
  "LunMasterDrivePhysArr" : [ "10" ],
  "EncryptionMode" : "ISV",
  "BarcodeAlign" : "left",
  "BarcodeLength" : "8",
  "AutoClean" : "TRUE",
  "WWNode" : "50010000000000071",
  "Micw" : "FALSE"
}, {
  "PartitionNumber" : "2",
  "Name" : "New Partition_2",
  "SerialNumber" : "DE00000000_LL02",
  "NumSlots" : "90",
  "NumIOSlots" : "5",
  "NumDrives" : "1",
```

```

"LunMasterDrive" : "2",
"LunMasterDrivePhys" : "11",
"LunMasterDriveArr" : [ "2" ],
"LunMasterDrivePhysArr" : [ "11" ],
"EncryptionMode" : "ISV",
"BarcodeAlign" : "left",
"BarcodeLength" : "8",
"AutoClean" : "TRUE",
"WWNode" : "5001000000000081",
"Micw" : "FALSE"
}, {
  "PartitionNumber" : "3",
  "Name" : "New Partition_3",
  "SerialNumber" : "DE00000000_LL03",
  "NumSlots" : "85",
  "NumIOSlots" : "5",
  "NumDrives" : "1",
  "LunMasterDrive" : "3",
  "LunMasterDrivePhys" : "12",
  "LunMasterDriveArr" : [ "3" ],
  "LunMasterDrivePhysArr" : [ "12" ],
  "EncryptionMode" : "ISV",
  "BarcodeAlign" : "left",
  "BarcodeLength" : "8",
  "AutoClean" : "TRUE",
  "WWNode" : "5001000000000091",
  "Micw" : "FALSE"
} ]

```

2.8.2. Inventory of a partition

```
GET /partition/inventory
```

Description

Returns the inventory of the selected partition including slots, drives and mailslots. It also returns the corresponding metadata which belongs to the different types of the elements

Parameters

Type	Name	Description	Schema
Query	partitionNum <i>required</i>	The partition number for which the inventory should be reported	integer

Responses

HTTP Code	Description	Schema
200	OK	Inventory
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	UserSecurity

Example HTTP response

Response 200

```
{
  "Slots" : [ {
    "PhysicalNumber" : "1",
    "LogicalNumber" : "1.1",
    "Module" : "1",
    "Partition" : "1",
    "Mailslot" : "FALSE",
    "Cartridge" : "FALSE",
    "CartridgeType" : "N/A",
    "Access" : "TRUE",
    "Blocked" : "TRUE"
  }, {
    "PhysicalNumber" : "2",
    "LogicalNumber" : "1.2",
    "Module" : "1",
    "Partition" : "1",
    "Mailslot" : "FALSE",
    "Cartridge" : "FALSE",
    "CartridgeType" : "N/A",
    "Access" : "TRUE",
    "Blocked" : "FALSE"
  }, {
    "PhysicalNumber" : "3",
    "LogicalNumber" : "1.3",
    "Module" : "1",
    "Partition" : "1",
    "Mailslot" : "FALSE",
```

```

"Cartridge" : "TRUE",
"Barcode" : "TT0013L4",
"CartridgeType" : "Data",
"CartridgeSubType" : 0,
"CartridgeGeneration" : "4",
"CartridgeEncrypted" : "FALSE",
"Access" : "TRUE",
"Blocked" : "FALSE"
}, { }, {
  "PhysicalNumber" : "280",
  "LogicalNumber" : "7.40",
  "Module" : "7",
  "Partition" : "0",
  "Mailslot" : "TRUE",
  "Cartridge" : "FALSE",
  "CartridgeType" : "N/A",
  "Access" : "TRUE",
  "Blocked" : "FALSE"
} ],
"Drives" : [ {
  "PhysicalNumber" : "10",
  "LogicalNumber" : "1",
  "Partition" : "0",
  "Vendor" : "VENDOR ",
  "Product" : "Ultrium 7-SCSI ",
  "FWRevision" : "ABCD",
  "SerialNumber" : 1234567890
}, {
  "PhysicalNumber" : "11",
  "LogicalNumber" : "2",
  "Partition" : "0",
  "Vendor" : "VENDOR ",
  "Product" : "Ultrium 7-SCSI ",
  "FWRevision" : "ABCD",
  "SerialNumber" : 1234567891
}, {
  "PhysicalNumber" : "12",
  "LogicalNumber" : "3",
  "Partition" : "0",
  "Vendor" : "VENDOR ",
  "Product" : "Ultrium 7-SCSI ",
  "FWRevision" : "ABCD",
  "SerialNumber" : 1234567892
}, {
  "PhysicalNumber" : "20",
  "LogicalNumber" : "4",
  "Partition" : "0",
  "Vendor" : "",
  "Product" : "",
  "FWRevision" : "",
  "SerialNumber" : ""

```

```
} ]  
}
```

2.8.3. List of media of a partition

```
GET /partition/mediainfo
```

Description

Returns the list of media which are assigned to the partition including detailed information about them

Parameters

Type	Name	Description	Schema
Query	partitionNum <i>required</i>	The partition number for which the media information should be reported	integer

Responses

HTTP Code	Description	Schema
200	OK	< MediaInfoData > array
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Security

Type	Name
apiKey	UserSecurity

Example HTTP response

Response 200

```
[ {
  "Barcode" : "TD239ML4",
  "LocationType" : "SLOT",
  "LogicalNumber" : "3.35",
  "PhysicalNumber" : "115",
  "Cleaning" : "FALSE",
  "Partition" : "2",
  "Generation" : "4",
  "SubType" : "0",
  "Protection" : "FALSE",
  "Encryption" : "FALSE",
  "NoLoads" : "0",
  "MBRead" : "0",
  "MBReadLoad" : "0",
  "MBWritten" : "0",
  "MBWrittenLoad" : "0"
}, {
  "Barcode" : "TC084ML5",
  "LocationType" : "SLOT",
  "LogicalNumber" : "5.10",
  "PhysicalNumber" : "170",
  "Cleaning" : "FALSE",
  "Partition" : "2",
  "Generation" : "5",
  "SubType" : "0",
  "Protection" : "TRUE",
  "Encryption" : "FALSE",
  "NoLoads" : "64",
  "MBRead" : "1643",
  "MBReadLoad" : "1603",
  "MBWritten" : "943",
  "MBWrittenLoad" : "943"
} ]
```

2.9. Partition Configuration

Partition Configuration

2.9.1. Create partitions in simple mode

```
POST /partition/createsimple
```

Description

Creates the number of partitions which are requested in the parameters. This command corresponds to the Basic Wizard in RMI to create partitions. The maximum number of partitions which can be created are limited by the number of installed drives.

Parameters

Type	Name	Description	Schema
Body	partitionParameters <i>required</i>	the parameters to create the partitions	PostPartitionCreateSimple

PostPartitionCreateSimple

Name	Description	Schema
numPartitions <i>required</i>	The number of partitions to create Minimum value : 1 Maximum value : 21 Example : 1	integer
barcodeLength <i>optional</i>	Minimum value : 6 Maximum value : 16 Example : 8	integer
barcodeAlignment <i>optional</i>	Alignment of the barcode for SCSI element status. Can be "left" or "right" Default : "left"	enum (left, right)
autoClean <i>optional</i>	Auto Cleaning for this partition. 'TRUE' for enabling 'FALSE' to disable it. Note: <i>At some product variants the default value is 'TRUE'</i> Default : "FALSE"	enum (TRUE, FALSE)
sequentialMode <i>optional</i>	Enable Sequential Mode for this partition. 'TRUE' for enabling 'FALSE' to disable it. Note: <i>Only applicable for some variants</i> Default : "FALSE"	enum (TRUE, FALSE)
sequentialModeLoop <i>optional</i>	Restarting sequence when finished. Requires sequentialMode to be 'TRUE' Note: <i>Only applicable for some variants</i> Default : "FALSE"	enum (TRUE, FALSE)

Name	Description	Schema
sequentialModeAutoload <i>optional</i>	Load first cartridge automatically in sequential mode. Requires sequentialMode to be 'TRUE' Note: <i>Only applicable for some variants</i> Default : "FALSE"	enum (TRUE, FALSE)

Responses

HTTP Code	Description	Schema
200	OK	No Content
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Consumes

- `application/json`

Security

Type	Name
apiKey	AdminSecurity

2.10. MLM

MLM Section

2.10.1. Save MLM Database

GET /mlm/database

Description

Save MLM Database from the stack in zip format returned as "application/octet-stream".

Responses

HTTP Code	Description	Schema
200	OK	file (binary)
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content
500	Error occurred at operation	Error

Produces

- `application/octet-stream`

Security

Type	Name
apiKey	AdminSecurity

2.11. Service

Service Request

2.11.1. Get library logs dump

```
GET /library/logs
```

Description

Generates and returns a log file from the stack in binary format "application/octet-stream". Other responses are returned as "application/json"

Responses

HTTP Code	Description	Schema
200	Log file (zipped)	file (binary)
401	Access denied - not authenticated	No Content
403	Access forbidden - insufficient rights	No Content

HTTP Code	Description	Schema
500	Error occurred at operation	Error

Produces

- `application/octet-stream`
- `application/json`

Security

Type	Name
apiKey	AdminSecurity

Chapter 3. Definitions

3.1. BaseInfoData

The definition of the physical base module information

Name	Description	Schema
SerialNumber <i>optional</i>	The serial number for the complete library/stack which is also reported through SCSI	string
MacAddress_1 <i>optional</i>	The MAC address of the Ethernet port 1	string
MacAddress_2 <i>optional</i>	The MAC address of the Ethernet port 2	string
Vendor <i>optional</i>	The product vendor name	string
ProductID <i>optional</i>	The Product ID	string
BaseFWRevision <i>optional</i>	Firmware Revision of the base library	string
BaseFWBuildDate <i>optional</i>	Build date of base library firmware YYYY-MM-DD	string (date)
ExpansionFWRevision <i>optional</i>	Firmware Revision of the expansion modules	string
WWNodeName <i>optional</i>	World Wide Node Name of the library	string
RoboticHWRevision <i>optional</i>	Hardware Revision Level of the Robotic	string

Name	Description	Schema
RoboticFWRevision <i>optional</i>	Firmware Revision Level of the Robotic	string
RoboticSerialNumber <i>optional</i>	Internal serial number of the Robotic	string
NoOfModules <i>optional</i>	Number of connected expansion modules	integer
LibraryType <i>optional</i>	Library Type (32/40) (Product specific)	integer

3.2. BaseStatusData

Overview status data of the library

Name	Description	Schema
Information <i>optional</i>	Activity Status of the library	string
RobStatus <i>optional</i>	Activity Status of the Robotic	enum (Idle, Moving)
MoveCount <i>optional</i>	Number of moves over lifetime	integer
PowerUpCount <i>optional</i>	Number of power cycles over lifetime	integer
PowerOnTime <i>optional</i>	Power up time over lifetime	string
LibHealth <i>optional</i>	Health Status of library (Summary)	string

3.3. DriveInfo

Name	Description	Schema
LogicalNumber <i>optional</i>	The logical drive number starts always at 1 with the first inserted drive in the lowest available module	integer
PhysicalNumber <i>required</i>	The physical drive number starting at 1. Formula is (ModuleNo - 1) * (total Drive Slots in single Module) + (DriveSlotNumber in Module)	integer
Module <i>optional</i>	The Expansion Module where the drive is located in	integer
Partition <i>optional</i>	The partition (logical library) where the drive is assigned to. If no partition is assigned the value will be 0	integer
Generation <i>optional</i>	LTO Generation of Drive	integer
Cartridge <i>optional</i>	TRUE - Cartridge is loaded / FALSE - no Cartridge loaded	enum (TRUE, FALSE)
Barcode <i>optional</i>	The Barcode Label of the cartridge which is loaded in the drive	string
Vendor <i>optional</i>	The drive vendor name	string
Product <i>optional</i>	The Product ID of the drive	string
FWRevision <i>optional</i>	The FW revision of the drive	string
SerialNumber <i>optional</i>	The serial number of the drive (Spoofed)	string
WWNodeName <i>optional</i>	The World Wide Node Name of the drive	string
Interface <i>optional</i>	Interface Type - SAS / Fibre Channel	string

Name	Description	Schema
MFGSerialNumber <i>optional</i>	The Manufacturing Serial number	string
ErrorState <i>optional</i>	Is Drive in an error state	enum (TRUE, FALSE)
Power <i>optional</i>	Is drive powered	enum (TRUE, FALSE)
Presence <i>optional</i>	Is drive present	enum (TRUE, FALSE)
ADTMode <i>optional</i>	ADT Transport Mode	string

3.4. DriveInfoInventory

Name	Description	Schema
PhysicalNumber <i>required</i>	The physical drive number starting at 1. Formula is (ModuleNo - 1) * (total Drive Slots in single Module) + (DriveSlotNumber in Module)	integer
LogicalNumber <i>optional</i>	The logical drive number starts always at 1 with the first inserted drive in the lowest available module	integer
Module <i>optional</i>	The Expansion Module where the drive is located in	integer
Partition <i>optional</i>	The partition (logical library) where the drive is assigned to. If no partition is assigned the value will be 0	integer
Barcode <i>optional</i>	The Barcode Label of the cartridge which is loaded in the drive	string
Vendor <i>optional</i>	The drive vendor name	string
Product <i>optional</i>	The Product ID of the drive	string

Name	Description	Schema
FWRevision <i>optional</i>	The FW revision of the drive	string
SerialNumber <i>optional</i>	The serial number of the drive (Spoofed)	string

3.5. Error

Name	Schema
code <i>optional</i>	string
message <i>optional</i>	string

3.6. EventEntry

Structure of event entry

Name	Description	Schema
Number <i>required</i>	Number of event in list	integer
TimeStamp <i>optional</i>	Date/ Time of event	string
EventCode <i>optional</i>	Eventcode, describing the event which occurred	integer
Text <i>optional</i>	Short description of EventCode	string
State <i>optional</i>	Shows the current state for the ticket.	enum (INFO, CONFIG, SERVICE, TICKET_PENDING, TICKET_RESOLVED, SINGULAR_TICKET)
Closed <i>optional</i>	Reflects the closed state for tickets	enum (TRUE, FALSE)

3.7. IOStatus

IO Station (Mailslot) status data

Type : < [IOStatus](#) > array

IOStatus

Name	Description	Schema
ModuleNo <i>required</i>	logical module number where the IO Station is located	integer
Configured <i>optional</i>	FALSE - IO Station Slots are configured as storage slots. TRUE - IO Station Slots are configured for IO Operation	enum (TRUE, FALSE)
Unlocked <i>optional</i>	FALSE - IO Station is locked. TRUE - IO Station is unlocked	enum (TRUE, FALSE)
OpenStatus <i>optional</i>	FALSE - IO Station is closed. TRUE - IO Station is opened	enum (TRUE, FALSE)

3.8. Inventory

Name	Schema
Slots <i>optional</i>	< Slot > array
Drives <i>optional</i>	< DriveInfoInventory > array

3.9. LibraryInfo

The definition of the physical library information

Name	Description	Schema
BaseInfo <i>required</i>		BaseInfoData
ModulesInfo <i>required</i>	Module Information	< Module > array

3.10. LibraryStatus

Overview status information of the library

Name	Schema
BaseStatus <i>required</i>	BaseStatusData
ModulesStatus <i>required</i>	< ModuleStatusData > array

3.11. LicenseInfo

Licensed Feature data

Name	Description	Schema
Feature <i>required</i>	Name of licensed feature	string
LicenseKey <i>required</i>	License Key string	string

3.12. MediaInfoData

Information about a tape media.

Name	Description	Schema
Barcode <i>optional</i>	The Barcode Label of the media	string
LocationType <i>optional</i>	Type of location where the media is stored. "Drive" or "Slot"	enum (DRIVE, SLOT)
LogicalNumber <i>optional</i>	The logical number of media location	string
PhysicalNumber <i>optional</i>	The physical number of media location	string

Name	Description	Schema
Cleaning <i>optional</i>	TRUE - media is a cleaning tape / FALSE - media is not a cleaning tape	enum (FALSE, TRUE)
Partition <i>optional</i>	The partition (logical library) where the media is assigned to. If no partition is assigned the value will be 0	integer
Generation <i>optional</i>	LTO Generation of media	integer
SubType <i>optional</i>	LTO SubType of media (0 = default, 1= Type M)	integer
Protection <i>optional</i>	TRUE - media is write protected / FALSE - media is not protected	enum (FALSE, TRUE)
Encryption <i>optional</i>	TRUE - media is encrypted / FALSE - media is not encrypted / null if not identified by drive	enum (FALSE, TRUE)
NoLoads <i>optional</i>	Number of loads of media. Shows how many times the media was loaded into a tape drive	integer
MBRead <i>optional</i>	Bytes read from this media in MB	integer
MBReadLoad <i>optional</i>	Bytes read from this media in MB during last load	integer
MBWritten <i>optional</i>	Bytes written to this media in MB	integer
MBWrittenLoad <i>optional</i>	Bytes written to this media in MB during last load	integer

3.13. Module

The definition of an expansion module informational data

Name	Description	Schema
PhysicalNumber <i>required</i>	The physical number of the expansion module. the numbering starts always at 1 with the bottom module counting to 7 regardless of the physical presence of this module. The base module has always the number 4	integer
LogicalNumber <i>optional</i>	The number of the module in an existing library stack. The lowest module always starts with 1	integer
ReadyStatus <i>optional</i>	This entry returns the current status of the module. If it is correctly initialized and running it will be set to TRUE	enum (TRUE, FALSE)
SerialNumber <i>optional</i>	The internal manufacturing serial number of the module	string

3.14. ModuleStatusData

Current Health Status of a module

Name	Description	Schema
PhysicalNumber <i>required</i>	Module Number (physically)	integer
LogicalNumber <i>optional</i>	Module Number (logical)	integer
Health <i>optional</i>	Current health status of Module	string

3.15. Modules

List of expansion module data

Type : < [Module](#) > array

3.16. PartitionInfo

Partition data

Name	Description	Schema
EncryptionMode <i>optional</i>	Encryption Mode which is currently active for this partition. The following values are possible <ul style="list-style-type: none"> • OFF - No encryption policy activated • ISV - Encryption can be controlled by host application • KMIP - KMIP Key Management is activated. A KMIP Key Server is required • ESKM - ESKM Key Management is activated. A ESKM Server is required • PLK - Local Key Management over a PLK Token is active. A PLK token is required 	enum (OFF, ISV, KMIP, ESKM, PLK, IPP)
BarcodeAlign <i>optional</i>	Barcode Label Alignment (left,right)	enum (left, right)
BarcodeLength <i>optional</i>	Barcode Label Length	integer
AutoClean <i>optional</i>	AutoClean enabled	enum (TRUE, FALSE)
WWNode <i>optional</i>	WWN of Partition	string
Micw <i>optional</i>	Multi-initiator conflict warning enabled	enum (TRUE, FALSE)

3.17. Slot

The definition of cartridge slot data

Name	Description	Schema
PhysicalNumber <i>required</i>	The physical slot number starting at 1. Formula is (ModuleNo - 1) * (total Slots in single Module) + (SlotNumber in Module)	integer
LogicalNumber <i>optional</i>	The logical slot number includes the module number as prefix and starts always at 1 in a module. Format - ModuleNo.(SlotNumber in Module)	string

Name	Description	Schema
Module <i>optional</i>	The Expansion Module where the slot is located in	integer
Partition <i>optional</i>	The partition (logical library) where the slot is assigned to. If no partition is assigned the value will be 0	integer
Mailslot <i>optional</i>	If the slot is mapped as a mailslot (I/O Station) this entry is set to TRUE	enum (TRUE, FALSE)
Cartridge <i>optional</i>	If a cartridge is detected in the slot this entry is set to TRUE	enum (TRUE, FALSE)
CartridgeType <i>optional</i>	Type of cartridge Data,Cleaning,Worm etc.	enum (Unknown, Data, Cleaning, Worm, N/A)
CartridgeSub Type <i>optional</i>	SubType of a cartridge 0 = Default, 1= Type M	integer
CartridgeGeneration <i>optional</i>	Generation of the cartridge	integer
CartridgeEncrypted <i>optional</i>	Cartridge encryption state	enum (TRUE, FALSE)
Barcode <i>optional</i>	The Barcode Label of the cartridge which is sitting in the slot	string
Access <i>optional</i>	If the slot is accessible by the system the entry is set to TRUE	enum (TRUE, FALSE)
Blocked <i>optional</i>	TRUE - The slot is not reachable by the robot / FALSE - Slot is reachable by the robot Default : "FALSE"	enum (TRUE, FALSE)

3.18. TestStatus

Status of current running test

Chapter 4. Security

4.1. UserSecurity

User role API Key, only allowing status requests.

Type : apiKey

Name : Authorization

In : HEADER

4.2. AdminSecurity

Admin role API Key, allowing status, Operations and operational requests

Type : apiKey

Name : Authorization

In : HEADER

4.3. ServiceSecurity

Service role API Key, allowing special service requests including Admin level

Type : apiKey

Name : Authorization

In : HEADER