

REST JSON API

REST JSON API

Table of Contents

Glossary	1
Introduction	ii
Version	ii
1. Service API	3
ServiceLogin	3
ServiceLogout	4
Audit Event	4
Get Service Client	5
Register User	16
2. Resource API	17
Parameter Listing	17
Device	17
Get Device	17
AddDevice	20
DeleteDevice	22
SetDeviceInfo	22
Device Object Response XML	24
Port	29
Get Port	29
Get Port Tags	40
Get Port Display Config	42
Get Port Display Status	42
SetPortInfo	43
SetPortObserverInfo	44
SetPortPVRInfo	45
SetPortStreamURL	46
Set Port Display Config	47
Set Port Window Properties	47
Port Object Response XML	51
Dir	52
Get Dir	52
Get Dir Tags	56
File	57
Get File	57
Get File Tags	69
SetFileMetaData	71
Delete File	73
Prepare File For download	76
Create Media From Cutlist	79
Trim File with Cutlist	81
Clear Download Cache	82
Inject Media	85
Download Media	87
User	87
Get User	87
Get User Tags	90
Set User Password	91
Get Tags	92
Get DisplayProfile	94
3. Resource Update Notifications	97
Resource ConnectionUpdate Notification	97

Resource ConnectionStateUpdate Notification	101
Port	102
Port Update	102
Port Connection Update	103
Port Connection State Update	104
Port Display Config Update	104
Port Display Status Update	104
File	105
File Update	105
File Connection Update	109
File Connection State Update	109
4. Session API	110
Session Info	110
Get Session Config	110
Get Session	112
Session Requests	120
Parameter Listing	120
StartSession	121
StopSession	129
Session Connection Operations	129
View	129
Playback	140
Play	146
Record	149
SwitchSrc	161
ResumeRecord	162
SkipBack	163
SkipFwd	164
PauseRecord	165
Pause	166
Catchup	167
Session Participant API	168
Invite	168
Join	169
Kick	170
Bookmark Operations	171
GetBookmark	171
AddBookmark	174
DeleteBookmark	175
NewBookmark	176
UpdateBookmark	177
Authorization	178
Authorize Request	178
5. Session Notifications	180
EntityPresence Notification	180
ConnState Notification	180
SessionState Notification	183
Timeline Notification	183
A. Data Model	185
RX Display Config/Status Parameter Listing	185
RX DisplayConfig	185
RX DisplayStatus	186
Remote Operated Client Display Config/Status Parameter Listing	186
Remote Operated Client DisplayConfig	188

Remote Operated Client DisplayStatus	191
B. REST API - Basic Tutorial	193
REST Tester	193
Service Login	193
GetPort - Sources	194
GetPort - Destination	194
View	194
StopSession	194
Record	195
GetFile - Recordings	195
Playback	195
View live source using mediaURL	196
Playback recorded file using mediaURL	197
Download Media	198
Create Media using Cutpoints	199
API	200
Release Notes	200
Summary of changes from 4.0 to 4.1	200
Summary of changes from 4.1 to 4.2	201

List of Tables

1. Sample Request Headers for every API call	ii
1.1. Service Login Request	3
1.2. Service Logout Request	4
1.3. Audit Event Request	5
1.4. Get Service Client Request	5
1.5. Register User Request	16
2.1. Get Device Request	17
2.2. Add Device Request	21
2.3. Delete Device Request	22
2.4. Set Device Info Request	22
2.5. Get Port List Request	29
2.6. Get Port Tags Request	40
2.7. GetPortDisplayConfig Request	42
2.8. GetPortDisplayStatus Request	43
2.9. Set Port Info Request	43
2.10. Set Port Observer Info Request	44
2.11. Set Port PVR Info Request	45
2.12. Set Port StreamURL Request	46
2.13. Set Port Display Config	47
2.14. Set Port Window Properties	48
2.15. Get Dir Request	52
2.16. Get Dir Tags Request	56
2.17. Get File Request	57
2.18. Get File Tags Request	70
2.19. Set File MetaData Request by NID or title	71
2.20. DeleteFile Request	74
2.21. Prepare File For download Request	76
2.22. Create Media From Cutlist	79
2.23. Trim File with Cutlist	81
2.24. Clear Download Cache	83
2.25. Get User Request	87
2.26. Get User Tags Request	90
2.27. Set User Password Request	91
2.28. Get Tags Request	92
2.29. Get DisplayProfile Request	94
4.1. Get Session Config Request	110
4.2. Get Session Request	112
4.3. Basic Start Recording Session Request	122
4.4. StopSession Request	129
4.5. Basic View Request by Title	130
4.6. Basic Playback Request by Title	141
4.7. Play Request	147
4.8. Basic Record Request	150
4.9. SwitchSrc Request	161
4.10. ResumeRecord Request	162
4.11. SkipBack Request	163
4.12. SkipFwd Request	164
4.13. PauseRecord Request	165
4.14. Pause Request	166
4.15. Catchup Request	167
4.16. Simple Invite by userId or jid	168

4.17. Join without invite Request	169
4.18. Kick	170
4.19. GetBookmark Request	171
4.20. AddBookmark Request	174
4.21. DeleteBookmark Request	175
4.22. NewBookmark Request	176
4.23. UpdateBookmark Request	177
4.24. Authorize Request	178
B.1.	200

List of Examples

1.1. Service Login Request	3
1.2. Service Login Response	4
1.3. Service Logout Request	4
1.4. Service Logout Response	4
1.5. Audit Event Request	5
1.6. Audit Event Response	5
1.7. Get Service Client Request	5
1.8. Get Service Client Response - BASIC Format	6
1.9. Get Service Client Response - BASIC Format (Continued)	7
1.10. Get Service Client Response - BASIC Format (Continued)	8
1.11. Get Service Client Response - EXTENDED Format	9
1.12. Get Service Client Response - EXTENDED Format (Continued)	10
1.13. Get Service Client Response - EXTENDED Format (Continued)	11
1.14. Get Service Client Response - EXTENDED Format (Continued)	12
1.15. Get Service Client Response - EXTENDED Format (Continued)	13
1.16. Get Service Client Response - EXTENDED Format (Continued)	14
1.17. Get Service Client Response - EXTENDED Format (Continued)	15
1.18. Register User Request	16
2.1. Get Device Request	17
2.2. Get Device Response - BASIC Format	19
2.3. Get Device Response - Basic Response (Continued)	20
2.4. Add Device Request	21
2.5. Add Device Response	21
2.6. Delete Device Request	22
2.7. Delete Device Response	22
2.8. Set Device Info Request	22
2.9. Set Device Info Response	23
2.10. Device Object Response XML	24
2.11. Device Object Response XML (Continued)	25
2.12. Device Object Response XML (Continued)	26
2.13. Device Object Response XML (Continued)	27
2.14. Device Object Response XML (Continued)	28
2.15. Device Object Response XML (Continued)	29
2.16. Get Port List Request	29
2.17. NOTE about fetching URLs from the System	30
2.18. Get Port Response - BASIC Format	31
2.19. Get Port Response - EXTENDED Format	32
2.20. Get Port Response - EXTENDED Format continued	33
2.21. Get Port Response - EXTENDED Format continued	34
2.22. Get Port Response - BASICCONNECTIONINFO Format	35
2.23. Get Port Response - BASICCONNECTIONINFO Format (Continued)	36
2.24. Get Port Response - EXTENDEDCONNECTIONINFO Format	37
2.25. Get Port Response - EXTENDEDCONNECTIONINFO Format (Continued)	38
2.26. Get Port Response - BASICURL Format	39
2.27. Get Port Response - EXTENDEDURL Format	40
2.28. Get Port Tags Request	40
2.29. Get Port Tags Response - NameList Format	41
2.30. Get Port Tags Response - NameValueList Format	41
2.31. Get Port Tags Response - NameValueTree Format	42
2.32. Get Port Display Configuration Request	42
2.33. Get Port Display Config Response	42

2.34. Get Port Display Status Request	43
2.35. Get Port Display Status Response	43
2.36. Set Port Info Request	43
2.37. Set Port Info Response	44
2.38. Set Port Observer Info Request	44
2.39. Set Port Observer Info Response	45
2.40. Set Port PVR Info Request	45
2.41. Set Port PVR Info Response	46
2.42. Set Port StreamURL Request	46
2.43. Set Port StreamURL Response	47
2.44. Set Port Display Configuration Request	47
2.45. Set Port Display Configuration Response	47
2.46. Set Port Window Properties Request	48
2.47. Set Port Window Properties Response	50
2.48. Port Object Response XML	51
2.49. Get Dir Request	52
2.50. Get Dir Response - BASIC Format	53
2.51. Get Dir Response - SNAPSHOTURL Format (Deprecated)	54
2.52. Get Dir Response - UPLOADURL Format	55
2.53. Get Dir Tags Request	56
2.54. Get Dir Tags Response - NameList Format	56
2.55. Get Dir Tags Response - NameValueList Format	57
2.56. Get Dir Tags Response - NameValueTree Format	57
2.57. Get File Request	57
2.58. Sample sortByTagName="Location"	60
2.59. Sample sortByMetaData="TestAPI"	60
2.60. Get File Response - EXTENDED Format	61
2.61. Get File Response - EXTENDED Format (Continued)	62
2.62. Get File Response - EXTENDED Format (Continued)	63
2.63. Get File Response - BASIC Format	64
2.64. Get File Response - BASICURL Format	65
2.65. Get File Response - EXTENDEDURL Format	66
2.66. Get File Response - EXTENDEDURL Format (Continued)	67
2.67. Get File Response - BASICCONNECTIONINFO Format	68
2.68. Get File Response - BASICCONNECTIONINFO Format (Continued)	69
2.69. Get File Tags Request	70
2.70. Get File Tags Response - NameList Format	70
2.71. Get File Tags Response - NameValueList Format	71
2.72. Get File Tags Response - NameValueTree Format	71
2.73. Set File MetaData Request by NID	71
2.74. Set File MetaData Response	73
2.75. DeleteFile Request	74
2.76. DeleteFile Response	75
2.77. DeleteFile Response.. continued	76
2.78. Prepare File For download Request	76
2.79. Prepare File For Download Response	78
2.80. Create Media From Cutlist	79
2.81. Create Media From Cutlist Response	80
2.82. Trim File with Cutlist	81
2.83. Trim File with Cutlist Response	82
2.84. Clear Download Cache	83
2.85. Clear Download Cache Response	84
2.86. Inject Media	86
2.87. Download Media	87

2.88. Get User Request	87
2.89. Get User Response - BASIC Format	88
2.90. Get User Response - BASIC Format (Continued)	89
2.91. Get User Response - EXTENDED Format	89
2.92. Get User Tags Request	90
2.93. Get User Tags Response - NameList Format	90
2.94. Get User Tags Response - NameValueList Format	91
2.95. Get User Tags Response - NameValueTree Format	91
2.96. Set User Password Request	91
2.97. Set User Password Response	92
2.98. Get Tags Request	92
2.99. Get Tags Response - NameList Format	93
2.100. Get Tags Response - NameValueList Format	93
2.101. Get Tags Response - NameValueTree Format	94
2.102. Get DisplayProfile Request	94
2.103. Get DisplayProfile Response - BASIC Format	95
2.104. Get DisplayProfile Response - EXTENDED Format	96
3.1. AuthConectionInfo element in AuthorizationRequired Notification	98
3.2. PortConectionInfo element in PortConnectionUpdate Notification - BASICCONNECTIONINFO format	99
3.3. PortConectionInfo element in PortConnectionUpdate Notification - EXTENDEDCONNECTIONINFO format	100
3.4. FileConnectionInfo element in FileConnectionUpdate Notification - BASICCONNECTIONINFO format	101
3.5. ConectionState element in ConnectionStateUpdate Notification	102
3.6. Port Update Notification BASIC	103
3.7. Port Update Notification EXTENDED	103
3.8. Port Connection Update Notification	104
3.9. Port Connection State Update Notification	104
3.10. Port DisplayConfigUpdate Notification	104
3.11. Port DisplayStatusUpdate Notification	105
3.12. File Update Notification BASIC	106
3.13. File Update Notification EXTENDED	107
3.14. File Update Notification EXTENDEDURL	108
3.15. File Connection Update Notification	109
3.16. File Connection State Update Notification	109
4.1. Get Session Config Request	110
4.2. Get Session Config Request	111
4.3. Get Session Config Response - BASIC Format	111
4.4. Get Session Config Response - EXTENDED Format	112
4.5. Get Session Request	112
4.6. Get Session Request	113
4.7. Get Session Response - BASIC Format[Not Implemented]	114
4.8. Get Session Response - BASIC Format	115
4.9. Get Session Response - EXTENDED Format	116
4.10. Get Session Response - EXTENDED Format (Continued)	117
4.11. Get Session Response - EXTENDED Format (Continued)	118
4.12. Get Session Response - EXTENDED Format (Continued)	119
4.13. Get Session Response - EXTENDED Format (Continued)	120
4.14. Basic Start Recording Session Request	122
4.15. Basic Start Recording Session Response (Continued)	124
4.16. Basic Start Recording Session Response (Continued)	125
4.17. Basic Start Recording Session Response (Continued)	126
4.18. Basic Start Playback Session Request	127

4.19. Basic Start PlayBack Session Response (Continued)	128
4.20. Basic Start PlayBack Session Response(Continued)	129
4.21. StopSession Request	129
4.22. Basic View Request	130
4.23. Basic View Request by Title	131
4.24. Basic View Request by NID	131
4.25. View Request with Profile Title	132
4.26. View Request with Profile NID	132
4.27. View Request with PVR	133
4.28. View Request to add a connection from an existing Src to a new Dst	134
4.29. View Request to add a connection from an existing Src to a new Dst with profileid	135
4.30. View Request to add a connection from an existing Src to existing Dst	136
4.31. View Request Response	137
4.32. View Response (Continued)	138
4.33. View Response (Continued)	139
4.34. View Response (Continued)	140
4.35. Basic Playback Request by Title	141
4.36. Basic Playback Request by NID	142
4.37. Basic Playback Request with file name/date spec	143
4.38. Advanced Playback Request including grpFileId and StorePort spec	143
4.39. Basic Playback Request by group file NID	144
4.40. Basic Playback Request by srcEntityId	145
4.41. Basic Playback Request by srcEntityId and dstEntityId	146
4.42. Playback Request Response	146
4.43. Play Request	147
4.44. Play Response	148
4.45. Play Request w/ nowOffset	148
4.46. Play Request w/ startOffset	148
4.47. Play Request w/ liveOffset	149
4.48. Play Request w/ mediaOffset	149
4.49. Play Request w/ mediaTime	149
4.50. Play Request w/ Bookmark	149
4.51. Basic Record Request	150
4.52. Basic Record Request with file name/dir spec	152
4.53. Record from an existing Src to file	153
4.54. Append to Recording Based on Title	153
4.55. Append to Recording Based on MetaData	154
4.56. Record Request with Profile NID	155
4.57. Record Request with Profile Title	156
4.58. Record from an existing Src with profile id	157
4.59. Advanced Record Request including grpFileId and StorePort spec	158
4.60. Record Request Response	159
4.61. Record Request Response (Continued)	160
4.62. Record Request Response (Continued)	161
4.63. SwitchSrc Request	161
4.64. SwitchSrc Response	162
4.65. ResumeRecord Request	162
4.66. ResumeRecord Response	163
4.67. SkipBack Request	163
4.68. SkipBack Response	164
4.69. SkipFwd Request	164
4.70. SkipFwd Response	165
4.71. PauseRecord Request	165
4.72. PauseRecord Response	166

4.73. Pause Request	166
4.74. Pause Response	167
4.75. Catchup Request	167
4.76. Catchup Response	168
4.77. Simple Invite by userId or jid	168
4.78. Invite by userId(who) and/or deviceId(from where)	169
4.79. Join without invite Request	169
4.80. Join without invite Response	170
4.81. Join without invite Response - MUC Room Invite Message	170
4.82. Join with invite	170
4.83. Kick	170
4.84. GetBookmark Request	171
4.85. GetBookmark Request	172
4.86. GetBookmark Response - BASIC Format	173
4.87. GetBookmark Response - EXTENDED Format	174
4.88. AddBookmark Request	174
4.89. AddBookmark Response	175
4.90. DeleteBookmark Request	175
4.91. DeleteBookmark Response	176
4.92. NewBookmark Request	176
4.93. NewBookmark Response	177
4.94. UpdateBookmark Request	177
4.95. UpdateBookmark Response	178
4.96. Authorize Request	178
4.97. Authorize Response	179
5.1. EntityPresence Notification	180
5.2. ConnState Notification: CONNECTED	181
5.3. ConnState Notification: CONNECTED(Continued)	182
5.4. ConnState Notification: Error	183
5.5. SessionState Notification	183
5.6. Timeline Notification	184

Glossary

REST Message

The REST message module uses HTTP Headers and functions to retrieve, create, update, or delete data on a web services server based on REST architecture

Introduction

Version

Document Version = 1.0
Service Version = 2.32.3
Software Version = 4.2.x

This document contains the description of the REST JSON calls

Headers Example

Table 1. Sample Request Headers for every API call

Authorization	Basic c3JjcmVsYXkxQHNyY3BjMTpzcmluZWxheTE=
Accept Header(No spaces or newlines)	application/vnd.barco.nimbus.Device+json
Content-Type	application/json
Resource Header	X-Barco-resource web-client_barco

- Every REST API call in the document uses this as the Base URL : `http://{sc_ip_addr}/controller/{ResourcePath}?{QueryParameters}`
- Authorization is basic http authorization using base64 encoded username:password
- X-Barco-resource is a custom header which accepts the resource name in the system. The resource name is of the format: 'resource_username' as shown in sample above.
- Accept Header specifies the format in which you expect a response. Some calls in the document use custom Accept Headers as shown in sample above
- Content-Type Header specifies the format in which you POST content to the system as shown in sample above
- Each call in the document specifies the ResourcePath and QueryParameters corresponding to that call.

Chapter 1. Service API

API Calls used in Service

ServiceLogin

- During this process the client's contact Info and state are updated in the Server DB
- The client version number can be obtained from the Version section of the API document corresponding to the client implementation
- ClientTemplateTitle is required to be specified. The server will create a client using the data available for the clientTemplate. Same resource should be used for the custom header X-Barco-resource
- Login API only supports a web client resource added in the system to used for clientTemplateTitle. e.g. web-client
- clientPrivateKey is an optional value in case the resource in the system has a password

Example 1.1. Service Login Request

Table 1.1. Service Login Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Service/Login
Authorization	Basic (Required)

```
{ "ServiceLogin": {  
  "clientPrivateKey": "",  
  "clientTemplateTitle": "",  
  "@clientVersion": ""  
}}
```

For E.g.

```
{ "ServiceLogin": {  
  "clientTemplateTitle": "webclient",  
  "@clientVersion": "2.31.6"  
}}
```

NOTE: If all is well, the response will contain a Long Polling URL in the response headers. Client is required to keep polling the server with an HTTP GET on this URL 'X-Barco-notification-channel'. This request requires the same set of headers as any other request. Please refer to the Introduction section above

for set of required Headers for every request. The server can send asynchronous notifications to the client as response to this request. Every response comes back with a new URL in 'X-Barco-notification-channel' response header. There may or may not be notifications in the notificationlist returned in response for the long polling URL.

Example 1.2. Service Login Response

```
{ "ServiceLogin": {  
  "clientPrivateData": " ",  
  "serverDate": " "  
}}
```

ServiceLogout

This will make the user presence off-line and remove the user from the system's local cache. This will cause the Long-Polling channel associated with the user to be terminated. Clients that are long-polling using the URL in X-Barco-notification-channel, will see the poll requests failing and should handle them appropriately.

Example 1.3. Service Logout Request

Table 1.2. Service Logout Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Service/Logout
Authorization	Basic (Required)

Example 1.4. Service Logout Response

```
username: User removed successfully!
```

Audit Event

It is an event to log user actions. Each event is saved as a single line (newline characters are not supported in the request) in a file on the management server. This request is an input to audit report.

This is a list of parameters for an audit event request.

- *type* [Required]

- *name* [Required]
- *level*="error/warning/info/debug/trace" [Optional]
- *target* [Optional]
- *status* [Optional]
- *text* [Required]

Example 1.5. Audit Event Request

Table 1.3. Audit Event Request

Request	HTTP POST
Accept(No spaces or newlines)	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Service/AuditEvent
Authorization	Basic (Required)

```
{ "AuditEvent": {  
  "@name": "",  
  "Description": { "@text": "" },  
  "@target": "",  
  "@type": "",  
  "@status": "",  
  "@level": ""  
}}
```

Example 1.6. Audit Event Response

```
{ "AuditEvent": "" }
```

Get Service Client

This will return the logged in client information

Example 1.7. Get Service Client Request

Table 1.4. Get Service Client Request

Request	HTTP GET
---------	----------

Accept	"application/json"
Status	200, 401, 500
Query Parameters	whereContactJID, selectFormat
Resource Path	/Service/ServiceClient
Authorization	Basic (Required)

- *whereContactJID*="" [Optional]

Example 1.8. Get Service Client Response - BASIC Format

```
{ "GetServiceClient": {
  "User": {
    "Groups": "",
    "@NID": "",
    "@externalAuthentication": "",
    "@userName": "",
    "UserPrivateData": ""
  },
  "Device": { "@NID": "" },
  "Asset": {
    "Groups": "",
    "@NID": "",
    "FeatureList": { "Feature": { "@Name": "" } },
    "AssetPrivateKey": {
      "KeyData": "",
      "KeySchema": ""
    },
    "@type": "",
    "@presence": "",
    "@state": "",
    "@title": "",
    "AssetPrivateData": ""
  },
  "PortList": { "Port": {
    "Groups": "",
    "@NID": "",
    "@streamType": "",
    "FeatureList": { "Feature": { "@name": "" } },
    "@portType": "",
    "@presence": "",
    "@state": "",
    "@title": ""
  } }
}
```

Example 1.9. Get Service Client Response - BASIC Format (Continued)

```
"Permissions": {
  "User": {
    "@add": " ",
    "@modify": " ",
    "@delete": " ",
    "@read": " "
  },
  "Dir": {
    "@record": " ",
    "@playback": " ",
    "@import": " ",
    "@update": " ",
    "@delete": " ",
    "@read": " "
  },
  "Port": {
    "@add": " ",
    "@modify": " ",
    "@delete": " ",
    "@streamto": " ",
    "@viewfrom": " ",
    "@read": " "
  },
  "Device": {
    "@add": " ",
    "@modify": " ",
    "@delete": " ",
    "@read": " "
  },
  "Policy": {
    "@add": " ",
    "@modify": " ",
    "@delete": " ",
    "@read": " "
  },
  "File": {
    "@playback": " ",
    "@export": " ",
    "@modify": " ",
    "@delete": " ",
    "@read": " "
  },
}
```

Example 1.10. Get Service Client Response - BASIC Format (Continued)

```
"Profile": {
  "@add": " ",
  "@modify": " ",
  "@delete": " ",
  "@read": " "
},
"SessionConfig": {
  "@add": " ",
  "@modify": " ",
  "@delete": " ",
  "@read": " "
},
"System": {
  "@backup": " ",
  "@upgrade": " ",
  "@sessionmanagement": " ",
  "@report": " ",
  "@restore": " "
}
}
}}
```

Example 1.11. Get Service Client Response - EXTENDED Format

```
{ "GetServiceClient": {
  "User": {
    "Groups": "",
    "@NID": "",
    "@userName": "",
    "UserPrivateData": ""
  },
  "Device": {
    "Storage": {
      "@hardDriveEnabled": "",
      "HardDiskList": { "HardDisk": {
        "@mountPoint": "",
        "@name": ""
      } },
      "@NASEnabled": "",
      "NASList": { "NAS": {
        "@name": "",
        "@server": "",
        "@enabled": "",
        "@readOnly": "true/false"
      } }
    },
    "@NID": "",
    "Services": {
      "XMPPServer": {
        "@dailyBackup": "Yes/No",
        "@enableExport": "",
        "@enabled": "true/false",
        "@backupTime": "",
        "@recoveryMode": "",
        "@port": ""
      }
    }
  },
  "@NID": ""
}
```

Example 1.12. Get Service Client Response - EXTENDED Format (Continued)

```
"SSHServer": {"@enabled": "true/false"},
"DHCPServer": {
  "@enabled": "true/false",
  "@endIP": "",
  "@publishedDNSServer": "",
  "@startIP": ""
},
"XMPPAgent": {
  "@xmppServer": "",
  "DestinationAgent": {"@enalbed": "true/false"},
  "AdminAgent": {"@enalbed": "true/false"},
  "@enabled": "true/false",
  "@userName": "",
  "PortCtrlAgent": {"@enalbed": "true/false"},
  "SourceAgent": {"@enalbed": "true/false"},
  "@domain": "",
  "XAuthAgent": {"@enalbed": "true/false"},
  "ConfigAgent": {"@enalbed": "true/false"}
},
"TFTPServer": {"@enabled": "true/false"},
"NTPServer": {
  "@enabled": "true/false",
  "@serverIP": ""
},
"TelnetServer": {"@enabled": "true/false"},
"HTTPServer": {"@enabled": "true/false"},
"StreamingServer": {
  "RTPRXPorts": {
    "@to": "",
    "@from": ""
  },
  "@enabled": "true/false",
  "V2DRxPorts": {
    "@to": "",
    "@from": ""
  },
  "TXcntrlPorts": {
    "@to": "",
    "@from": ""
  }
},
```

Example 1.13. Get Service Client Response - EXTENDED Format (Continued)

```
        "MulticastIP": {
            "@to": "",
            "@from": ""
        },
        "MPEGTSRXPorts": {
            "@to": "",
            "@from": ""
        },
        "UDPRXPorts": {
            "@to": "",
            "@from": ""
        }
    },
    "Network": {
        "RouteTable": {"Route": {
            "@netmask": "",
            "@destination": "",
            "@gateway": "",
            "@listIndex": ""
        }},
        "@DNSServerIP": "",
        "@DNSName": "",
        "EthernetPortList": {"EthernetPort": {
            "@netmask": "",
            "@enabled": "true/false",
            "@useDHCP": "true/false",
            "@portID": "",
            "@isPrimary": "",
            "@multicast": "true/false",
            "@listIndex": "",
            "@ipAddress": ""
        }},
        "NATTable": {"NAT": {
            "@name": "",
            "@ip": "",
            "@listIndex": ""
        }},
        "@isPrimaryEnabled": "",
        "@hostName": ""
    },
    "System": {"NTPClient": {
        "@enabled": "true/false",
        "@serverIP": ""
    }},
    },
}
```

Example 1.14. Get Service Client Response - EXTENDED Format (Continued)

```
"AVCodec": {"Port": {"
  "Displayconfig": [
    {
      "@type": "IPVSCClientDisplayConfig",
      "Display": {
        "Panel": {
          "@content": "",
          "Geometry": {
            "@transparency": "",
            "@hres": "100%",
            "@border": "",
            "@zIndex": "",
            "@vpos": 0,
            "@aspect": "",
            "@vres": "100%",
            "@hpos": 0
          },
          "@id": "",
          "@enable": ""
        },
        "@hres": "AUTO",
        "@vres": "AUTO",
        "@configurable": ""
      },
      "Canvas": {
        "Layout": {
          "@nRow": "",
          "@border": "hide",
          "@header": "hide",
          "@type": "",
          "@hpad": "",
          "@borderStyle": "[8px]",
          "@nStrip": "",
          "@colWidth": "",
          "@active": true,
          "@headerStyle": "basic",
          "@stripPos": "",
          "@rowHeight": "",
          "@nCol": "",
          "@id": "LayoutId1",
          "@vpad": ""
        },

```


Example 1.15. Get Service Client Response - EXTENDED Format (Continued)

```
    "@hres": 1280,
    "@vres": 1024,
    "Background": {
      "@content": "",
      "ImageContent": {
        "@imageUrl": "",
        "@color": ""
      },
      "SessionContent": {
        "@enablePlayerPanel": "",
        "@enableCtrlPanel": "",
        "@enableBookmarkPanel": "",
        "@sessionId": "",
        "@enableConnPanel": "",
        "@enableHeaderPanel": ""
      },
      "@zIndex": "",
      "@enable": ""
    },
    "@id": "",

    "Window": {
      "@content": "",
      "ThumbnailContent": {
        "@imageUrl": "",
        "@MediaStreamIOPortElementID": "",
        "@MediaStreamIOPortTypeTitle": "",
        "@refreshInterval": ""
      },
      "ConnectionContent": {
        "@hres": "",
        "@srcKbmCtrl": "",
        "@MediaStreamIOPortElementID": "",
        "@vres": "",
        "@MediaStreamIOPortTypeTitle": "",
        "@sessionId": "",
        "@connId": ""
      },
    },
```

Example 1.16. Get Service Client Response - EXTENDED Format (Continued)

```
        "@id": "Win1",
        "Properties": {
            "@scrollHPos": 0,
            "@scale": "fitWindow",
            "@border": "",
            "@hres": "100%",
            "@maximize": false,
            "@zIndex": 100,
            "@aspect": "",
            "@scrollVPos": 0,
            "@transparency": 0,
            "@remoteKbmCtrl": false,
            "@vpos": 0,
            "@layoutId": "LayoutId1",
            "@vres": "100%",
            "@layoutPos": "",
            "@visibility": true,
            "@hpos": 0,
            "@fps": 15,
            "@mute": false,
            "@volume": "50%"
        }
    },
    {
        "Transparency": 0,
        "VerticalOffset": 0,
        "HorizontalResolution": 0,
        "@type": "XP200DisplayConfig",
        "HorizontalLocation": "Left",
        "Aspect": "Auto",
        "HorizontalOffset": 0,
        "VerticalScale": 100,
        "HorizontalPosition": "Location",
        "HorizontalTransformation": "Scale/Source",
        "KeepAspect": true,
        "Override": "None",
```

Example 1.17. Get Service Client Response - EXTENDED Format (Continued)

```
        "OverlayType": "Manual",
        "VerticalTransformation": "Scale/Source",
        "HorizontalScale": 100,
        "VerticalLocation": "Top",
        "VerticalPosition": "Location",
        "DisplayType": "Remote",
        "VerticalResolution": 0
    }
},
"@id": ""
}}
},

"Asset": {
    "Groups": "",
    "@NID": "",
    "FeatureList": {"Feature": {"@Name": ""}},
    "AssetPrivateKey": {
        "KeyData": "",
        "KeySchema": ""
    },
    "@type": "",
    "@presence": "",
    "@state": "",
    "@title": "",
    "AssetPrivateData": ""
},
"PortList": {"Port": {
    "Groups": "",
    "@NID": "",
    "ObserverInfoList": {"Observer": {
        "ResourceNID": "",
        "UserNID": "",
        "@xAuth": "",
        "@monitor": ""
    }},
    "@streamType": "",
    "FeatureList": {"Feature": {"@name": ""}},
    "@portType": "",
    "@presence": "",
    "@state": "",
    "@title": "",
    "StreamURL": ""
}}
}}
```

Register User

Register user in database. Self Registration on the MS should be enabled to use this API. Use the bootstrap wizard available at http://{ms_ip_address} to enable self registration

This is a list of parameters for RegisterUser request.

- *name* [Required]
- *password* [Required]

Example 1.18. Register User Request

Table 1.5. Register User Request

Request	HTTP POST
Accept	"application/json"
Status	200, 500, 204, 403
Query Parameters	None
Resource Path	/Service/RegisterUser OR / RegisterUser(deprecated)
Authorization	Basic (Required)

```
{
  "User": {
    "@name": "",
    "@password": ""
  }
}
```

Chapter 2. Resource API

API Calls used in Resource

Parameter Listing

This is the list of common parameters in Get Requests

- *startFrom* [Optional]
- *countToFetch* [Optional]
If the countToFetch is -1 , then it should get entire list of objects .
- *sortBy*= " " [Optional]
- *sortOrder*= "ascending/descending" [Optional]
- *selectFormat*= "BASIC/EXTENDED" [Required]
- *whereNID*= 'nid1,nid2,nid3' [Optional]
- *whereTagExpr*= "Name%3D'Location' and Value%3D'Sunnyvale' " [Optional]
- *whereTextSubStr*= " " [Optional]
- *subscriptionScope*=ALLNODES [Optional]
- *subscriptionID* [Optional]

Device

API calls used for Devices

Get Device

Lists the devices present in the database.

Example 2.1. Get Device Request

Table 2.1. Get Device Request

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Device+json"
Status	200, 401, 500
Query Parameters	sortBy, sortOrder, selectFormat, whereNID, whereTitle, whereType, whereTextSubStr, whereTagExpr
Resource Path	/Resource/Device

Authorization	Basic (Required)
---------------	------------------

- *sortBy*="title/type" [Optional]
- *sortOrder*="ascending/descending" [Optional]
- *whereNID* [Optional]
- *whereTitle* [Optional]
- *whereType*="IPVSClient-Device/Proxy-Device/WebClient-Device/DMS-25-Device/DMS-100-Device/DMS-120-Device/DMS-200-Device/ DMS-1000-Device/DMS-1200-Device/DMS-1500-Device/DMS-1600-Device/DMS-1700-Device/ MPEGTS-Rx-Device/MPEGTS-Tx-Device/MS-25-Device/MS-Device/RTP-Tx-Device/RTP-Rx-Device/ UDP-Tx-Device/UDP-Rx-Device/V2D-Tx-Device/V2D-Rx-Device/V2D-XP100-Device/V2D-XP200-Device/ V2D-XP200S-Device/V2D-XP220-Device/V2D-XP220S-Device/V2D-XPi-Device" [Optional (' separates the various options. Supports comma separated options)]
- *whereTextSubStr* [Optional]
- *whereTagExpr*="Name%3D'Location' and Value%3D'Sunnyvale' " [Optional]
- *subscriptionID* [Optional]

Example 2.2. Get Device Response - BASIC Format

```
{ "Resource": {
  "Device": {
    "@NID": " ",
    "@type": " ",
    "@unManaged": "true/false",
    "@presence": " ",
    "@unManned": "true/false",
    "@password": " ",
    "DeviceList": { "Device": {
      "@NID": " ",
      "@type": "WebClient-Device",
      "@unManaged": "true/false",
      "@presence": " ",
      "@unManned": "true/false",
      "@password": " ",
      "Groups": { "Tag": {
        "@name": " ",
        "@value": " "
      } },
    } },
    "@msStatus": "Active/Primary/PrimaryStandby/Secondary/SecondaryStandby",
    "Status": {
      "@systemElements": " ",
      "@serviceElements": " ",
      "@device": " ",
      "@hardDisk": " ",
      "@networkElements": " ",
      "@NAS": " "
    },
    "@softwareVersion": " ",
    "@ownerUserJID": " ",
```

Example 2.3. Get Device Response - Basic Response (Continued)

```
"PortList": { "Port": {
  "@NID": "",
  "@portType": "",
  "@deviceElementID": ""
}},
"TemplateNID": "",
"DevicePrivateData": "",
"@title": "",
"@state": "",
"@description": ""
}},
"Groups": { "Tag": {
  "@name": "",
  "@value": ""
}},
"@msStatus": "Active/Primary/PrimaryStandby/Secondary/SecondaryStandby",
"Status": {
  "@systemElements": "",
  "@serviceElements": "",
  "@device": "",
  "@hardDisk": "",
  "@networkElements": "",
  "@NAS": ""
},
"@softwareVersion": "",
"@ownerUserJID": "",
"PortList": { "Port": {
  "@NID": "",
  "@portType": "",
  "@deviceElementID": ""
}},
"TemplateNID": "",
"DevicePrivateData": "",
"@title": "",
"@state": "",
"@description": ""
},
"@total": 0,
"@from": 0
}}
```

AddDevice

Add device in database

This is a list of parameters for add device request.

- *type* [Required]

- *title* [Required [Does not allow XML restricted special characters]]
- *description* [optional]
- *password* [Required]
- *unManned* [Required]
- *unManaged* [Required]
- *applyTagsToPorts* [Required]
- *Groups* [optional]
- *DevicePrivateData* [optional]

Example 2.4. Add Device Request

Table 2.2. Add Device Request

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Device+json"
Status	201, 401, 500
Query Parameters	None
Resource Path	/Resource/Device
Authorization	Basic (Required)

```
{ "Device": {  
  "Groups": { "Tag": {  
    "@Name": " ",  
    "@Value": " "  
  }},  
  "@type": " ",  
  "@unManaged": "true/false",  
  "@unManned": "true/false",  
  "DevicePrivateData": " ",  
  "@title": " ",  
  "@applyTagsToPorts": "true/false",  
  "@description": " ",  
  "@password": " "  
}}
```

Example 2.5. Add Device Response

```
{ "Resource":  
  <!-- "Check Device Object Response XML" -->  
}
```

DeleteDevice

Delete device from database

Example 2.6. Delete Device Request

Table 2.3. Delete Device Request

Request	HTTP DELETE
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Device+json"
Status	200, 401, 500
Query Parameters	NID or title
Resource Path	/Resource/Device
Authorization	Basic (Required)

Example 2.7. Delete Device Response

```
{ "Resource": :  
  <!-- "Check Device Object Response XML" -->  
}
```

SetDeviceInfo

Update device description in database

Example 2.8. Set Device Info Request

Table 2.4. Set Device Info Request

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Device-Info+json"
Status	200, 401, 500
Query Parameters	NID or title
Resource Path	/Resource/Device

Authorization	Basic (Required)
---------------	------------------

```
{ "DeviceInfo": {  
  "Groups": { "Tag": {  
    "@Name": "",  
    "@Value": ""  
  }},  
  "DevicePrivateData": "",  
  "@applyTagsToPorts": "true/false",  
  "@description": "",  
  "@password": ""  
}}
```

Example 2.9. Set Device Info Response

```
{ "Resource": { "DeviceInfo":  
  "_comment": "Check Device Object Response XML"  
}}
```

Device Object Response XML

Example 2.10. Device Object Response XML

```
{ "Device": {
  "@NID": "",
  "@type": "",
  "@unManaged": "true/false",
  "@presence": "",
  "@unManned": "true/false",
  "DeviceConfig": {
    "Storage": {
      "@hardDriveEnabled": "",
      "HardDiskList": { "HardDisk": {
        "@mountPoint": "",
        "@name": ""
      }},
      "@NASEnabled": "",
      "NASList": { "NAS": {
        "@name": "",
        "@server": "",
        "@enabled": "",
        "@readOnly": "true/false"
      }}
    }
  },
  "@NID": "",
  "Services": {
    "XMPPServer": {
      "@dailyBackup": "Yes/No",
      "@enableExport": "",
      "@enabled": "true/false",
      "@backupTime": "",
      "@recoveryMode": "",
      "@port": ""
    }
  },
}
```

Example 2.11. Device Object Response XML (Continued)

```
"SSHServer": {"@enabled": "true/false"},
"DHCPServer": {
  "@enabled": "true/false",
  "@endIP": "",
  "@publishedDNSServer": "",
  "@startIP": ""
},
"XMPPAgent": {
  "@xmppServer": "",
  "DestinationAgent": {"@enabled": "true/false"},
  "AdminAgent": {"@enabled": "true/false"},
  "@enabled": "true/false",
  "@userName": "",
  "PortCtrlAgent": {"@enabled": "true/false"},
  "SourceAgent": {"@enabled": "true/false"},
  "@domain": "",
  "XAuthAgent": {"@enabled": "true/false"},
  "ConfigAgent": {"@enabled": "true/false"}
},
"TFTPServer": {"@enabled": "true/false"},
"NTPServer": {
  "@enabled": "true/false",
  "@serverIP": ""
},
"TelnetServer": {"@enabled": "true/false"},
"HTTPServer": {"@enabled": "true/false"},
"StreamingServer": {
  "RTPRXPorts": {
    "@to": "",
    "@from": ""
  },
  "@enabled": "true/false",
  "V2DRxPorts": {
    "@to": "",
    "@from": ""
  },
}
```

Example 2.12. Device Object Response XML (Continued)

```
"TXcntrlPorts": {
  "@to": "",
  "@from": ""
},
"MulticastIP": {
  "@to": "",
  "@from": ""
},
"MPEGTSRXPorts": {
  "@to": "",
  "@from": ""
},
"UDPRXPorts": {
  "@to": "",
  "@from": ""
}
},
"Network": {
  "@DNSServerIP": "",
  "@DNSName": "",
  "EthernetPortList": {"EthernetPort": {
    "@netmask": "",
    "@enabled": "true/false",
    "@useDHCP": "true/false",
    "@portID": "",
    "@isPrimary": "",
    "@multicast": "true/false",
    "@ipAddress": ""
  }},
  "NATTable": {"NAT": {
    "@name": "",
    "@ip": "",
    "@listIndex": ""
  }},
  "@isPrimaryEnabled": "",
  "@hostName": ""
},
"System": {"NTPClient": {
  "@enabled": "true/false",
  "@serverIP": ""
}},
}}
```

Example 2.13. Device Object Response XML (Continued)

```
"AVCodec": {"Port": {
  "Displayconfig": [
    {
      "@type": "IPVSCClientDisplayConfig",
      "Display": {"@configurable": ""},
      "Canvas": {
        "Background": {
          "@content": "OS/Image/Session",
          "ImageContent": {
            "@imageUrl": "",
            "@color": ""
          },
        },
        "SessionContent": {
          "@enablePlayerPanel": "",
          "@enableCtrlPanel": "",
          "@enableBookmarkPanel": "",
          "@sessionId": "",
          "@enableConnPanel": "",
          "@enableHeaderPanel": ""
        },
      },
      "@zIndex": 0,
      "@enable": "true/false"
    },
  ],
  "Window": {
    "@content": "MediaThumbnail/MediaStream",
    "ThumbnailContent": {
      "@imageUrl": "",
      "@MediaStreamIOPortElementID": "",
      "@MediaStreamIOPortTypeTitle": "",
      "@refreshInterval": ""
    },
  },
}

"Geometry": {
  "@transparency": "NNN%",
  "@hres": "NNNpx/%",
  "@border": "NONE/THIN/THICK",
  "@zIndex": "",
  "@vpos": "ALIGNTOP/ALIGNMIDDLE/ALIGNBOTTOM/NNNpx",
  "@aspect": "SourceWidth/SourceHeight/Auto/None",
  "@vres": "NNNpx/%",
  "@hpos": "ALIGNLEFT/ALIGNMIDDLE/ALIGNRIGHT/NNNpx"
},
```

Example 2.14. Device Object Response XML (Continued)

```
"ConnectionContent": {
  "@MediaStreamIOPortElementID": "",
  "@MediaStreamIOPortTypeTitle": "",
  "@sessionId": "",
  "@connId": ""
},
"@id": "userDefinedStr",
"@enable": "true/false"
}
},
{
  "Transparency": 0,
  "VerticalOffset": 0,
  "HorizontalResolution": 0,
  "@type": "XP200DisplayConfig",
  "HorizontalLocation": "Left",
  "Aspect": "Auto",
  "HorizontalOffset": 0,
  "VerticalScale": 100,
  "HorizontalPosition": "Location",
  "HorizontalTransformation": "Scale/Source",
  "KeepAspect": true,
  "Override": "None",
  "OverlayType": "Manual",
  "VerticalTransformation": "Scale/Source",
  "HorizontalScale": 100,
  "VerticalLocation": "Top",
  "VerticalPosition": "Location",
  "DisplayType": "Remote",
  "VerticalResolution": 0
}
],
"@id": ""
}}
},
  "@password": "",
"@msStatus": "",
"Groups": {"Tag": {
  "@name": "",
  "@value": ""
}},
}}
```


Example 2.15. Device Object Response XML (Continued)

```

"@softwareVersion": "",
"PortList": { "Port": {
  "Groups": "",
  "@NID": "",
  "ObserverInfoList": { "Observer": {
    "ResourceNID": "",
    "UserNID": "",
    "@xAuth": "",
    "@monitor": ""
  }},
  "@streamType": "",
  "FeatureList": { "Feature": { "@name": "" }},
  "@portType": "",
  "@presence": "",
  "@state": "",
  "@title": "",
  "StreamURL": ""
}},
"DevicePrivateData": "",
"@title": "",
"@state": "",
"@description": ""
}}

```

Port

API calls used for Ports

Get Port

This is the list of Source and Destination Ports that the Client has access to. Refer to the Appendix for the format of the Port object

Example 2.16. Get Port List Request**Table 2.5. Get Port List Request**

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port+json"
Status	200, 401, 500
Query Parameters	startFrom, countToFetch, sortOrder, sortBy, selectFormat, whereNID, whereTextSubStr, whereTagExpr, whereStreamType, wherePermissions, subscriptionID, dstPortTitle,whereState,subscriptionScope,expireTime

Resource Path	/Resource/Port
Authorization	Basic (Required)

- *Parameters are optional*
- *sortBy="title/state"*
- *selectFormat="BASIC/EXTENDED/EXTENDEDCONNECTIONINFO/BASICCONNECTIONINFO/BASICURL/EXTENDEDURL"*
- *sortBy="NID/portType/presence"* [Not Supported]
- *type="SrcPort/DstPort"*
- *whereStreamType="V2D/RTP/MPEGTS/UDP/RTP-RAW/RTP-PCM/RFB"*
- *wherePermissions="ADD,MODIFY,DELETE,VIEWFROM,STREAMTO"*
- *dstPortTitle* [Optional]
- *whereState="Ready/Busy/Error/NotReady/Offline"* [Optional]
- *expireTime* [Optional. expireTime should be in Seconds format. For permanent URL use INFINITE keyword in expireTime]
- *whereTagExpr="Name%3D'Location' and Value%3D'Sunnyvale' "* [Check for correct URL encoding when using this]

For e.g.

To get a list of Source Ports: HTTP GET http://<msip>/controller/Resource/Port

To get a list of Destination Ports: HTTP GET http://<msip>/controller/Resource

Example 2.17. NOTE about fetching URLs from the System

For all URLs that the system returns, if the destination information is not provided or is incorrect, in the 'dstPortTitle' parameter, then the URL will be for primary MS IP address. For mediaURL, if destination information is not provided, then logged-in resource will be used as destination. If this logged-in resource happens to be a 'Proxy-Client' the mediaURL may not work. For All URL's, if expireTime is not provided system will use logged in user credentials to generate validation signature. Validity of URL is tied to logged in user session. If user logged out URL is no longer valid If expireTime is provided system will generate expiry based validation signature. Validity of URL is tied to expireTime. After expiry time URL is no longer valid. For permanent URL's send INFINITE keyword in expireTime.

Example 2.18. Get Port Response - BASIC Format

```
{ "Resource": {
  "Port": {
    "Groups": { "Tag": {
      "@Name": " ",
      "@Value": " "
    } },
    "@NID": "default.mediastreamdstportresourcelist.msdpri_091145e5-a419-4",
    "@streamType": "V2D MPEGTS RTP",
    "@portType": "StreamDst",
    "@portId": 1,
    "@presence": "available",
    "@state": "Ready",
    "@title": "web-client_barco",
    "Permissions": {
      "@add": true,
      "@modify": true,
      "@delete": true,
      "@streamto": true,
      "@viewfrom": false,
      "@read": true
    }
  },
  "@total": 18,
  "@from": 1
}}
```

Example 2.19. Get Port Response - EXTENDED Format

```
{ "Resource": {
  "Port": {
    "@NID": "default.mediastreamdstportresourcelist.msdpri_2eb6ab7c-2c3e-4c10-",
    "FeatureList": { "Feature": {
      "content": true,
      "@name": "TrickPlay"
    } },
    "Description": "",
    "@portType": "StreamDst",
    "MaxInstanceCount": 16,
    "@presence": "unavailable",
    "Permissions": {
      "@add": true,
      "@modify": true,
      "@delete": true,
      "@streamto": true,
      "@viewfrom": true,
      "@read": true
    },
    "Groups": { "Tag": {
      "@Name": "",
      "@Value": ""
    } },
    "@streamType": "V2D MPEGTS RTP",
    "ObserverInfo": { "Observer": {
      "@userNID": "",
      "@xAuth": "",
      "@monitor": "",
      "@resourceNID": ""
    } },
    "PVRInfo": {
      "@mediaStreamDirResourceNID": "",
      "@sessionId": "",
      "@trimLength": "",
      "@mediaStreamFileResourceNID": "",
      "@enable": ""
    },
    "@portId": 1,
    "Asset": {
      "@NID": "default.assetresourcelist.arl_6f7a0ad0-267c-4d51-9b9f-254f5c4",
      "@deviceType": "WebClient-Device",
      "@contactJID": "",
      "@title": "web-client"
    },
    "@title": "web-client",
    "@state": "Template",
```

Example 2.20. Get Port Response - EXTENDED Format continued

```
"InstanceList": { "Instance": {
  "@streamURL": "",
  "ProfileXML": {
    "V2DEncoderConnectionParameterKbmConfig": {
      "EnableKbm": false,
      "KIdleTime": 0,
      "AllowKMSwitch": true
    },
    "V2DStreamConfig": {
      "EnableSRDTranscode": true,
      "SRDTranscodeBandwidth": "",
      "IsMulticast": true,
      "AVOption": "Both"
    },
    "MPEGEncoderParameterConfig": [
      {
        "SDP-a-Template": "",
        "VideoBitrate": "",
        "Brightness": "",
        "VideoFormat": "",
        "VideoInputSource": "",
        "DeinterlaceMode": "",
        "Contrast": "",
        "AudioBitrate": "",
        "PictureSize": "",
        "Saturation": "",
        "Hue": "",
        "VideoCompression": "",
        "AudioCompression": "",
        "Encapsulation": ""
      },
      {
        "SDP-a-Template": "",
        "VideoBitrate": "",
        "Brightness": "",
        "VideoFormat": "",
        "VideoInputSource": "",
        "DeinterlaceMode": "",
        "Contrast": "",
        "AudioBitrate": "",
        "PictureSize": "",
        "Saturation": "",
        "Hue": "",
        "VideoCompression": "",
        "AudioCompression": "",
        "Encapsulation": ""
      }
    ]
  }
},
```

Example 2.21. Get Port Response - EXTENDED Format continued

```
"MPEGTSStreamConfig": {"IsMulticast": ""},
  "V2DEncoderAudioConfig": {
    "MonoStereo": "",
    "SampleRate": "",
    "AudioEncodeEnable": "",
    "SampleSize": ""
  },
  "RTPStreamConfig": {
    "SDP-m-Template": "",
    "IsMulticast": "",
    "AVOption": ""
  },
  "SDP-Template-Src": "",
  "V2DEncoderConnectionParameterConfig": {
    "IFrameInterval": "",
    "ColorSampling": "4:2:2",
    "BurstRatio": 1,
    "FrameRateDivider": 1,
    "BlockThreshold": 5,
    "Refresh": {
      "Max": 30,
      "Min": 10
    },
    "BurstDuration": 50,
    "Slice": {
      "Max": 15,
      "Min": 0
    },
    "Bandwidth": 10000,
    "Compression": {
      "Low": 6,
      "High": 6
    }
  },
  "@mode": "Dynamic",
  "@sessionId": ""
}},
"@total": 1,
"@from": 1
}}
```

Example 2.22. Get Port Response - BASICCONNECTIONINFO Format

```
{ "Resource": {
  "Port": {
    "@NID": "",
    "FeatureList": { "Feature": { "@name": "" } },
    "Description": "",
    "@portType": "",
    "MaxInstanceCount": "",
    "@presence": "",
    "Permissions": {
      "@add": "",
      "@modify": "",
      "@delete": "",
      "@streamto": "",
      "@viewfrom": "",
      "@read": ""
    },
  },
  "Groups": { "Tag": {
    "@Name": "",
    "@Value": ""
  } },
  "@streamType": "",
  "ObserverInfo": { "Observer": {
    "@userNID": "",
    "@xAuth": "",
    "@monitor": "",
    "@resourceNID": ""
  } },
  "PVRInfo": {
    "@mediaStreamDirResourceNID": "",
    "@sessionId": "",
    "@trimLength": "",
    "@mediaStreamFileResourceNID": "",
    "@enable": ""
  },
  "@portId": "",
  "Asset": {
    "@NID": "",
    "@deviceType": "",
    "@contactJID": "",
    "@title": ""
  },
},
```

Example 2.23. Get Port Response - BASICCONNECTIONINFO Format (Continued)

```
"ConnectionStatus": {"PortSessionInfo": {
  "@name": "",
  "@id": "",
  "PortConnectionInfo": {
    "@windowId": "",
    "@errorCode": "",
    "@name": "",
    "@errorData": "",
    "@dstNID": "",
    "@srcNID": "",
    "@type": "VIEW/PLAYBACK/RECORD/PVR",
    "@dstTitle": "",
    "@canvasId": "",
    "@state": "",
    "@id": "",
    "@srcTitle": ""
  }
}},
"@title": "",
"@state": "",
"@thumbnailURL": "",
"@mediaURL": "",
"@hls": "",
  "InstanceList": {"Instance": {"@streamURL": ""}}
},
"@type": "SrcPort/DstPort/RelayPort/StorePort",
"@total": 0,
"@from": 0
}}
```


Example 2.24. Get Port Response - EXTENDEDCONNECTIONINFO Format

```
{ "Resource": {
  "Port": {
    "@NID": " ",
    "FeatureList": { "Feature": { "@name": " " } },
    "Description": " ",
    "@portType": " ",
    "MaxInstanceCount": " ",
    "@presence": " ",
    "Permissions": {
      "@add": " ",
      "@modify": " ",
      "@delete": " ",
      "@streamto": " ",
      "@viewfrom": " ",
      "@read": " "
    },
  },
  "Groups": { "Tag": {
    "@Name": " ",
    "@Value": " "
  } },
  "@streamType": " ",
  "ObserverInfo": { "Observer": {
    "@userNID": " ",
    "@xAuth": " ",
    "@monitor": " ",
    "@resourceNID": " "
  } },
  "PVRInfo": {
    "@mediaStreamDirResourceNID": " ",
    "@sessionId": " ",
    "@trimLength": " ",
    "@mediaStreamFileResourceNID": " ",
    "@enable": " "
  },
  "@portId": " ",
  "Asset": {
    "@NID": " ",
    "@deviceType": " ",
    "@contactJID": " ",
    "@title": " "
  },
},
```

Example 2.25. Get Port Response - EXTENDEDCONNECTIONINFO Format (Continued)

```
"ConnectionStatus": {"PortSessionInfo": {
  "@name": "",
  "@id": "",
  "PortConnectionInfo": {
    "@windowId": "",
    "@dstNID": "",
    "@srcUserName": "",
    "@destUserJID": "",
    "@type": "VIEW/PLAYBACK/RECORD/PVR",
    "Profile": {
      "@burstDuration": "",
      "@profileName": "",
      "@maxBandwidth": "",
      "@multicast": "",
      "@unicast": "",
      "@profileNID": "",
      "@avOption": "",
      "@id": "",
      "@allowKMSSwitch": "",
      "@burstRatio": "",
      "@lowCompression": "",
      "@remoteKM": "",
      "@highCompression": ""
    },
    "@dstTitle": "",
    "@srcTitle": "",
    "@destUserName": "",
    "@name": "",
    "@errorCode": "",
    "@errorData": "",
    "@srcNID": "",
    "@srcUserJID": "",
    "@id": "",
    "@state": "",
    "@canvasId": ""
  },
  },
  "@title": "",
  "@state": "",
  "@thumbnailURL": "",
  "@mediaURL": "",
  "hls": "",
  "InstanceList": {"Instance": {"@streamURL": ""}}
},
"@type": "SrcPort/DstPort/RelayPort/StorePort",
"@total": 0,
"@from": 0
}}
```

Example 2.26. Get Port Response - BASICURL Format

```
{ "Resource": {  
  "Port": {  
    "@NID": " ",  
    "@streamType": " ",  
    "@portType": " ",  
    "@portId": " ",  
    "@presence": " ",  
    "@mediaURL": " ",  
    "@state": " ",  
    "@title": " ",  
    "Permissions": {  
      "@add": " ",  
      "@modify": " ",  
      "@delete": " ",  
      "@streamto": " ",  
      "@viewfrom": " ",  
      "@read": " "  
    }  
  },  
  "@type": "SrcPort/DstPort",  
  "@total": 50,  
  "@from": 0  
}}
```

Example 2.27. Get Port Response - EXTENDEDURL Format

```

{
  "Resource": {
    "Port": {
      "Groups": {
        "Tag": {
          "@name": "",
          "@value": ""
        }
      },
      "@NID": "",
      "@streamType": "",
      "FeatureList": {
        "Feature": {
          "@name": ""
        }
      },
      "@portType": "",
      "MaxInstanceCount": "",
      "description": "",
      "@portId": "",
      "Asset": {
        "@NID": "",
        "@deviceType": "",
        "@contactJID": "",
        "@title": ""
      },
      "@presence": "",
      "@mediaURL": "",
      "@state": "",
      "@title": "",
      "Permissions": {
        "@add": "",
        "@modify": "",
        "@delete": "",
        "@streamto": "",
        "@viewfrom": "",
        "@read": ""
      }
    },
    "@type": "SrcPort/DstPort",
    "@total": 50,
    "@from": 0
  }
}

```

Get Port Tags

This is the list of Tags that the Client has access to.

Example 2.28. Get Port Tags Request**Table 2.6. Get Port Tags Request**

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-Tags+json"

Status	200, 401, 500
Query Parameters	sortBy, sortOrder, whereNameTextSubString, whereValueTextSubString, wherePermissions, selectFormat
Resource Path	/Resource/Port
Authorization	Basic (Required)

- *sortBy*="Name/Value"
- *whereNameTextSubString*=" " [Optional]
- *whereValueTextSubString* [Optional]
- *wherePermissions*="ADD,MODIFY,DELETE,VIEWFROM,STREAMTO"
- *selectFormat*="NameList/NameValueList/NameValueTree"
[Default=NameValueList]

Example 2.29. Get Port Tags Response - NameList Format

```
{ "Resource": {
  "PortTags": { "Tag": [
    { "@Name": "Location" },
    { "@Name": "Network" }
  ] },
  "@total": 0,
  "@from": 0
}}
```

Example 2.30. Get Port Tags Response - NameValueList Format

```
{ "Resource": {
  "PortTags": { "Tag": [
    {
      "@Name": "Location",
      "@Value": "Sunnyvale"
    },
    {
      "@Name": "Location",
      "@Value": "Hyderabad"
    }
  ] },
  "@total": 0,
  "@from": 0
}}
```

Example 2.31. Get Port Tags Response - NameValueTree Format

```
{ "Resource": {
  "PortTags": { "Tag": {
    "Value": [
      "Sunnyvale",
      "Hyderabad"
    ],
    "@Name": "Location"
  }},
  "@total": 0,
  "@from": 0
}}
```

Get Port Display Config

Gets the display configuration for given port.

Example 2.32. Get Port Display Configuration Request**Table 2.7. GetPortDisplayConfig Request**

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-DisplayConfig+json"
Status	200, 401, 500
Query Parameters	whereNID, whereTextSubStr, subscriptionID
Resource Path	/Resource/Port
Authorization	Basic (Required)

Example 2.33. Get Port Display Config Response

```
{ "Resource": { "PortDisplayConfig": {
  "@NID": "",
  "@title": "",
  "_comment": "Look Appendix A for device specific Display Config"
}}}
```

Get Port Display Status

Gets the display status for given port.

Example 2.34. Get Port Display Status Request**Table 2.8. GetPortDisplayStatus Request**

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-DisplayStatus+json"
Status	200, 401, 500
Query Parameters	whereNID, whereTextSubStr, subscriptionID
Resource Path	/Resource/Port
Authorization	Basic (Required)

Example 2.35. Get Port Display Status Response

```
{ "Resource": { "PortDisplayStatus": {  
  "@NID": " ",  
  "@title": " "  
  "_comment": "Look Appendix A for device specific Display Status"  
}}}
```

SetPortInfo

Update port title, description, and groups in the database

Example 2.36. Set Port Info Request**Table 2.9. Set Port Info Request**

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-Info+json"
Status	200, 401, 500
Query Parameters	NID or title
Resource Path	/Resource/Port

Authorization	Basic (Required)
---------------	------------------

For e.g.

To set Port info using NID: HTTP POST `http://<msip>/controller/`

`Resource/Port?NID=default.mediastreamsrcportresource.mssprl_091145e5-a419`

To set Port info using title: HTTP POST `http://<msip>/controller/`

`Resource/Port?title=rtp`

- *title* [Required [Does not allow XML restricted special characters]]
- *description* [Optional]
- *Groups* [Optional]

```
{ "PortInfo": {
  "Groups": { "Tag": {
    "@Name": " ",
    "@Value": " "
  } },
  "@title": " ",
  "@description": " "
}}
```

Example 2.37. Set Port Info Response

```
{ "Resource": { "PortInfo":
  "_comment": "Check Port Object Response XML"
}}
```

SetPortObserverInfo

Update observer information in the database

Example 2.38. Set Port Observer Info Request

Table 2.10. Set Port Observer Info Request

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-ObserverInfo+json"

Status	200, 401, 500
Query Parameters	NID or title
Resource Path	/Resource/Port
Authorization	Basic (Required)

- *xAuth(true/false)* [Optional]
- *monitor(true/false)* [Optional]
- *UserNID* [Required if xAuth/monitor present. NID of the User to be set as an Observer]
- *ResourceNID* [Required if xAuth/monitor present. NID of an asset resource in the system]

```

    {"PortObserverInfo": {
      "Observer": {
        "ResourceNID": " ",
        "UserNID": " ",
        "@xAuth": " ",
        "@monitor": " "
      },
    },
  }}

```

Example 2.39. Set Port Observer Info Response

```

{"Resource": {"PortObserverInfo":
  "_comment": "Check Port Object Response XML"
}}

```

SetPortPVRInfo

Update the PVR settings in the database

Example 2.40. Set Port PVR Info Request

Table 2.11. Set Port PVR Info Request

Request	HTTP POST
Accept (No spaces or newlines)	"application/vnd.barco.nimbus.Port-PVRInfo+json"
Status	200, 401, 500
Query Parameters	NID or title
Resource Path	/Resource/Port

Authorization	Basic (Required)
---------------	------------------

- *enable(true/false)* [Required]
- *trimLength* [Optional - In milliseconds. Default is 4hrs]
- *mediaStreamDirResourceNID* [Required. Use GetDir call to get this NID for required media directory]

```
{ "PortPVRInfo": {
"@mediaStreamDirResourceNID": " ",
"@trimLength": " ",
"@enable": "true/false"
}}
```

Example 2.41. Set Port PVR Info Response

```
{ "Resource": { "PortPVRInfo":
  "_comment": "Check Port Object Response XML"
}}
```

SetPortStreamURL

Update port stream URL in the database

Example 2.42. Set Port StreamURL Request

Table 2.12. Set Port StreamURL Request

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-StreamURL+json"
Status	200, 401, 500
Query Parameters	NID or title
Resource Path	/Resource/Port
Authorization	Basic (Required)

- *streamURL* [Optional]

```
{ "PortStreamURL": {
"@streamURL": " ",
}}
```

Example 2.43. Set Port StreamURL Response

```
{ "Resource": { "PortStreamURL":  
  "_comment": "Check Port Object Response XML"  
}}
```

Set Port Display Config

updates the given display configuration.

Tip

Suggest get config from the database update the required parameters and send the set port display configuration request

Example 2.44. Set Port Display Configuration Request**Table 2.13. Set Port Display Config**

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-DisplayConfig+json"
Status	200, 401, 500
Query Parameters	NID or title
Resource Path	/Resource/Port
Authorization	Basic (Required)

```
{"PortDisplayConfig": ""}  
<!--Check the appendix for device specific config-->
```

Example 2.45. Set Port Display Configuration Response

```
{"PortDisplayConfig": ""}
```

Set Port Window Properties

Sets the unmanned client window Properties. Note: Send all properties or one or combination.

Example 2.46. Set Port Window Properties Request**Table 2.14. Set Port Window Properties**

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Port-WindowProperties+json"
Status	200, 401, 500
Query Parameters	NID
Resource Path	/Resource/Port

Authorization	Basic (Required)
---------------	------------------

```
{ "PortWindowProperties":
  "@byWindowId": "",
  { "WindowProperties": {
    "@scrollHPos": "",
    "@scale": "",
    "@maximize": "true/false",
    "@border": "",
    "@hres": "",
    "@aspect": "",
    "@transparency": "",
    "@scrollVPos": "",
    "@remoteKbmCtrl": "true/false",
    "@vpos": "",
    "@layoutPos": "",
    "@vres": "",
    "@visibility": "true/false",
    "@hpos": "",
    "@fps": "",
    "@mute": "true/false",
    "@volume": "50%"
  }}}
```

- *fps* [optional]
- *maximize*='true/false' [optional]
- *mute*='true/false' [optional]
- *remoteKbmCtrl*='true/false' [optional]
- *scale* [optional]
- *visibility*='true/false' [optional]
- *volume* [optional]
- *aspect* [optional]
- *border* [optional]
- *hpos* [optional]
- *vpos* [optional]
- *hres* [optional]
- *vres* [optional]
- *layoutPos* [optional]
- *scrollHPos* [optional]
- *scrollVPos* [optional]
- *transparency* [optional]

Example 2.47. Set Port Window Properties Response

```
{"PortWindowProperties": ""}
```

Port Object Response XML

Example 2.48. Port Object Response XML

```
{ "Port": {
  "@NID": "",
  "FeatureList": { "Feature": { "@name": "" } },
  "Description": "",
  "@portType": "",
  "MaxInstanceCount": "",
  "@presence": "",
  "Permissions": {
    "@add": "",
    "@modify": "",
    "@delete": "",
    "@streamto": "",
    "@viewfrom": "",
    "@read": ""
  },
},
"Groups": { "Tag": {
  "@Name": "",
  "@Value": ""
} },
"@streamType": "",
"ObserverInfo": { "Observer": {
  "@userNID": "",
  "@xAuth": "",
  "@monitor": "",
  "@resourceNID": ""
} },
"PVRInfo": {
  "@mediaStreamDirResourceNID": "",
  "@sessionId": "",
  "@trimLength": "",
  "@mediaStreamFileResourceNID": "",
  "@enable": ""
},
"Asset": {
  "@NID": "",
  "@deviceType": "",
  "@deviceNID": "",
  "@title": ""
},
"@title": "",
"@state": "",
"InstanceList": { "Instance": { "@streamURL": "" } }
}
```

Dir

API calls used for Dir.

Get Dir

This is the list of Media Directories that the Client has access to.

Example 2.49. Get Dir Request

Table 2.15. Get Dir Request

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Dir+json"
Status	200, 401, 500
Query Parameters	countToFetch, selectFormat, sortBy, sortOrder, startFrom, whereDateRange, whereNID, whereTagExpr, whereTextSubStr, whereParentDirNID, wherePermissions, whereFilePermissions, subscriptionID, subscriptionScope, expireTime, dstPortTitle
Resource Path	/Resource/Dir
Authorization	Basic (Required)

- *sortBy*="title/state/size/sizeUsed/dateCreated/dateModified"
- *sortBy*="NID" [Not Supported]
- *whereNID*="DirNID" [dirNID Required]
- *whereParentDirNID*="parentDirNID" [Optional]
- *wherePermissions*="MODIFY,DELETE,IMPORT,RECORD" [Optional]
- *whereFilePermissions*="MODIFY,DELETE,EXPORT,PLAYBACK" [Optional]
- *subscriptionID* [Optional]
- *selectFormat*="BASIC/SNAPSHOTURL(Deprecated)/UPLOADURL" [Optional.
default='BASIC']
- *expireTime* [Optional. expireTime should be in Seconds format. Used along with UPLOADURL selectFormat. For permanent URL use INFINITE keyword in expireTime]
- *whereDateRange*="From:YYYY-MM-DD HH:MM:SS.MMMM, To:YYYY-MM-DD HH:MM:SS.MMMM, Interval:xx(Y/M/D/H/m/S/s)" [Optional]
- *dstPortTitle* [Optional]

Example 2.50. Get Dir Response - BASIC Format

```
{ "Resource": {
  "Dir": [
    {
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@sizeUsed": "",
      "@NID": "",
      "@dateCreated": "",
      "@dateModified": "",
      "@state": "",
      "@title": "",
      "Permissions": {
        "@record": "",
        "@import": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@readOnly": "",
      "@size": ""
    },
    {
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@sizeUsed": "",
      "@NID": "",
      "@dateCreated": "",
      "@dateModified": "",
      "@state": "",
      "@title": "",
      "Permissions": {
        "@record": "",
        "@import": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@readOnly": "",
      "@size": ""
    }
  ],
  "@total": 50,
  "@from": 0
}}
```

Example 2.51. Get Dir Response - SNAPSHOTURL Format (Deprecated)

```
{ "Resource": {
  "Dir": [
    {
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@sizeUsed": "",
      "@NID": "",
      "@dateCreated": "",
      "@dateModified": "",
      "@state": "",
      "@title": "",
      "Permissions": {
        "@record": "",
        "@import": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@readOnly": "",
      "@size": ""
    },
    {
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@sizeUsed": "",
      "@NID": "",
      "@dateCreated": "",
      "@dateModified": "",
      "@state": "",
      "@title": "",
      "Permissions": {
        "@record": "",
        "@import": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@readOnly": "",
      "@size": "",
      "@snapshotURL": ""
    }
  ],
  "@total": 50,
  "@from": 0
}}
```

Example 2.52. Get Dir Response - UPLOADURL Format

```
{ "Resource": {
  "Dir": [
    {
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@sizeUsed": "",
      "@NID": "",
      "@dateCreated": "",
      "@dateModified": "",
      "@state": "",
      "@title": "",
      "Permissions": {
        "@record": "",
        "@import": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@readOnly": "",
      "@size": ""
    },
    {
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@sizeUsed": "",
      "@NID": "",
      "@dateCreated": "",
      "@dateModified": "",
      "@state": "",
      "@title": "",
      "Permissions": {
        "@record": "",
        "@import": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@readOnly": "",
      "@size": "",
      "@uploadURL": ""
    }
  ],
  "@total": 50,
  "@from": 0
}}
```

Get Dir Tags

This is the list of Tags that the Client has access to.

Example 2.53. Get Dir Tags Request

Table 2.16. Get Dir Tags Request

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Dir-Tags+json"
Status	200, 401, 500
Query Parameters	sortBy, sortOrder, whereNameTextSubStr, whereValueTextSubStr, wherePermissions, selectFormat
Resource Path	/Resource/Dir
Authorization	Basic (Required)

- *sortBy*="Name/Value"
- *whereNameTextSubString*=" " [Optional]
- *whereValueTextSubString* [Optional]
- *wherePermissions*="MODIFY,DELETE,IMPORT,RECORD" [Optional]
- *selectFormat*="NameList/NameValueList/NameValueTree"
[Default=NameValueList]

Example 2.54. Get Dir Tags Response - NameList Format

```
{ "Resource": { "DirTags": {
  "Tag": [
    { "@Name": "Location" },
    { "@Name": "Network" }
  ],
  "@total": 0,
  "@from": 0
}}}
```

Example 2.55. Get Dir Tags Response - NameValueList Format

```
{ "Resource": { "DirTags": {  
  "Tag": [  
    {  
      "@Name": "Location",  
      "@Value": "Sunnyvale"  
    },  
    {  
      "@Name": "Location",  
      "@Value": "Hyderabad"  
    }  
  ],  
  "@total": 0,  
  "@from": 0  
}}}
```

Example 2.56. Get Dir Tags Response - NameValueTree Format

```
{ "Resource": { "Tag": {  
  "Value": [  
    "Sunnyvale",  
    "Hyderabad"  
  ],  
  "@Name": "Location"  
}}}
```

File

API calls used for File

Get File

This is the list of Files that the Client has access to. If there are Files of type MediaGroup then the associated MediaClip Files will be returned as nested sub-elements of the Media Group File.

Example 2.57. Get File Request**Table 2.17. Get File Request**

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.File+json"
Status	200, 401, 500

Query Parameters	countToFetch, selectFormat, sortBy, sortByMetaData, sortByTagName, sortOrder, startFrom, whereType, whereMediaGroupID, whereDateRange, whereNID, whereTagExpr, whereTextSubStr, searchTextSubStrIn, whereState, whereMetaDataExpr, whereParentDirNID, whereSessionConfigNID, wherePermissions, subscriptionID, subscriptionScope, expireTime, dstPortTitle, whereDownloadState
Resource Path	/Resource/File

Authorization	Basic (Required)
<ul style="list-style-type: none"> • <i>searchTextSubStrIn=TITLE,DESCRIPTION,METADATA,BOOKMARK</i> [Used along with whereTextSubStr] • <i>sortBy="title/state/playlength/dateCreated/size"</i> [Either sortBy/sortByMetaData/sortByTagName] • <i>sortByMetaData</i> [Either sortBy/sortByMetaData/sortByTagName] • <i>sortByTagName</i> [Either sortBy/sortByMetaData/sortByTagName] • <i>sortBy="NID"</i> [Not Supported] • <i>whereParentDirNID=" "</i> [Optional] • <i>whereState="Ready/Busy/Error/NotReady/Offline"</i> [Optional] • <i>whereMetaDataExpr="Name%3D'TestAPI' and Value%3D'RecordTest' "</i> [Optional] • <i>whereType="MediaClip,MediaGroup,MediaImage,ALL"</i> [Optional Default=ALL] • <i>whereMediaGroupID=" "</i> [Optional] • <i>whereSessionConfigNID=" "</i> [Optional] • <i>wherePermissions="MODIFY,DELETE,PLAYBACK,IMPORT"</i> [Optional] • <i>subscriptionID</i> [Optional] • <i>expireTime</i> [Optional. expireTime should be in Seconds format. Used along with BASICURL or EXTENDEDURL selectFormat. For permanent URL use INFINITE keyword in expireTime] • <i>selectFormat="BASIC/EXTENDED/BASICURL/EXTENDEDURL/BASICCONNECTIONINFO"</i> [Optional. Default=BASIC] • <i>dstPortTitle</i> [Optional] • <i>whereDateRange="From:YYYY-MM-DD HH:MM:SS.MMMM Z, To:YYYY-MM-DD HH:MM:SS.MMMM Z, Interval:xx(Y/M/D/H/m/S/s)"</i> [Optional] <p>For example from 2014-01-01 00:00:00.0000 +05:30 to 2014-01-02 00:00:00.0000 +05:30 (give value in percent encoded form as below):</p> <ol style="list-style-type: none"> 1. <i>whereDateRange="From:2014-01-01%2000%3A00%3A00.000%20%2B05%3A30, To:2014-01-02%2000%3A00%3A00.000%20%2B05%3A30"</i> 2. <i>whereDateRange="From:2014-01-01%2000%3A00%3A00.000%20%2B05%3A30, Interval:01D"</i> 3. <i>whereDateRange="To:2014-01-02%2000%3A00%3A00.000%20%2B05%3A30, Interval:01D"</i> <ul style="list-style-type: none"> • <i>whereDownloadState="Ready/Busy/NotReady"</i> [Optional (filter for cached files)] • Refer to Example 2.17 for Note related to URLs 	

For e.g. Get all the recorded medioclips:

`http://<msip>/controller/Resource/File?whereType=Mediaclip`

Example 2.58. Sample sortByTagName="Location"

```
{
  "Groups": {
    "Tag": {
      "@Name": "Location",
      "@Value": "Sunnyvale"
    }
  }
}
```

Example 2.59. Sample sortByMetaData="TestAPI"

```
{
  "MetaData": {
    "Tag": {
      "@Name": "TestAPI",
      "content": "RecordTest"
    }
  }
}
```


Example 2.60. Get File Response - EXTENDED Format

```
{ "Resource": {
  "File": [
    {
      "@NID": "",
      "Permissions": {
        "@playback": "",
        "@export": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@size": "",
      "@readOnly": "",
      "@bypassAutoStorageRules": "",
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@streamType": "",
      "MetaData": { "Tag": {
        "content": "Value",
        "@Name": ""
      } },
      "@dateCreated": "",
      "@fileType": "MediaClip",
      "@dateModified": "",
      "@playlength": "",
      "@title": "",
      "@state": "",
      "@mediaGroupId": "",
      "@ffTracks": ""
    },
  ],
}
```

Example 2.61. Get File Response - EXTENDED Format (Continued)

```
{
  "@NID": "",
  "@presence": "",
  "Permissions": {
    "@playback": "",
    "@export": "",
    "@modify": "",
    "@delete": "",
    "@read": ""
  },
  "@size": "",
  "@readOnly": "",
  "@bypassAutoStorageRules": "",

  "Groups": {"Tag": {
    "@Name": "",
    "@Value": ""
  }},
  "@streamType": "",
  "MetaData": {"Tag": {
    "content": "Value",
    "@Name": ""
  }},
  "@dateCreated": "",
  "@fileType": "MediaGroup",
  "@dateModified": "",
  "@playlength": "",
```

Example 2.62. Get File Response - EXTENDED Format (Continued)

```
"FileList": {"File": {
  "@NID": "",
  "@presence": "",
  "Permissions": {
    "@playback": "",
    "@export": "",
    "@modify": "",
    "@delete": "",
    "@read": ""
  },
  "@size": "",
  "@readOnly": "",
  "@bypassAutoStorageRules": "",
  "Groups": {"Tag": {
    "@Name": "",
    "@Value": ""
  }},
  "@streamType": "",
  "MetaData": {"Tag": {
    "content": "Value",
    "@Name": ""
  }},
  "@dateCreated": "",
  "@fileType": "MediaClip",
  "@dateModified": "",
  "@playlength": "",
  "@title": "",
  "@state": "",
  "@ffTracks": "",
  "@mediaGroupId": ""
}},
"@title": "",
"@state": "",
"@ffTracks": "",
"@mediaGroupId": ""
}
],
"@total": 400,
"@from": 200
}}
```

Example 2.63. Get File Response - BASIC Format

```
{ "Resource": { "File": {
  "@NID": "",
  "Dir": {
    "@dirTitle": "",
    "@dirNID": ""
  },
  "@presence": "",
  "Permissions": {
    "@playback": "",
    "@export": "",
    "@modify": "",
    "@delete": "",
    "@read": ""
  },
  "@size": "",
  "@readOnly": "",
  "@bypassAutoStorageRules": "true/false",
  "Groups": { "Tag": {
    "@Name": "",
    "@Value": ""
  } },
  "@streamType": "",
  "MetaData": { "Tag": {
    "content": "Value",
    "@Name": ""
  } },
  "description": "",
  "@dateCreated": "",
  "@fileType": "",
  "@dateModified": "",
  "@playlength": "",
  "@title": "",
  "@state": "",
  "Download": { "Media": [
    {
      "@format": "Standard",
      "@state": "NotReady",
      "@stale": "",
    },
    {
      "@format": "Proprietary",
      "@state": "NotReady",
      "@stale": "",
    }
  ] },
  "@ffTracks": "",
  "@mediaGroupId": ""
} },
"@total": 400,
"@from": 200
}
```

Example 2.64. Get File Response - BASICURL Format

```
{ "Resource": {
  "File": {
    "@NID": "",
    "Permissions": {
      "@playback": "",
      "@export": "",
      "@modify": "",
      "@delete": "",
      "@read": ""
    },
    "@size": "",
    "@readOnly": "",
    "@bypassAutoStorageRules": "",
    "Groups": { "Tag": {
      "@Name": "",
      "@Value": ""
    } },
    "@streamType": "",
    "@thumbnailURL": "",
    "MetaData": { "Tag": {
      "content": "Value",
      "@Name": ""
    } },
    "@dateCreated": "",
    "@fileType": "MediaClip",
    "@dateModified": "",
    "@playlength": "",
    "@mediaURL": "",
    "@title": "",
    "@state": "",
    "@mediaGroupId": "",
    "@ffTracks": ""
  },
  "@total": 400,
  "@from": 200
}}
```

Example 2.65. Get File Response - EXTENDEDURL Format

```
{ "Resource": { "File": {
  "@NID": "",
  "Dir": {
    "@dirTitle": "",
    "@dirNID": ""
  },
  "@presence": "",
  "Permissions": { "@playback": "", "@export": "", "@modify": "", "@delete": "" },
  "@size": "",
  "@readOnly": "",
  "@bypassAutoStorageRules": "true/false",
  "Groups": { "Tag": {
    "@Name": "",
    "@Value": ""
  } },
  "@streamType": "",
  "MetaData": { "Tag": {
    "content": "Value",
    "@Name": ""
  } },
  "description": "",
  "@thumbnailURL": "",
  "@dateCreated": "",
  "@fileType": "",
  "@dateModified": "",
  "@playlength": "",
  "@title": "",
  "@mediaURL": "",
  "@state": "",
```

Example 2.66. Get File Response - EXTENDEDURL Format (Continued)

```
"Download": {"Media": [  
  {  
    "@format": "Standard",  
    "@state": "NotReady",  
    "@stale": "",  
    "@downloadURL": ""  
  },  
  {  
    "@format": "Proprietary",  
    "@state": "NotReady",  
    "@stale": "",  
    "@downloadURL": ""  
  }  
]},  
"@ffTracks": "",  
"@mediaGroupId": ""  
}}  
  
"@total": 400,  
"@from": 200  
}}
```

Example 2.67. Get File Response - BASICCONNECTIONINFO Format

```
{ "Resource": { "File": {
  "@NID": "",
  "Dir": {
    "@dirTitle": "",
    "@dirNID": ""
  },
  "@presence": "",
  "Permissions": { "@playback": "", "@export": "", "@modify": "", "@delete": "" },
  "@size": "",
  "@readOnly": "",
  "@bypassAutoStorageRules": "true/false",
  "Groups": { "Tag": {
    "@Name": "",
    "@Value": ""
  } },
  "@streamType": "",
  "MetaData": { "Tag": {
    "content": "Value",
    "@Name": ""
  } },
  "description": "",
  "@thumbnailURL": "",
  "@dateCreated": "",
  "@fileType": "",
  "@dateModified": "",
  "@playlength": "",
  "@title": "",
  "@mediaURL": "",
  "@state": "",
```


Example 2.68. Get File Response - BASICCONNECTIONINFO Format (Continued)

```
"Download": {"Media": [
  {
    "@format": "Standard",
    "@state": "NotReady",
    "@stale": "",
    "@downloadURL": ""
  },
  {
    "@format": "Proprietary",
    "@state": "NotReady",
    "@stale": "",
    "@downloadURL": ""
  }
]},
"ConnectionStatus": {"FileSessionInfo": {
  "@name": "",
  "@id": "",
  "FileConnectionInfo": {
    "@windowId": "",
    "@errorCode": "",
    "@name": "",
    "@errorData": "",
    "@dstNID": "",
    "@srcNID": "",
    "@type": "VIEW/PLAYBACK/RECORD/PVR",
    "@dstTitle": "",
    "@canvasId": "",
    "@state": "",
    "@id": "",
    "@srcTitle": ""
  }
}},
  "@ffTracks": "",
  "@mediaGroupId": ""
}}

"@total": 400,
"@from": 200
}}
```

Get File Tags

This is the list of Tags that the Client has access to.

Example 2.69. Get File Tags Request**Table 2.18. Get File Tags Request**

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.File-Tags+json"
Status	200, 401, 500
Query Parameters	sortBy, sortOrder, whereNameTextSubString, whereValueTextSubString, wherePermissions, selectFormat
Resource Path	/Resource/File
Authorization	Basic (Required)

- *sortBy*="Name/Value"
- *whereNameTextSubString*=" " [Optional]
- *whereValueTextSubString* [Optional]
- *wherePermissions*="MODIFY,DELETE,PLAYBACK,EXPORT" [Optional]
- *selectFormat*="NameList/NameValueList/NameValueTree"
[Default=NameValueList]

Example 2.70. Get File Tags Response - NameList Format

```
{ "Resource": {
  "FileTags": { "Tag": [
    { "@Name": "Location" },
    { "@Name": "Network" }
  ] },
  "@total": 0,
  "@from": 0
}}
```

Example 2.71. Get File Tags Response - NameValueList Format

```
{ "Resource": {
  "FileTags": { "Tag": [
    {
      "@Name": "Location",
      "@Value": "Sunnyvale"
    },
    {
      "@Name": "Location",
      "@Value": "Hyderabad"
    }
  ] },
  "@total": 0,
  "@from": 0
}}
```

Example 2.72. Get File Tags Response - NameValueTree Format

```
{ "Resource": {
  "FileTags": { "Tag": {
    "Value": [
      "Sunnyvale",
      "Hyderabad"
    ],
    "@Name": "Location"
  } },
  "@total": 0,
  "@from": 0
}}
```

SetFileMetaData

Replaces file metadata in database

- *MetaData* [Required]

Example 2.73. Set File MetaData Request by NID**Table 2.19. Set File MetaData Request by NID or title**

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.File-MetaData+json"
Status	200, 401, 500

Query Parameters	NID or title
Resource Path	/Resource/File
Authorization	Basic (Required)

```
{ "FileMetaData": {  
  "MetaData": { "Tag": [  
    {  
      "content": 12345,  
      "@Name": "ID"  
    },  
    {  
      "content": "UpdateWithByNID",  
      "@Name": "TestCase"  
    }  
  ] },  
}
```

Example 2.74. Set File MetaData Response

```
{ "SetFileMetaData": { "File": {
  "@NID": " ",
  "Dir": {
    "@dirTitle": " ",
    "@dirNID": " "
  },
  "@presence": " ",
  "@size": " ",
  "@readOnly": " ",
  "@bypassAutoStorageRules": " ",
  "Groups": { "Tag": {
    "@name": " ",
    "@value": " "
  } },
  "@streamType": " ",
  "MetaData": { "Tag": {
    "content": "Value",
    "@Name": " "
  } },
  "Permissions": {
    "@playback": true,
    "@export": true,
    "@modify": true,
    "@delete": true,
    "@read": true
  },
  "description": " ",
  "@dateCreated": " ",
  "@fileType": " ",
  "@dateModified": " ",
  "@playlength": " ",
  "@title": " ",
  "@state": " ",
  "@mediaGroupId": " ",
  "Download": { "Media": {
    "@format": " ",
    "@state": "NotReady",
    "@stale": " "
  } }
} }
```

Delete File

Delete the given file from the system

- *NID* [Required. NID of the file you want to delete]

Example 2.75. DeleteFile Request**Table 2.20. DeleteFile Request**

Request	HTTP DELETE
Accept	"application/json"
Status	200, 401, 500
Query Parameters	NID, moveToRecycleBin
Resource Path	/Resource/File
Authorization	Basic (Required)

- *moveToRecycleBin*='true/false' [if the value is true it will move the file to recycle bin. Otherwise it will delete from the system.Default is True]

Example 2.76. DeleteFile Response

```
{ "Resource": { "File": {
  "@NID": "default.mediastreamfileresourcelist.msfrl_72ed6633-4072-45cb-b785-4dd",
  "Dir": {
    "@NID": "default.mediastreamdirresourcelist.msdr1_04e5116b-940c-4acc-aa3e-",
    "@dirTitle": "RECYCLEBIN"
  },
  "@clipID": "091b8ef9-99e3-4028-abf3-c54fd6836334",
  "@mediaID": "387cf54f-7667-43eb-be8b-b4752b9cd3c2",
  "Permissions": {
    "@playback": true,
    "@export": true,
    "@modify": true,
    "@delete": true,
    "@read": true
  },
  "@size": 21752456,
  "@readOnly": false,
  "@bypassAutoStorageRules": false,
  "Groups": { "Tag": {
    "@Name": "",
    "@Value": ""
  } },
  "@streamType": "RTP",
  "@hls": true,
  "MetaData": { "Tag": [
    {
      "content": 12345,
      "@Name": "ID"
    },
    {
      "content": "UpdateWithByNID",
      "@Name": "TestCase"
    }
  ] },
}
```

Example 2.77. DeleteFile Response.. continued

```
"description": "",
"@dateCreated": 1394129863550,
"@fileType": "MediaClip",
"@dateModified": 1394129863550,
"@playlength": 77180,
"@title": "axis7412",
"@state": "Deleted",
"@sessionConfigNID": "",
"@mediaGroupID": "",
"Download": {"Media": {
  "@format": "Standard",
  "@state": "Prepared",
  "@stale": true
}}
}}}
```

Prepare File For download

Prepare file For download to client PC. Please subscribe for file updates using a subscriptionID and selectFormat as BASICURL/EXTENDEDURL in GetFile request, to get the file notification with a downloadURL, when media is Ready for download. Any edits to the media may cause the prepared status of the file to become stale, but still return a downloadURL on GetFile. It is the user's discretion whether to download a stale copy of the media or to issue another prepare to update the cached media for download.

- *byNID* [fileNID required]
- *format="Proprietary/Standard/Both"* [Optinal Default='Proprietary']

Example 2.78. Prepare File For download Request**Table 2.21. Prepare File For download Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	Resource/File/PrepareForDownload

Authorization	Basic (Required)
---------------	------------------

```
{
  "PrepareFileForDownload": {
    "@byNID": "",
    "@format": ""
  }
}
```

Example 2.79. Prepare File For Download Response

```
{
  "PrepareFileForDownload": {
    "File": {
      "@NID": "",
      "Dir": {
        "@dirTitle": "",
        "@dirNID": ""
      },
      "@presence": "",
      "Permissions": {
        "@playback": "",
        "@export": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@size": "",
      "@readOnly": "",
      "@bypassAutoStorageRules": "true/false",
      "Groups": {
        "Tag": {
          "@Name": "",
          "@Value": ""
        }
      },
      "@streamType": "",
      "MetaData": {
        "Tag": {
          "content": "Value",
          "@Name": ""
        }
      },
      "description": "",
      "@dateCreated": "",
      "@fileType": "",
      "@dateModified": "",
      "@playlength": "",
      "@title": "",
      "@state": "",
      "Download": {
        "Media": [
          {
            "@format": "Standard",
            "@state": "Busy",
            "@stale": "",
            "@downloadURL": ""
          },
          {
            "@format": "Proprietary",
            "@state": "Busy",
            "@stale": "",
            "@downloadURL": ""
          }
        ]
      },
      "@ffTracks": "",
      "@mediaGroupId": ""
    }
  }
}
```

Create Media From Cutlist

Creates Media from existing media using start and end offsets within it. Generates a file notification when the new media creation is complete. Please subscribe for file updates using a subscriptionID in GetFile request for receiving file notifications.

- *byNID* [fileNID of existing media - required]
- *cutList*="StartOffset:EndOffset,StartOffset:EndOffset" [offsets of existing media in milliseconds. E.g. 0:30000 is 30 seconds of media from start offset. - optional. Skip the value of cutList if you wish to create a copy of parent file]

Note: Cutlist values are validated against certain rules in the backend and will be rejected as invalid if not in proper order. Cutlist values should be integer values in an ascending order and they should not overlap. (for e.g. 0:30000,20000:40000 is an invalid cutlist range)

- *title* [Title for new media - required]
- *MetaData* [MetaData for new media - Optional. Skip the metadata element from request, if you wish to copy over metadata from parent file]
- *Description* [Description for new media - required]

Example 2.80. Create Media From Cutlist

Table 2.22. Create Media From Cutlist

Request	HTTP POST
Accept	"application/json"
Status	201, 401, 500
Query Parameters	None
Resource Path	Resource/File/CreateMediaFromCutList
Authorization	Basic (Required)

```
{ "CreateMediaFromCutList": {
  "@byNID": " ",
  "@cutList": "StartOffset:EndOffset,StartOffset:EndOffset",
  "@title": " ",
  "MetaData": { "Tag": {
    "content": "Value",
    "@Name": " "
  } },
  "Description": " ",
}
```

Example 2.81. Create Media From Cutlist Response

```
{
  "CreateMediaFromCutList": {
    "File": {
      "@NID": "default.mediastreamfileresourcelist.msfrl_a9504438-5bb1-4e09-a389-c60",
      "Dir": {
        "@NID": "default.mediastreamdirresourcelist.msdr1_4e09075d-cf15-44f8-98c3-",
        "@dirTitle": "srcrelay1"
      },
      "@clipID": "",
      "@mediaID": "480193eb-7930-4d04-8d7a-0af670f2e5dd",
      "Permissions": {
        "@playback": true,
        "@export": true,
        "@modify": true,
        "@delete": true,
        "@read": true
      },
      "@size": "",
      "@readOnly": false,
      "@bypassAutoStorageRules": false,
      "Groups": {
        "Tag": [
          {
            "@Name": "Location",
            "@Value": "Sunnyvale"
          },
          {
            "@Name": "Users",
            "@Value": "Admin"
          }
        ]
      },
      "@streamType": "",
      "MetaData": {
        "Tag": {
          "content": "Value",
          "@Name": ""
        }
      },
      "description": "",
      "@dateCreated": 1397505139401,
      "@fileType": "MediaClip",
      "@dateModified": 1397505139401,
      "@playlength": 0,
      "@title": "tx4",
      "@state": "NotReady",
      "@sessionConfigNID": "",
      "@mediaGroupID": "",
      "Download": {
        "Media": {
          "@format": "",
          "@state": "NotReady",
          "@stale": ""
        }
      }
    }
  }
}
```

Trim File with Cutlist

Trim out sections of an existing mediafile using start and end offsets within it. Generates a file notification when the trimmed media is ready. Please subscribe for file updates using a subscriptionID in GetFile request for receiving file notifications.

- *byNID* [fileNID of existing media - required]
- *cutList*="*StartOffset:EndOffset,StartOffset:EndOffset*" [offsets of existing media in milliseconds. E.g. 0:30000 is 30 seconds of media from start offset. - required]

Note: Cutlist values are validated against certain rules in the backend and will be rejected as invalid if not in proper order. Cutlist values should be integer values in an ascending order and they should not overlap. (for e.g. 0:30000,20000:40000 is an invalid cutlist range)

Example 2.82. Trim File with Cutlist

Table 2.23. Trim File with Cutlist

Request	HTTP POST
Accept	"application/json"
Status	202, 401, 500
Query Parameters	None
Resource Path	Resource/File/Trim
Authorization	Basic (Required)

```
{ "TrimFile": {
  "@byNID": " ",
  "@cutList": "StartOffset:EndOffset ,StartOffset:EndOffset "
}}
```

Example 2.83. Trim File with Cutlist Response

```
{
  "TrimFile": {
    "File": {
      "@NID": "default.mediastreamfileresourcelist.msfrl_a9504438-5bb1-4e09-a38",
      "Dir": {
        "@NID": "default.mediastreamdirresourcelist.msdr1_4e09075d-cf15-4",
        "@dirTitle": "srcrelay1"
      },
      "@clipID": "",
      "@mediaID": "480193eb-7930-4d04-8d7a-0af670f2e5dd",
      "Permissions": {
        "@playback": true,
        "@export": true,
        "@modify": true,
        "@delete": true,
        "@read": true
      },
      "@size": "",
      "@readOnly": false,
      "@bypassAutoStorageRules": false,
      "Groups": {
        "Tag": [
          {
            "@Name": "Location",
            "@Value": "Sunnyvale"
          },
          {
            "@Name": "Users",
            "@Value": "Admin"
          }
        ]
      },
      "@streamType": "",
      "MetaData": {
        "Tag": {
          "content": "Value",
          "@Name": ""
        }
      },
      "description": "",
      "@dateCreated": 1397505139401,
      "@fileType": "MediaClip",
      "@dateModified": 1397505139401,
      "@playlength": 0,
      "@title": "tx4",
      "@state": "Busy",
      "@sessionConfigNID": "",
      "@mediaGroupID": "",
      "Download": {
        "Media": {
          "@format": "",
          "@state": "NotReady",
          "@stale": ""
        }
      }
    }
  }
}
```

Clear Download Cache

Free the disk space which is occupied by cached media file, after it is downloaded.

- *byNID* [fileNID of existing media - required]

Example 2.84. Clear Download Cache

Table 2.24. Clear Download Cache

Request	HTTP POST
Accept	"application/json"
Status	202, 200, 401, 500
Query Parameters	None
Resource Path	Resource/File/ClearDownloadCache
Authorization	Basic (Required)

```
{ "ClearDownloadCache": {  
  "@byNID": " ",  
}  
}
```

Example 2.85. Clear Download Cache Response

```
        {"ClearDownloadCache": {"File": {
"@NID": "default.mediastreamfileresourcelist.msfrl_a9504438-5bb1-4e09-a389-c60
"Dir": {
    "@NID": "default.mediastreamdirresourcelist.msdr1_4e09075d-cf15-44f8-98c3-
    "@dirTitle": "srcrelay1"
},
"@clipID": "",
"@mediaID": "480193eb-7930-4d04-8d7a-0af670f2e5dd",
"Permissions": {
    "@playback": true,
    "@export": true,
    "@modify": true,
    "@delete": true,
    "@read": true
},
"@size": "",
"@readOnly": false,
"@bypassAutoStorageRules": false,
"Groups": {"Tag": [
    {
        "@Name": "Location",
        "@Value": "Sunnyvale"
    },
    {
        "@Name": "Users",
        "@Value": "Admin"
    }
]}},
"@streamType": "",
"MetaData": {"Tag": {
    "content": "Value",
    "@Name": ""
}},
"description": "",
"@dateCreated": 1397505139401,
"@fileType": "MediaClip",
"@dateModified": 1397505139401,
"@playlength": 0,
"@title": "tx4",
"@state": "Ready",
"@sessionConfigNID": "",
"@mediaGroupID": "",
"Download": {"Media": {
    "@format": "",
    "@state": "Busy",
    "@stale": ""
}}
}}}
```


Inject Media

Uploads the provided media file to the media store. To get the injectURL, you can use the Get DIR request with format as INJECTURL.

Example 2.86. Inject Media

Sample inject URL:

```
http://<DMS_IP>/mediajs/file/upload?dirID=/data1
&timestamp=1389123260606
&apiKey=-1110313941673189477
&signature=fd154a45722592daa9c63ba4ccf8130e
&sparam=bc8afb2b6575e6032ddc4eb90c8d0aa2
```

You can upload the file by appending the query parameters to the injectURL and sending HTTP POST with this URL. This should return 200:OK

Sample injectURL with appended parameters:

```
http://<DMS_IP>/mediajs/file/upload?dirID=/data1
&timestamp=1389123260606
&apiKey=-1110313941673189477
&signature=fd154a45722592daa9c63ba4ccf8130e
&sparam=bc8afb2b6575e6032ddc4eb90c8d0aa2
&title=new_title
&patientId=1234
&patientName=new_patient
&patientBirthDate=01011980
&patientGender=male
&accessionNumber=34534
&doctorName=doctor_who
```

In case server configuration "Enable HLS Streaming" is true and you do not want to create hls then append the query parameter "options=donotcreatehls" to the injectURL.

Sample inject URL if you do not want to create hls:

```
http://<DMS_IP>/mediajs/file/upload?dirID=/data1
&timestamp=1389123260606
&apiKey=-1110313941673189477
&signature=fd154a45722592daa9c63ba4ccf8130e
&sparam=bc8afb2b6575e6032ddc4eb90c8d0aa2
&options=donotcreatehls
```

Download Media

Download a media file from the media store on the system. To get the downloadURL, you can use the Get File request with format as EXTENDEDURL. Send an HTTP GET request using this downloadURL to download and save media to local machine. This should return 200:OK

Example 2.87. Download Media

Sample Download URL:

```
http://<DMS_IP>/mediajs/file/download?mediaID=316581ac-5168-42ae-adc7-e6c6609
&dirID=/data1
&timestamp=1389367743953
&apiKey=5754498016574178712
&signature=0895c6bd4404d70ab62f6f85104fa0f5
&format=proprietary
&uuid=txl_89e9d445_53c6_4b7f_ae84_4c9b7f9e4905
&persistence=temporary
```

Note: if downloadURL not present in the response for Get File, may suggest that the File is not prepared for download. Please use Prepare File Request to make the file available for download. This will result in a File Notification which includes the downloadURL for the file. Please subscribe for file updates using a subscriptionID in GetFile request in order to receive any file notifications.

User

API calls for User

Get User

This is the list of Users that the Client has access to.

Example 2.88. Get User Request

Table 2.25. Get User Request

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.User+json"
Status	200, 401, 500
Query Parameters	startFrom, countToFetch, sortBy, sortOrder, whereNID, whereUsername, selectFormat, whereTextSubStr, whereTagExpr, subscriptionID

Resource Path	/Resource/User
Authorization	Basic (Required)

- *sortBy*="username/fullName" [Optional]
- *subscriptionID* [Optional]

Example 2.89. Get User Response - BASIC Format

```
{ "Resource": {  
  "User": [  
    {  
      "Groups": { "Tag": {  
        "@Name": " ",  
        "@Value": " "  
      }},  
      "@fullName": " ",  
      "@NID": " ",  
      "@externalAuthentication": " ",  
      "@presence": " ",  
      "@username": " ",  
      "@state": " ",  
      "@email": " ",  
      "@modifiedDate": " ",  
      "@creationDate": " "  
    },  
    {  
      "Groups": { "Tag": {  
        "@Name": " ",  
        "@Value": " "  
      }},  
      "@fullName": " ",  
      "@NID": " ",  
      "@externalAuthentication": " ",  
      "@presence": " ",  
      "@username": " ",  
      "@state": " ",  
      "@email": " ",  
      "@modifiedDate": " ",  
      "@creationDate": " "  
    },  
  ],  
}
```

Example 2.90. Get User Response - BASIC Format (Continued)

```
{
  "Groups": {"Tag": {
    "@Name": "",
    "@Value": ""
  }},
  "@fullName": "",
  "@NID": "",
  "@externalAuthentication": "",
  "@presence": "",
  "@username": "",
  "@state": "",
  "@email": "",
  "@modifiedDate": "",
  "@creationDate": ""
},
"@total": 50,
"@from": 0
}}
```

Example 2.91. Get User Response - EXTENDED Format

```
{"Resource": {
  "User": {
    "Groups": {"Tag": {
      "@Name": "",
      "@Value": ""
    }},
    "@fullName": "",
    "@NID": "",
    "@externalAuthentication": "",
    "@presence": "",
    "@username": "",
    "@state": "",
    "UserPrivateData": "",
    "@email": "",
    "@modifiedDate": "",
    "@creationDate": ""
  },
  "@total": 50,
  "@from": 0
}}
```

Get User Tags

This is the list of Tags that the Client has access to.

Example 2.92. Get User Tags Request

Table 2.26. Get User Tags Request

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.User-Tags+json"
Status	200, 401, 500
Query Parameters	sortBy, sortOrder, whereNameTextSubString, whereValueTextSubStr, selectFormat
Resource Path	/Resource/User
Authorization	Basic (Required)

- *sortBy*="Name/Value"
- *whereNameTextSubString*=" " [Optional]
- *whereValueTextSubString* [Optional]
- *selectFormat*="NameList/NameValueList/NameValueTree"
[Default=NameValueList]

Example 2.93. Get User Tags Response - NameList Format

```
{ "Resource": {
  "@total": 0,
  "UserTags": { "Tag": [
    { "@Name": "Location" },
    { "@Name": "Network" }
  ] },
  "@from": 0
}}
```

Example 2.94. Get User Tags Response - NameValueList Format

```
{ "Resource": {
  "@total": 0,
  "UserTags": { "Tag": [
    {
      "@Name": "Location",
      "@Value": "Sunnyvale"
    },
    {
      "@Name": "Location",
      "@Value": "Hyderabad"
    }
  ] },
  "@from": 0
}}
```

Example 2.95. Get User Tags Response - NameValueTree Format

```
{ "Resource": {
  "@total": 0,
  "UserTags": { "Tag": {
    "Value": [
      "Sunnyvale",
      "Hyderabad"
    ],
    "@Name": "Location"
  } },
  "@from": 0
}}
```

Set User Password

This is used to set the user password.

Example 2.96. Set User Password Request**Table 2.27. Set User Password Request**

Request	HTTP POST
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.User-Password+json"
Status	200, 401, 500
Query Parameters	NID
Resource Path	/Resource/User

Authorization	Basic (Required)
---------------	------------------

```
{ "UserPassword": {
  "@oldPassword": " ",
  "@newPassword": " "
}}
```

Example 2.97. Set User Password Response

```
{ "Resource": { "User": {
  "Groups": { "Tag": {
    "@Name": " ",
    "@Value": " "
  }},
  "@fullName": " ",
  "@NID": " ",
  "@externalAuthentication": " ",
  "@presence": " ",
  "@username": " ",
  "@state": " ",
  "UserPrivateData": " ",
  "@email": " ",
  "@modifiedDate": " ",
  "@creationDate": " ",
  "@password": " "
}}}
}
```

Get Tags

This is the list of Tags that the Client has access to.

Example 2.98. Get Tags Request

Table 2.28. Get Tags Request

Request	HTTP GET
Accept(No spaces or newlines)	"application/vnd.barco.nimbus.Tags+json"
Status	200, 401, 500
Query Parameters	sortBy, sortOrder, whereNameTextSubString, whereValueTextSubString, wherePermissions, whereObjectType, selectFormat
Resource Path	/Resource/Tags

Authorization	Basic (Required)
---------------	------------------

- *sortBy*="Name/Value"
- *whereNameTextSubString*="" [Optional]
- *whereValueTextSubString* [Optional]
- *selectFormat*="NameList/NameValueList/NameValueTree"
[Default=NameValueList]
- *whereObjectType*="Device/Users/MediaRoom/Profiles/Ports/Media/ALL"

Example 2.99. Get Tags Response - NameList Format

```
{ "Resource": {
  "Tags": { "Tag": [
    {
      "@Name": "Location",
      "@objectType": "Devices/Users/MediaRoom/Profiles/Ports/Media/ALL"
    },
    {
      "@Name": "Network",
      "@objectType": "Devices/Users/MediaRoom/Profiles/Ports/Media/ALL"
    }
  ] },
  "@total": 0,
  "@from": 0
}}
```

Example 2.100. Get Tags Response - NameValueList Format

```
{ "Resource": {
  "Tags": { "Tag": [
    {
      "@Name": "Location",
      "@objectType": "Devices/Users/MediaRoom/Profiles/Ports/Media/ALL",
      "@Value": "Sunnyvale"
    },
    {
      "@Name": "Location",
      "@objectType": "Devices/Users/MediaRoom/Profiles/Ports/Media/ALL",
      "@Value": "Hyderabad"
    }
  ] },
  "@total": 0,
  "@from": 0
}}
```

Example 2.101. Get Tags Response - NameValueTree Format

```

{
  "Resource": {
    "Tags": {
      "Tag": {
        "Value": [
          "Sunnyvale",
          "Hyderabad"
        ],
        "@Name": "Location",
        "@objectType": "Devices/Users/MediaRoom/Profiles/Ports/Media/ALL"
      }
    },
    "@total": 0,
    "@from": 0
  }
}

```

Get DisplayProfile

This is the list of DisplayProfile objects that the Client has access to.

Example 2.102. Get DisplayProfile Request**Table 2.29. Get DisplayProfile Request**

Request	HTTP GET
Accept	"application/vnd.barco.nimbus.DisplayProfile+json"
Status	200, 401, 500
Query Parameters	selectFormat, sortBy, sortOrder, countToFetch, startFrom, whereDateRange, whereNID, whereTagExpr, whereTextSubStr
Resource Path	Resource/DisplayProfile
Authorization	Basic (Required)

- *sortBy*="title"
- *whereTextSubStr*="" [Optional]
- *whereTagExpr* [Optional]
- *selectFormat*="BASIC/EXTENDED" [Default=BASIC]

Example 2.103. Get DisplayProfile Response - BASIC Format

```
{
  "Resource": {
    "DisplayProfile": {
      "@NID": "default.displayprofilelist.dpl_1d31d3f3-f0ea-42a1-8f6a-cac8b3",
      "@priority": "",
      "@portType": "",
      "@title": "2x1_Canvas_1920x1080"
    },
    "@total": 12,
    "@from": 1
  }
}
```

Example 2.104. Get DisplayProfileResponse - EXTENDED Format

```

        "@layoutId": "LayoutId1",
        "@vres": "100%",
        "@layoutPos": "",
        "ResourceAbility": true,
        "@hpos": "0%",
        "@fps": 15,
        "@mute": false,
        "@volume": "50%"
    },
    {
        "@content": "",
        "ThumbnailContent": {
            "@imageUrl": "",
            "@MediaStreamIOPortElementID": "",
            "@MediaStreamIOPortTypeTitle": "",
            "@refreshInterval": ""
        },
        "ConnectionContent": {
            "@hres": "",
            "@srcKbmCtrl": "",
            "@MediaStreamIOPortElementID": "",
            "@vres": "",
            "@MediaStreamIOPortTypeTitle": "",
            "@sessionId": "",
            "@connId": ""
        },
        "@id": "Win2",
        "Properties": {
            "@scrollHPos": 0,
            "@scale": "fitWindow",
            "@border": "",
            "@hres": "50%",
            "@maximize": false,
            "@zIndex": 100,
            "@aspect": "",
            "@scrollVPos": 0,
            "@transparency": 0,
            "@remoteKbmCtrl": false,
            "@vpos": "0%",
            "@layoutId": "LayoutId1",
            "@vres": "100%",
            "@layoutPos": "",
            "@visibility": true,
            "@hpos": "50%",
            "@fps": 15,
            "@mute": false,
            "@volume": "50%"
        }
    }
],
    },
    "@title": "2x1_Canvas_1920x1080"
},
"@total": 12,
"@from": 1
}
}

```

Chapter 3. Resource Update Notifications

Resource ConnectionUpdate Notification

- *sessionId* [session id]
- *id* [connection id]
- *name* [connection name]
- *state* [connection state: Check connectionNotification State listing for enum values]
- *errorCode* [optional if there is no error.]
- *errorData* [error description. optional if there is no error.]
- *windowId* [window id]
- *canvasId* [canvas Id]
- *SrcEntity:id* [source entity id]
- *SrcEntity:resourceTitle* [source title]
- *SrcEntity:resourceNID* [source nid]
- *SrcEntity:userJID* [source user JID]
- *SrcEntity:userFullName* [source user full name]
- *DstEntity:id* [destination entity id]
- *DstEntity:resourceTitle* [destination title]
- *DstEntity:resourceNID* [destination nid]
- *DstEntity:userJID* [destination user JID]
- *DstEntity:userFullName* [destination user full name]
- *Profile:id* [id of profile]
- *Profile:profileNID* [Pofile nid]
- *Profile:profileName* [Pofile Name]
- *Profile:allowKMSSwitch*='true/false' [allow key boeard mouse switch]
- *Profile:remoteKM*='true/false' [describes remote key boeard mouse allowed or not]
- *Profile:avOption*='Audio/Video/Both' [information about audio video options]
- *Profile:maxBandwidth* [bandwidth used for this connection]
- *Profile:lowCompression* [low compression used for this connection]
- *Profile:highCompression* [high compression used for this connection]

- *Profile:burstRatio* [burst Ratio used for this connection]
- *Profile:burstDuration* [burst duration used for this connection]
- *Profile:unicast='true/false'* [information about unicast connection]
- *Profile:multicast='true/false'* [information about multicast connection]

Example 3.1. AuthConectionInfo element in AuthorizationRequired Notification

```
{ "AuthConnectionInfo": {
  "@windowId": "",
  "SrcEntity": {
    "@userJID": "",
    "@resourceTitle": "",
    "@id": "",
    "@userFullName": "",
    "@resourceNID": ""
  },
  "@errorCode": "",
  "@name": "",
  "@errorData": "",
  "@type": "VIEW/PLAYBACK/RECORD/PVR",
  "Profile": {
    "@burstDuration": "",
    "@profileName": "",
    "@maxBandwidth": "",
    "@multicast": "",
    "@unicast": "",
    "@profileNID": "",
    "@avOption": "",
    "@id": "",
    "@allowKMSwitch": "",
    "@burstRatio": "",
    "@lowCompression": "",
    "@remoteKM": "",
    "@highCompression": ""
  },
  "@sessionId": "",
  "@canvasId": "",
  "@state": "",
  "@id": "",
  "DstEntity": {
    "@userJID": "",
    "@resourceTitle": "",
    "@id": "",
    "@userFullName": "",
    "@resourceNID": ""
  }
}
}}
```

Example 3.2. PortConectionInfo element in PortConnectionUpdate Notification - BASICCONNECTIONINFO format

```
        {"ConnectionStatus": {"PortSessionInfo": {
"@name": "",
"@id": "",
"PortConnectionInfo": {
    "@windowId": "",
    "@errorCode": "",
    "@name": "",
    "@errorData": "",
    "@dstNID": "",
    "@srcNID": "",
    "@type": "VIEW/PLAYBACK/RECORD/PVR",
    "@dstTitle": "",
    "@canvasId": "",
    "@state": "",
    "@id": "",
    "@srcTitle": ""
}}}
    }
```

Example 3.3. PortConectionInfo element in PortConnectionUpdate Notification - EXTENDEDCONNECTIONINFO format

```
        {"ConnectionStatus": {"PortSessionInfo": {
"@name": "",
"@id": "",
"PortConnectionInfo": {
    "@windowId": "",
    "@dstNID": "",
    "@srcUserName": "",
    "@destUserJID": "",
    "@type": "VIEW/PLAYBACK/RECORD/PVR",
    "Profile": {
        "@burstDuration": "",
        "@profileName": "",
        "@maxBandwidth": "",
        "@multicast": "",
        "@unicast": "",
        "@profileNID": "",
        "@avOption": "",
        "@id": "",
        "@allowKMSwitch": "",
        "@burstRatio": "",
        "@lowCompression": "",
        "@remoteKM": "",
        "@highCompression": ""
    },
    "@dstTitle": "",
    "@srcTitle": "",
    "@destUserName": "",
    "@name": "",
    "@errorCode": "",
    "@errorData": "",
    "@srcNID": "",
    "@srcUserJID": "",
    "@id": "",
    "@state": "",
    "@canvasId": "",
    "@thumbnailURL": ""
    }
    }
    }
    }
```


Example 3.4. FileConnectionInfo element in FileConnectionUpdate Notification - BASICCONNECTIONINFO format

```
        {"ConnectionStatus": {"FileSessionInfo": {
"@name": "",
"@id": "",
"FileConnectionInfo": {
    "@windowId": "",
    "@errorCode": "",
    "@name": "",
    "@errorData": "",
    "@dstNID": "",
    "@srcNID": "",
    "@type": "VIEW/PLAYBACK/RECORD/PVR",
    "@dstTitle": "",
    "@canvasId": "",
    "@state": "",
    "@id": "",
    "@srcTitle": ""
}}}
    }
```

Resource ConnectionStateUpdate Notification

Check ConnState and Timeline notifications in SessionNotifications for paramter listing

Example 3.5. ConnectionState element in ConnectionStateUpdate Notification

```
{ "ConnectionState": {
  "@errorCode": " ",
  "@errorData": " ",
  "@value": " ",
  "@sessionId": " ",
  "Timeline": {
    "Last": {
      "@requestedAction": "Pause",
      "@TimeScale": 0,
      "@WC": 7444674,
      "@Offset": 9678687,
      "@TC": " "
    },
    "Start": {
      "@parentTimelineId": " ",
      "@WC": 2341234,
      "@Offset": 3456,
      "@TC": " "
    },
    "@connGroupId": " ",
    "@sessionId": " ",
    "@id": " ",
    "@connId": " "
  },
  "@connId": " "
}}
```

Port

This is a list of update notifications corresponding to the port.

Port Update

This is a notification for port update.

Example 3.6. Port Update Notification BASIC

```
{ "PortUpdate": { "Port": {
"@NID": " ",
"@streamType": " ",
"@portType": " ",
"@presence": " ",
"@operation": "Add/Delete/Update",
"@state": " ",
"@title": " ",
"@subscriptionID": " ",
"@selectFormat": "BASIC"
}}}

```

Example 3.7. Port Update Notification EXTENDED

```
{ "PortUpdate": { "Port": {
"Groups": { "Tag": {
"@name": " ",
"@value": " "
}},
"@NID": " ",
"@streamType": " ",
"FeatureList": { "Feature": { "@name": " " } },
"@portType": " ",
"MaxInstanceCount": " ",
"description": " ",
"Asset": {
"@NID": " ",
"@deviceType": " ",
"@deviceNID": " ",
"@title": " "
},
"@presence": " ",
"@operation": "Add/Delete/Update",
"@state": " ",
"@title": " ",
"@subscriptionID": " ",
"@selectFormat": "EXTENDED"
}}}

```

Port Connection Update

This is a notification for port connection update.

Example 3.8. Port Connection Update Notification

```
        {"PortConnectionUpdate": {"Port": {
"@NID": "",
"@operation": "Add/Delete",
"@subscriptionID": "",
"PortConnectionInfo": {".....": ""}
}}}}
<!-- Check PortConnectionInfo element in ConnectionUpdate Section of Update Notifi
```

Port Connection State Update

This is a notification for port connection state update.

Example 3.9. Port Connection State Update Notification

```
        {"PortConnectionStateUpdate": {"Port": {
"@NID": "",
"ConnectionState": {"...": ""},
"@subscriptionID": ""
}}}}
<!-- check ConnectionState element in ConnectionStateUpdate section in Update N
```

Port Display Config Update

This is a notification for port Display Config Updates.

Example 3.10. Port DisplayConfigUpdate Notification

```
        {"PortDisplayConfigUpdate": {"Port": {
"@NID": "",
"@operation": "Update",
"@subscriptionID": ""
}}}}
        <!--Check the appendix for device specific config-->
```

Port Display Status Update

This is a notification for port Display Status Updates.

Example 3.11. Port DisplayStatusUpdate Notification

```
    {"PortDisplayStatusUpdate": {"Port": {  
      "@NID": " ",  
      "@operation": "Update",  
      "@subscriptionID": " "  
    }}}  
    <!--Check the appendix for device specific Status-->
```

File

This is a list of update notifications corresponding to the File.

File Update

This is a notification for File update.

Example 3.12. File Update Notification BASIC

```
{ "FileUpdate": { "File": {
  "@NID": " ",
  "Permissions": {
    "@playback": " ",
    "@export": " ",
    "@modify": " ",
    "@delete": " ",
    "@read": " "
  },
  "@size": " ",
  "@readOnly": " ",
  "@bypassAutoStorageRules": "true/false",
  "Groups": { "Tag": {
    "@Name": " ",
    "@Value": " "
  } },
  "@streamType": " ",
  "MetaData": { "Tag": {
    "content": "Value",
    "@Name": " "
  } },
  "@dateCreated": " ",
  "@fileType": "MediaClip",
  "@dateModified": " ",
  "@playlength": " ",
  "@title": " ",
  "@state": " ",
  "@mediaGroupId": " ",
  "@ffTracks": " "
  } }
}
```

Example 3.13. File Update Notification EXTENDED

```
{ "FileUpdate": { "File": {
  "@NID": "",
  "Dir": {
    "@dirTitle": "",
    "@dirNID": ""
  },
  "@presence": "",
  "Permissions": {
    "@playback": "",
    "@export": "",
    "@modify": "",
    "@delete": "",
    "@read": ""
  },
  "@size": "",
  "@readOnly": "",
  "@bypassAutoStorageRules": "true/false",
  "Groups": { "Tag": {
    "@Name": "",
    "@Value": ""
  } },
  "@streamType": "",
  "MetaData": { "Tag": {
    "content": "Value",
    "@Name": ""
  } },
  "description": "",
  "@dateCreated": "",
  "@fileType": "",
  "@dateModified": "",
  "@playlength": "",
  "@title": "",
  "@state": "",
  "Download": { "Media": [
    {
      "@format": "Standard",
      "@state": "NotReady",
      "@stale": "",
    },
    {
      "@format": "Proprietary",
      "@state": "NotReady",
      "@stale": "",
    }
  ] },
  "@ffTracks": "",
  "@mediaGroupId": ""
} } }
```

Example 3.14. File Update Notification EXTENDEDURL

```
{
  "FileUpdate": {
    "File": {
      "@NID": "",
      "Dir": {
        "@dirTitle": "",
        "@dirNID": ""
      },
      "@presence": "",
      "Permissions": {
        "@playback": "",
        "@export": "",
        "@modify": "",
        "@delete": "",
        "@read": ""
      },
      "@size": "",
      "@readOnly": "",
      "@bypassAutoStorageRules": "true/false",
      "Groups": {
        "Tag": {
          "@Name": "",
          "@Value": ""
        }
      },
      "@streamType": "",
      "MetaData": {
        "Tag": {
          "content": "Value",
          "@Name": ""
        }
      },
      "description": "",
      "@dateCreated": "",
      "@fileType": "",
      "@dateModified": "",
      "@playlength": "",
      "@title": "",
      "@state": "",
      "Download": {
        "Media": [
          {
            "@format": "Standard",
            "@state": "NotReady",
            "@stale": "",
            "@downloadURL": ""
          },
          {
            "@format": "Proprietary",
            "@state": "NotReady",
            "@stale": "",
            "@downloadURL": ""
          }
        ]
      },
      "@ffTracks": "",
      "@mediaGroupId": ""
    }
  }
}
```


File Connection Update

This is a notification for file connection update.

Example 3.15. File Connection Update Notification

```
      {"FileConnectionUpdate": {"File": {
"@NID": " ",
"@operation": "Add/Delete",
"@subscriptionID": " ",
"FileConnectionInfo": {".....": " "}}
}}}
<!-- Check FileConnectionInfo element in ConnectionUpdate Section of Update Notifi
```

File Connection State Update

This is a notification for file connection state update.

Example 3.16. File Connection State Update Notification

```
 {"FileConnectionStateUpdate": {"File": {
  "@NID": " ",
  "ConnectionState": {"...": " "},
  "@subscriptionID": " "
}}}
<!-- check ConnectionState element in ConnectionStateUpdate section in Update N
```

Chapter 4. Session API

API Calls used in Session

Session Info

API calls for Session Info

Get Session Config

This is the list of runtime Session Configurations that the Client has access to. Refer to the Appendix for the format of the Session object

Example 4.1. Get Session Config Request

Table 4.1. Get Session Config Request

Request	HTTP POST
Accept(No spaces or newlines)	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

- *startFrom*
- *countToFetch*
- *sortBy*
- *sortOrder*
- *whereNID*
- *whereTagExpr*
- *whereTextSubStr*
- *subscriptionID*

Example 4.2. Get Session Config Request

```
{ "GetSessionConfig": {  
  "@sortBy": "",  
  "@whereNID": "",  
  "@sortOrder": "ascending/descending",  
  "@startFrom": 1,  
  "@whereTagExpr": "",  
  "@subscriptionID": "",  
  "@countToFetch": 100,  
  "@whereTextSubStr": "",  
  "@selectFormat": "BASIC/EXTENDED"  
}}
```

Example 4.3. Get Session Config Response - BASIC Format

```
{ "GetSessionConfig": {  
  "@total": 50,  
  "SessionConfig": [  
    {  
      "Groups": { "Tag": {  
        "@Name": "",  
        "@Value": ""  
      }},  
      "@playbackOption": "",  
      "@NID": "",  
      "@type": "",  
      "@dateCreated": "",  
      "@title": ""  
    },  
    {  
      "Groups": { "Tag": {  
        "@Name": "",  
        "@Value": ""  
      }},  
      "@playbackOption": "",  
      "@NID": "",  
      "@type": "",  
      "@dateCreated": "",  
      "@title": ""  
    }  
  ],  
  "@from": 0  
}}
```

Example 4.4. Get Session Config Response - EXTENDED Format

```
{
  "GetSessionConfig": {
    "@total": 50,
    "SessionConfig": {
      "Groups": { "Tag": {
        "@Name": "",
        "@Value": ""
      } },
      "@NID": "",
      "description": "",
      "@title": "",
      "SessionInfo": ""
    },
    "@from": 0
  }
}
```

Get Session

[Not Implemented]

This is the list of runtime Sessions that the Client has access to. Refer to the Appendix for the format of the Session object

Example 4.5. Get Session Request**Table 4.2. Get Session Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

- *startFrom* [Not Supported]
- *countToFetch* [Not Supported]
- *sortBy* [Not Supported]
- *sortOrder* [Not Supported]
- *whereNID* [Not Supported]
- *whereTagExpr* [Not Supported]
- *whereTextSubStr* [Not Supported]

Example 4.6. Get Session Request

```
{ "GetSession": {  
  "@sortBy": "",  
  "@whereNID": "",  
  "@sortOrder": "ascending/descending",  
  "@startFrom": 1,  
  "@whereTagExpr": "",  
  "@subscriptionID": "",  
  "@countToFetch": 100,  
  "@whereTextSubStr": "",  
  "@selectFormat": "BASIC/EXTENDED"  
}}
```

Example 4.7. Get Session Response - BASIC Format[Not Implemented]

```
{ "GetSession": {  
  "Session": [  
    {  
      "@name": "",  
      "@recordTimelineId": "",  
      "@type": "",  
      "@timelineId": "",  
      "@state": "",  
      "@id": "",  
      "Properties": {  
        "@dateCreated": "",  
        "@ownerUserJID": "",  
        "@playlength": "",  
        "@sessionConfigNID": ""  
      }  
    },  
    {  
      "@name": "",  
      "@recordTimelineId": "",  
      "@type": "",  
      "@timelineId": "",  
      "@state": "",  
      "@id": "",  
      "Properties": {  
        "@dateCreated": "",  
        "@ownerUserJID": "",  
        "@playlength": "",  
        "@sessionConfigNID": ""  
      }  
    },  
    {  
      "@name": "",  
      "@recordTimelineId": "",  
      "@type": "",  
      "@timelineId": "",  
      "@state": "",  
      "@id": "",  
      "Properties": {  
        "@dateCreated": "",  
        "@ownerUserJID": "",  
        "@playlength": "",  
        "@sessionConfigNID": ""  
      }  
    }  
  ],  
  "@total": 50,  
  "@from": 0  
}}
```

Example 4.8. Get Session Response - BASIC Format

```
{ "GetSession": {
  "Session": [
    {
      "@name": "",
      "@recordTimelineId": "",
      "@type": "",
      "@timelineId": "",
      "@state": "",
      "@id": "",
      "Properties": {
        "@dateCreated": "",
        "@ownerUserJID": "",
        "@sessionConfigNID": ""
      }
    },
    {
      "@name": "",
      "@recordTimelineId": "",
      "@type": "",
      "@timelineId": "",
      "@state": "",
      "@id": "",
      "Properties": {
        "@dateCreated": "",
        "@ownerUserJID": "",
        "@sessionConfigNID": ""
      }
    },
    {
      "@name": "",
      "@recordTimelineId": "",
      "@type": "",
      "@timelineId": "",
      "@state": "",
      "@id": "",
      "Properties": {
        "@dateCreated": "",
        "@ownerUserJID": "",
        "@sessionConfigNID": ""
      }
    }
  ],
  "@total": 50,
  "@from": 0
}}
```

Example 4.9. Get Session Response - EXTENDED Format

```
{ "GetSession": {  
  "Session": {  
    "Groups": { "Tag": {  
      "@Name": "",  
      "@Value": ""  
    }},  
    "TimelineList": { "Timeline": {  
      "Last": {  
        "@requestedAction": "",  
        "@TimeScale": "",  
        "@WC": "",  
        "@Offset": "",  
        "@TC": ""  
      },  
      "Start": {  
        "@parentTimelineId": "",  
        "@WC": "",  
        "@Offset": "",  
        "@TC": ""  
      },  
      "@id": ""  
    }},  
  },  
}
```


Example 4.10. Get Session Response - EXTENDED Format (Continued)

```
"@name": "",
"@recordTimelineId": "",
"MetaData": {"Tag": {"@Name": ""}},
"@type": "",
"@timelineId": "",
"GrpFileList": {"GrpFile": [
  {
    "@syncEntityId": "",
    "MetaData": {"Tag": {"@Name": ""}},
    "@type": "SyncPly",
    "@dateCreated": "",
    "@recSessionConfigId": "",
    "@resourceTitle": "",
    "@id": "",
    "@recSessionId": "",
    "@resourceNID": ""
  },
  {
    "MetaData": {"Tag": {"@Name": ""}},
    "@type": "SyncRec",
    "@resourceTitle": "",
    "@id": "",
    "@resourceNID": ""
  }
]},
"@state": "",
"@id": "",
"ConnectionList": {"Connection": {
  "@streamURL": "",
  "@errorCode": "",
  "@dstEntityId": "",
  "@srcEntityId": "",
  "@errorData": "",
  "@timelineId": "",
  "@state": "",
  "@id": "",
  "ConnSeg": {
    "@profileTitle": "",
    "@errorCode": "",
    "@errorData": "",
    "@profileNID": "",
```

118

Example 4.12. Get Session Response - EXTENDED Format (Continued)

```
    "@errorData": "",
    "@type": "DstPort",
    "@gc": "",
    "@presence": "",
    "@state": "",
    "@id": ""
  },
  {
    "@errorCode": "",
    "@streamType": "",
    "@name": "",
    "@errorData": "",
    "@type": "PlyFile",
    "File": {
      "@playLength": "",
      "@type": "",
      "@startOffset": "",
      "@grpFileId": "",
      "@dirId": "",
      "@resourceTitle": "",
      "@resourceNID": ""
    },
    "@gc": "",
    "@presence": "",
    "@state": "",
    "@id": ""
  },
  {
    "@errorCode": "",
    "@streamType": "",
    "@name": "",
    "@errorData": "",
    "@type": "RecFile",
    "File": {
      "@action": "",
      "@key": "",
      "MetaData": { "Tag": { "@Name": "" } },
      "@type": "",
      "@grpFileId": "",
      "@dirId": "",
      "@resourceTitle": "",

```

Example 4.13. Get Session Response - EXTENDED Format (Continued)

```
"@keyMetaDataKeyName": "",
  "@resourceNID": ""
},
"@gc": "",
"@presence": "",
"@state": "",
"@id": ""
}
]},
"Properties": {
  "@dateCreated": "",
  "@ownerUserJID": "",
  "@sessionConfigNID": ""
}
},
"@total": 50,
"@from": 0
}}
```

Session Requests

API calls for Session Requests

Parameter Listing

This is the list of Parameters in Session Setup Requests

1. Common Parameters

- *sessionId* [Required]
- *sessionConfigId* [Optional]

2. GrpFile Parameters [Either resourceNID or resourceTitle Required]

- *type*="SyncRec/SyncPly" [Required]
- *id* [Required]
- *resourceNID* [Required]
- *resourceTitle* [Required]

3. Entity Parameters

- *type*="SrcPort/DstPort/PlyFile/RecFile/PVRFile" [Required]
- *id* [Required]

- Port Parameters [Either resourceNID or resourceTitle Required]
 - *resourceNID* [Required]
 - *resourceTitle* [Required]
- PVR Parameters [TBD]
- File Parameters [Either resourceNID or resourceTitle Required]
 - *resourceNID* [Required]
 - *resourceTitle* [Required]
 - *grpFileId* [Optional]
 - *key="Title/MetaData"* [Optional]
 - *action="Append/Overwrite"* [Optional]
 - *keyMetaDataKeyName* [Optional]
 - *MetaData element* [Optional]

4. Connection Parameters

- *id* [Required]
- *srcEntityId* [Required]
- *dstEntityId* [Required]

StartSession

- Note on Groups:

The Groups tag is used to set the Group of the recordings created as a result of this session

- Note on GrpFile:

GrpFile is referenced by Files in the SessionRequest to indicate their corresponding media group file information

GrpFile.resourceTitle with (type=SyncRec) is used to set the title of the GroupFile for the recordings created as a result of this session

- Note on EntityList:

This is the list of sources,destinations in the session

Entity where type=SrcPort is a live source referened by its title or NID

Entity.PVR.enable where type=SrcPort indicates that this should have PVR

Entity where type=DstPort is a destination that can receive from a live or playback source refernced by its title or NID

Entity where type=RecFile is a recording file to be created with the specified title

Entity.File.grpFileId indicates that this recording will be group under the GrpFile element with the same id

Entity where type=PlyFile is a recording file to be played back referenced by the specified NID

- Note on ConnectionList:

This is the list of connections in the session

Connection.srcEntityId references the source entity with the matching id

Connection.dstEntityId references the destination entity with the matching id

Connection.id will set the id for the connection between the source and destination

Connection.type will set the type of connection. Possible Connections are 'VIEW/PLAYBACK/RECORD/PVR'

Example 4.14. Basic Start Recording Session Request

Table 4.3. Basic Start Recording Session Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session

Authorization	Basic (Required)
---------------	------------------

```
{
  "StartSession": {
    "Session": {
      "ProfileList": {
        "Profile": {
          "@name": "",
          "@profileNID": "",
          "@id": "profileid",
          "@ffTracks": ""
        }
      },
      "Groups": {
        "Tag": {
          "@Name": "Location",
          "@Value": "Sunnyvale"
        }
      },
      "MetaData": {
        "Tag": {
          "@Name": ""
        }
      },
      "GrpFileList": {
        "GrpFile": {
          "MetaData": {
            "Tag": {
              "@Name": ""
            }
          },
          "@type": "SyncRec",
          "@resourceTitle": "GroupFileTitle",
          "@id": "gfId",
          "@resourceNID": ""
        }
      },
      "DirList": {
        "Dir": {
          "@resourceTitle": "",
          "@id": "dirId",
          "@resourceNID": ""
        }
      },
      "@id": "recordSessionConfigNID",
      "ConnectionList": {
        "Connection": [
          {
            "@windowId": "",
            "@dstEntityId": "dst1",
            "@srcEntityId": "src1",
            "@profileId": "profileId",
            "@canvasId": "",
            "@id": "strmc1"
          }
        ]
      }
    }
  }
}
```

Example 4.15. Basic Start Recording Session Response (Continued)

```
    },
    {
      "@windowId": "",
      "@dstEntityId": "recordd1",
      "@srcEntityId": "src1",
      "@profileId": "profileId",
      "@canvasId": "",
      "@id": "recordc1"
    },
    {
      "@windowId": "",
      "@dstEntityId": "dst2",
      "@srcEntityId": "src2",
      "@profileId": "profileId",
      "@canvasId": "",
      "@id": "strmc2"
    },
    {
      "@windowId": "",
      "@dstEntityId": "recordd2",
      "@srcEntityId": "src2",
      "@profileId": "profileId",
      "@canvasId": "",
      "@id": "recordc2"
    }
  ]},
  "EntityList": {"Entity": [
    {
      "PVR": {
        "@recEntityId": "",
        "@plyEntityId": "",
        "@trimLength": "",
        "@connID": "",
        "@SessionId": "",
        "@ffTracks": "",
        "@startTC": "",
        "@enable": true
      },
      "Port": {
        "@trickPlay": "",
        "@resourceTitle": "tx1",
        "@resourceNID": ""
      },
      "@type": "SrcPort",
      "@id": "src1"
    },
  ],
```


Example 4.16. Basic Start Recording Session Response (Continued)

```
{
  "Port": {
    "@trickPlay": false,
    "@resourceTitle": "playerpc",
    "@resourceNID": ""
  },
  "@type": "DstPort",
  "@id": "dst1"
},
{
  "@type": "RecFile",
  "File": {
    "@action": "",
    "@key": "",
    "MetaData": {"Tag": {"@Name": ""}},
    "@grpFileId": "gfId",
    "@resourceTitle": "title1",
    "@keyMetaDataKeyName": "",
    "@resourceNID": ""
  },
  "@id": "recorddl"
},
{
  "PVR": {
    "@recEntityId": "",
    "@plyEntityId": "",
    "@trimLength": "",
    "@connID": "",
    "@sessionId": "",
    "@ffTracks": "",
    "@startTC": "",
    "@enable": true
  },
  "Port": {
    "@trickPlay": "",
    "@resourceTitle": "rtp_tx1",
    "@resourceNID": ""
  },
  "@type": "SrcPort",
  "@id": "src2"
},
```

Example 4.17. Basic Start Recording Session Response (Continued)

```
{
  "Port": {
    "@trickPlay": true,
    "@resourceTitle": "playerpc",
    "@resourceNID": ""
  },
  "@type": "DstPort",
  "@id": "dst2"
},
{
  "@type": "RecFile",
  "File": {
    "@action": "",
    "@key": "",
    "MetaData": {"Tag": {"@Name": ""}},
    "@dirId": "",
    "@grpFileId": "gfId",
    "@resourceTitle": "title2",
    "@keyMetaDataKeyName": "",
    "@resourceNID": ""
  },
  "@id": "recordd2"
}
]},
"Properties": {
  "@dateCreated": "",
  "@ownerUserJID": "",
  "@playlength": "",
  "@sessionConfigNID": ""
}
},
"@sessionId": "recordSessionID"
}}
```

msfml

Example 4.19. Basic Start PlayBack Session Response (Continued)

```
{
  "@errorCode": "",
  "@streamType": "",
  "@name": "",
  "@errorData": "",
  "@type": "PlyFile",
  "File": {
    "@type": "",
    "@grpFileId": "",
    "@resourceTitle": "",
    "@resourceNID": "default.mediastreamfileresourcelist.\r\n\t\t",
  },
  "@gc": "",
  "@presence": "",
  "@state": "",
  "@id": "recordd1"
},
{
  "@errorCode": "",
  "@streamType": "",
  "@name": "",
  "Port": {
    "@trickPlay": false,
    "@type": "",
    "@resourceTitle": "playerpc",
    "@resourceNID": ""
  },
  "@errorData": "",
  "@type": "DstPort",
  "@gc": "",
  "@presence": "",
  "@state": "",
  "@id": "clientd2"
},
{
  "@errorCode": "",
  "@streamType": "",
  "@name": "",
  "@errorData": "",
  "@type": "PlyFile",
```

Example 4.20. Basic Start PlayBack Session Response(Continued)

```

    "File": {
      "@type": "",
      "@grpFileId": "",
      "@resourceTitle": "",
      "@resourceNID": "default.mediastreamfileresourcelist.\r\n\t\t",
    },
    "@gc": "",
    "@presence": "",
    "@state": "",
    "@id": "recordd2"
  }
]
},
"@sessionId": "playbackSessionID",
"@sessionConfigNID": "none"
}}

```

StopSession

Used to terminate Sessions.

Example 4.21. StopSession Request**Table 4.4. StopSession Request**

Request	HTTP POST
Accept(No spaces or newlines)	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{"StopSession": {"@sessionId": ""}}
```

Session Connection Operations

API calls for Session Connection Operations

View

Used to setup connections between Live sources and Playback destinations.

Use the SrcPort and DstPort to specify the resources to connect. Use either the name or id field returned in the GetPortList command. Use the srcEntityId or dstEntityId attribute specs in order to specify a connection to an existing src/dst. Use the canvasId and windowId to specify the canvas/window to be used for the connection.

Specify an optional Profile to further constrain the max available profile based on the admin settings. Again use the attribute form to reuse an existing profile.

Specify a PVR buffer length (in minutes) to request a buffered relay in the path of the connection. Use custom routing to override the default location of relays and PVR buffers.

Example 4.22. Basic View Request

A new session is always started. Returns an error if sessionId already exists. If no sessionId is specified a sessionId is generated by the system

Table 4.5. Basic View Request by Title

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{
  "View": {
    "@windowId": "",
    "@dstEntityId": "",
    "@name": "",
    "@srcEntityId": "",
    "@restartSession": true,
    "@sessionId": "session1",
    "@canvasId": "",
    "@id": "strmc1",
    "DstPort": {
      "@portNID": "",
      "@portTitle": "srcpc1",
      "@id": ""
    },
    "@sessionName": "",
    "SrcPort": {
      "@portNID": "",
      "@portTitle": "tx1",
      "@pvrEnable": false,
      "@id": ""
    }
  }
}
```

Example 4.23. Basic View Request by Title

A new session is always started. If no sessionId is specified a sessionId is generated by the system

```

        For E.g. to connect a SrcPort "tx1" to a DstPort "xp2001_CHANNEL2"
        {"View":{
"SrcPort":{
    "@portTitle":"tx1"
  },
"DstPort":{
"@portTitle":"xp2001_CHANNEL2"
}
} }

```

Example 4.24. Basic View Request by NID

A new session is always started. Returns an error if sessionId already exists. If no sessionId is specified a sessionId is generated by the system

```
{ "View": {  
    "@windowId": "",  
    "@dstEntityId": "",  
    "@name": "",  
    "@srcEntityId": "",  
    "@restartSession": true,  
    "@sessionId": "session1",  
    "@canvasId": "",  
    "@id": "strmc1",  
    "DstPort": {  
        "@portNID": "default.mediastreamdstportresourcelist.\r\n\t\t\t\tmsdprl_1d9c4",  
        "@portTitle": "",  
        "@id": ""  
    },  
    "@sessionName": "",  
    "SrcPort": {  
        "@portNID": "default.mediastreamsrcportresourcelist.\r\n\t\t\t\tmssprl_e973a",  
        "@portTitle": "",  
        "@pvrEnable": false,  
        "@id": ""  
    }  
}  
}}
```

Example 4.25. View Request with Profile Title

```
{
  "View": {
    "@name": "",
    "Profile": {
      "@profileTitle": "V2D_LAN",
      "@id": ""
    },
    "@sessionId": "mySession1234",
    "@id": "myConn1234",
    "DstPort": {
      "@portNID": "",
      "@id": ""
    },
    "@sessionName": "",
    "SrcPort": {
      "@portNID": "",
      "@pvrEnable": true,
      "@pvrBufferLength": 30,
      "@id": ""
    }
  }
}
```

Example 4.26. View Request with Profile NID

```
{
  "View": {
    "@name": "",
    "Profile": {
      "@profileNID": "nid1",
      "@id": ""
    },
    "@sessionId": "mySession1234",
    "@id": "myConn1234",
    "DstPort": {
      "@portNID": "",
      "@id": ""
    },
    "@sessionName": "",
    "SrcPort": {
      "@portNID": "",
      "@pvrEnable": true,
      "@pvrBufferLength": 30,
      "@id": ""
    }
  }
}
```


Example 4.27. View Request with PVR

[Not Implemented]

```
{ "View": {  
  "@name": "",  
  "Profile": {  
    "@profileNID": "",  
    "@id": ""  
  },  
  "@sessionId": "mySession1234",  
  "@id": "myConn1234",  
  "DstPort": {  
    "@portNID": "",  
    "@id": ""  
  },  
  "@sessionName": "",  
  "SrcPort": {  
    "@portNID": "",  
    "@pvrEnable": true,  
    "@pvrBufferLength": 30,  
    "@id": ""  
  }  
}}
```

Example 4.28. View Request to add a connection from an existing Src to a new Dst

```
{ "View": {
  "@windowId": "",
  "@dstEntityId": "",
  "@name": "",
  "@srcEntityId": "src1",
  "@restartSession": "",
  "@sessionId": "recordSessionID",
  "@canvasId": "",
  "@id": "vstrm1",
  "DstPort": {
    "@portNID": "",
    "@portTitle": "srcpc1",
    "@id": "vdst1"
  },
  "@sessionName": "",
  "SrcPort": {
    "@portNID": "",
    "@portTitle": "",
    "@pvrEnable": "",
    "@id": ""
  }
}
}}
```

Example 4.29. View Request to add a connection from an existing Src to a new Dst with profileid

```
{ "View": {
  "@windowId": "",
  "@dstEntityId": "",
  "@name": "",
  "@srcEntityId": "src1",
  "@profileid": "profileId1",
  "@restartSession": "",
  "@sessionId": "recordSessionID",
  "@canvasId": "",
  "@id": "vstrm1",
  "DstPort": {
    "@portNID": "",
    "@portTitle": "srcpc1",
    "@id": "vdst1"
  },
  "@sessionName": "",
  "SrcPort": {
    "@portNID": "",
    "@portTitle": "",
    "@pvrEnable": "",
    "@id": ""
  }
}
}}
```

Example 4.30. View Request to add a connection from an existing Src to existing Dst

```
{ "View": {
  "@windowId": "",
  "@dstEntityId": "vdst1",
  "@name": "",
  "@srcEntityId": "src1",
  "@restartSession": false,
  "@sessionId": "recordSessionID",
  "@canvasId": "",
  "@id": "vstrm2",
  "DstPort": {
    "@portNID": "",
    "@portTitle": "",
    "@id": ""
  },
},
"@sessionName": "",
"SrcPort": {
  "@portNID": "",
  "@portTitle": "",
  "@pvrEnable": "",
  "@id": ""
}
}}
```

Example 4.31. View Request Response

The response to the view request will consist of the updated Session element

- *sessionId* : *id of session (Default UUID)* [Optional]
- *id* : *id of connection (Default UUID)* [Optional]
- *name* : *name of the Connection (Default conn_UUID)* [Optional]
- *sessionName* : *name of the session. Default is username and session start Time(srcuser1 Jun 25 00:15)* [Optional]
- *dstEntityId* : *Destination entity id in session* [Either entity id or portTitle Required]
- *srcEntityId* : *source entity id in session* [Either entity id or portTitle Required]
- *portTitle* : *Source/Destination resource Title (Matches to Entity name in the session doc)* [Either entity id or portTitle Required]
- *profileTitle* : *profile title to be used in session* [Optional]
- *profileNID* : *profile nid to be used in session* [Optional]
- *profileId* : *profile Id in session* [Optional]

```
{"Session": {
  "ProfileList": {"Profile": {
    "@name": "V2D_LAN",
    "@profileNID": "default.mediastreamprofilelist.mspl_elcb45a9-f433-40b2-8156-25",
    "@id": "3980fb57-5fd7-4398-be26-2ca9056b0d63",
    "@ffTracks": ""
  }},
  "@type": "LIVE",
  "@timelineId": "505f1b62-9a8e-4008-8c22-2d683ed40dfa.StreamTrack",
  "GrpFileList": "",
  "DirList": "",
  "ConnectionList": {"Connection": {
    "@streamURL": "v2d://10.1.5.169/port=6060?bandwidth=10240000?\r\n",
    "@name": "tx1",
    "@dstEntityId": "72a3d1d7-6aa3-481c-ae32-02e7a0a9929a",
    "@srcEntityId": "c5d9ac1f-fb26-4f35-9626-a82e00aad228",
    "@type": "VIEW",
    "@state": "STREAMING",
    "@id": "Conn_4A12955D-E857-4540-BC45-B951AE9F4A68",
    "ConnSeg": [
      {
        "@profileTitle": "V2D_LAN",
        "@errorCode": "",
```

Example 4.32. View Response (Continued)

```

    "@errorData": "",
    "@profileNID": "profileNID",
    "@dstResourceNID": "default.mediastreamrelayportresourcelist.\r\n",
    "@state": "STREAMING",
    "@srcResourceTitle": "tx1",
    "@srcResourceNID": "default.mediastreamsrcportresourcelist.\r\n",
    "@dstResourceTitle": "srcrelay1"
  },
  {
    "@profileTitle": "V2D_LAN",
    "@errorCode": "",
    "@errorData": "",
    "@profileNID": "default.mediastreamprofilelist.\r\n",
    "@dstResourceNID": "",
    "@state": "STREAMING",
    "@srcResourceTitle": "srcrelay1",
    "@srcResourceNID": "",
    "@dstResourceTitle": "padmajapc"
  }
]
}},
"EntityList": {"Entity": [
  {
    "@errorCode": "",
    "@streamType": "V2D",
    "@name": "tx1",
    "Port": {
      "@trickPlay": "",
      "@resourceTitle": "tx1",
      "@resourceNID": "default.mediastreamsrcportresourcelist.\r\n",
    },
    "@errorData": "",
    "@type": "SrcPort",
    "@gc": true,
    "@presence": "available",
    "@state": "",
    "@id": "c5d9ac1f-fb26-4f35-9626-a82e00aad228"
  },

```

mspl_b3

msspr1

Example 4.33. View Response (Continued)

```

        {
            "@errorCode": "",
            "@streamType": "MULTI",
            "@name": "pcl",
            "Port": {
                "@trickPlay": true,
                "@resourceTitle": "pcl",
                "@resourceNID": "default.mediastreamdstportresourcelist.\r\n
            },
            "@errorData": "",
            "@type": "DstPort",
            "@gc": true,
            "@presence": "available",
            "@state": "",
            "@id": "72a3d1d7-6aa3-481c-ae32-02e7a0a9929a"
        }
    ]},
    "Properties": {
        "@ownerUserJID": "srcuser1@localhost/pcl",
        "@dateCreated": 1277424909504,
        "@playlength": "",
        "@sessionConfigNID": ""
    },
    "Groups": { "Tag": {
        "@Name": "Location",
        "@Value": "Sunnyvale"
    } },
    "@name": "srcuser1 Jun 25 00:15",
    "TimelineList": { "Timeline": [
        {
            "Last": {
                "@requestedAction": "",
                "@WC": 1277424909504,
                "@TimeScale": 1,
                "@Offset": 0,
                "@TC": ""
            },
            "Start": {
                "@parentTimelineId": "",
                "@WC": 1277424909504,
                "@Offset": 0,
                "@TC": ""
            }
        },

```

msdpr

Example 4.34. View Response (Continued)

```
"@connGroupId": "",
"@sessionId": "Session_29BAD04A-DB32-466B-95EF-40D6729D6DFD",
"@id": "505f1b62-9a8e-4008-8c22-2d683ed40dfa.StreamTrack",
"@connId": ""
},
{
  "Last": {
    "@requestedAction": "",
    "@WC": 1277424909504,
    "@TimeScale": 1,
    "@Offset": 0,
    "@TC": ""
  },
  "Start": {
    "@parentTimelineId": "",
    "@WC": 1277424909504,
    "@Offset": 0,
    "@TC": ""
  },
  "@connGroupId": "",
  "@sessionId": "Session_29BAD04A-DB32-466B-95EF-40D6729D6DFD",
  "@id": "505f1b62-9a8e-4008-8c22-2d683ed40dfa.RecordTrack",
  "@connId": ""
}
]},
"@recordTimelineId": "505f1b62-9a8e-4008-8c22-2d683ed40dfa.RecordTrack",
"@gc": true,
"@id": "Session_29BAD04A-DB32-466B-95EF-40D6729D6DFD",
"@state": "Started"
}}
```

Playback

Used to setup connections between Recorded media and Playback destinations.

Use the SrcFile/GrpFile and DstPort to specify the resources to connect. Identify the file by specifying the resourceNID or the dir, filename and recording date Optional sessionId can be specified Use the srcEntityId or dstEntityId attribute specs in order to specify a connection to an existing src/dst. Use the canvasId and windowId to specify the canvas/window to be used for the connection.

Specify an optional Media Store Port to override the default based on the admin settings.

Specify a SyncMediaId shared among media that should be played back "in Sync".

Support for Playback by Title/fileDate NOT implemented.

Example 4.35. Basic Playback Request by Title

[Not Implemented]

Table 4.6. Basic Playback Request by Title

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{ "Playback": {  
  "SrcFile": {  
    "@fileTitle": "",  
    "@fileDate": "",  
    "@id": ""  
  },  
  "DstPort": {  
    "@portTitle": "",  
    "@id": ""  
  }  
}}
```

Example 4.36. Basic Playback Request by NID

A new session is always started. Returns an error if sessionId already exists. If no sessionId is specified a sessionId is generated by the system

```
{
  "Playback": {
    "@windowId": "",
    "@dstEntityId": "",
    "GrpFile": {
      "@fileNID": "",
      "@fileTitle": "",
      "@id": "grpId1"
    },
    "@srcEntityId": "",
    "@restartSession": "",
    "@sessionId": "playbackAPISessionID",
    "SrcFile": {
      "@fileNID": "nid1",
      "@fileTitle": "",
      "@grpFileId": "grpId1",
      "@id": ""
    },
    "@canvasId": "",
    "@id": "strmc1",
    "DstPort": {
      "@portNID": "",
      "@portTitle": "srcpcl",
      "@id": ""
    }
  }
}
```

Example 4.37. Basic Playback Request with file name/date spec

[Not Implemented]

```
{ "Playback": {
  "@windowId": "",
  "@connGroupId": "",
  "@sessionId": "",
  "SrcFile": {
    "@fileTitle": "",
    "@fileDate": "",
    "@id": ""
  },
  "@canvasId": "",
  "@id": "",
  "DstPort": {
    "@portNID": "",
    "@id": ""
  }
}
}}
```

Example 4.38. Advanced Playback Request including grpFileId and StorePort spec

[Not Implemented]

```
{ "Playback": {
  "@windowId": "",
  "@connGroupId": "",
  "@sessionId": "",
  "SrcFile": {
    "@fileNID": "",
    "@fileTitle": "",
    "StorePort": {
      "@portNID": "",
      "@id": ""
    },
    "@grpFileId": "",
    "@fileDate": "",
    "@id": ""
  },
  "@canvasId": "",
  "@id": "",
  "DstPort": {
    "@portNID": "",
    "@id": ""
  }
}
}}
```

Example 4.39. Basic Playback Request by group file NID

A new session is always started with connections for all the clip files under the groupfile specified by the NID. Any existing session with the given sessionId is terminated. If no sessionId is specified a sessionId is generated by the system

```
{ "Playback": {
  "@windowId": "",
  "@dstEntityId": "",
  "GrpFile": {
    "@fileNID": "default.mediastreamfileresourcelist.msfrl_3ed1f399-5026-43e8-9a8",
    "@fileTitle": "",
    "@id": "grpId1"
  },
  "@srcEntityId": "",
  "@restartSession": "",
  "@sessionId": "playbackAPISessionID",
  "SrcFile": {
    "@fileNID": "",
    "@fileTitle": "",
    "@grpFileId": "grpId1",
    "@id": ""
  },
  "@canvasId": "",
  "@id": "strmc1",
  "DstPort": {
    "@portNID": "",
    "@portTitle": "srcpc1",
    "@id": ""
  }
}
}}
```

Example 4.40. Basic Playback Request by srcEntityId

Start a playback connection to an existing playback source in an existing session

```
{ "Playback": {
  "@windowId": "",
  "@dstEntityId": "",
  "GrpFile": {
    "@fileNID": "",
    "@fileTitle": "",
    "@id": "grpId1"
  },
  "@srcEntityId": "recordd1",
  "@restartSession": "",
  "@sessionId": "playbackSessionID",
  "SrcFile": {
    "@fileNID": "",
    "@fileTitle": "",
    "@grpFileId": "grpId1",
    "@id": ""
  },
  "@canvasId": "",
  "@id": "vstrml",
  "DstPort": {
    "@portNID": "",
    "@portTitle": "srcpcl",
    "@id": "vdst1"
  }
}
}}
```

Example 4.41. Basic Playback Request by srcEntityId and dstEntityId

Start a playback connection to an existing playback source in an existing session

```
{
  "Playback": {
    "@windowId": "",
    "@dstEntityId": "vdst1",
    "GrpFile": {
      "@fileNID": "",
      "@fileTitle": "",
      "@id": "grpId1"
    },
    "@srcEntityId": "recordd1",
    "@restartSession": "",
    "@sessionId": "playbackSessionID",
    "SrcFile": {
      "@fileNID": "",
      "@fileTitle": "",
      "@grpFileId": "grpId1",
      "@id": ""
    },
    "@canvasId": "",
    "@id": "vstrm2",
    "DstPort": {
      "@portNID": "",
      "@portTitle": "",
      "@id": ""
    }
  }
}
```

Example 4.42. Playback Request Response

The response is similar to the view request response will consist of the updated Session element

Play

This is the equivalent of issuing "Resume" with new speed.

timeScale	Playback Speed. Supported Speed 1, 2, 5, 10, 60
nowOffset	An Offset relative to NOW.
startOffset	An Offset relative to the START Time of the current Scope.
liveOffset	An Offset relative to the LIVE Time of the current Scope.
mediaTime	A Time reference to the TC attached to the media in the Media File
mediaOffset	An Offset relative to the start of the Media File. This provides a sequential index into a Media File irrespective of the TCs of the media in it.

mediaId	The ID of the media associated with the mediaOffset/mediaTime. The session/connGrp/conn offset is found using the mediaId + mediaOffset/mediaTime This is used to resolve ambiguity when there is more than one media in the scope
mediaConnId	The connId associated with the mediaOffset/mediaTime. The session/connGrp/conn offset is found using the connId + mediaOffset/mediaTime This is used to resolve ambiguity when the same media is in more than one connection in the scope
bookmarkId	The ID of the bookmark associated with the mediaOffset/mediaTime. The session/connGrp/conn offset is found using the bookmarkId + mediaOffset/mediaTime This is used to resolve ambiguity when there is more than one media in the scope

Example 4.43. Play Request

Table 4.7. Play Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{ "Play": {
  "@timeScale": " ",
  "@sessionId": " "
}}
```

Example 4.44. Play Response

```
{ "Timeline": {
  "Last": {
    "@requestedAction": "ResumeFrom",
    "@TimeScale": 1,
    "@WC": 1252016160137,
    "@Offset": 87627,
    "@TC": ""
  },
  "Start": {
    "@parentTimelineId": "",
    "@WC": 1252016072510,
    "@Offset": 0,
    "@TC": ""
  },
  "@connGroupId": "",
  "@sessionId": "xyz123",
  "@id": "",
  "@connId": ""
}}
```

Example 4.45. Play Request w/ nowOffset

```
    {"Play": {
"@nowOffset": "",
"@sessionId": ""
}}
```

Example 4.46. Play Request w/ startOffset

```
    {"Play": {
"@nowOffset": "",
"@sessionId": ""
}}
```


Example 4.47. Play Request w/ liveOffset

```
        {"Play": {
"@liveOffset": " ",
"@sessionId": " "
}}
```

Example 4.48. Play Request w/ mediaOffset

```
        {"Play": {
"@mediaOffset": " ",
"@mediaId": " ",
"@mediaConnId": " ",
"@sessionId": " "
}}
```

Example 4.49. Play Request w/ mediaTime

```
        {"Play": {
"@mediaTime": " ",
"@mediaId": " ",
"@mediaConnId": " ",
"@sessionId": " "
}}
```

Example 4.50. Play Request w/ Bookmark

```
        {"Play": {
"@bookmarkId": " ",
"@sessionId": " ",
mediaConnId=" "
}}
```

Record

Used to setup connections for Recording media.

Use the SrcPort and DstFile elements to specify the resources to connect. Identify the file by specifying the resourceNID or the dir, filename and recording date
Specify an optional Media Store Port to override the default based on the admin settings.
Specify a SyncMediaId shared among media that should be recorded "in Sync".

- *sessionId* : *id of session (Default UUID)* [Optional]
- *id* : *id of connection (Default UUID)* [Optional]
- *name* : *name of the Connection (Default conn_UUID)* [Optional]
- *sessionName* : *name of the session. Default is username and session start Time(srcuser1 Jun 25 00:15)* [Optional]
- *srcEntityId* : *source entity id in session* [Either entity id or portTitle Required]
- *portTitle* : *Source resource Title (Matches to Entity name in the session doc)* [Either entity id or portTitle Required]
- *DstFile id* : *id of rec entity* [Either entity id or portTitle Required]
- *DstFile fileTitle* : *recording file title* [Required if key=Title or new recording otherwise optional. [Does not allow XML restricted special characters]]
- *DstFile Key* : *will be used to find out file existed or not. Values could be Title/MetaData* [optional]
- *DstFile action* : *what actions we can perform on Append/Overwrite(Default is Error on Exist if don't specify any thing) (Overwritten[not implemented])* [optional]
- *DstFile keyMetaDataTagName* : *used along with key attribute. If key=Metadata then have to specify keyMetaDataTagName* [Required if key=MetaData otherwise optional]
- *DstFile MetaData element* [optional]

Example 4.51. Basic Record Request

Table 4.8. Basic Record Request

Request	HTTP POST
Accept(No spaces or newlines)	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session

Authorization	Basic (Required)
---------------	------------------

```

        {"Record": {
"@name": "",
"DstFile": {
  "MetaData": {"Tag": [
    {
      "content": 1299792256,
      "@Name": "ID"
    },
    {
      "content": "BasicRecording",
      "@Name": "TestCase"
    }
  ]},
  "@fileTitle": "RecordingTitle",
  "@id": ""
},
"@sessionId": "",
"@id": "",
"@sessionName": "",
"SrcPort": {
  "@portNID": "",
  "@portTitle": "tx1",
  "@id": ""
}
}}

```

E.g. Record a file named Recording1 from SrcPort tx1

```

{"Record": {
  "DstFile": {
    "@fileTitle": "Recording1",
  },
  "SrcPort": {
    "@portTitle": "tx1",
  }
}}

```

Example 4.52. Basic Record Request with file name/dir spec

```
        {"Record": {
"@name": " ",
"DstFile": {
  "MetaData": {"Tag": [
    {
      "content": 1299792256,
      "@Name": "ID"
    },
    {
      "content": "BasicRecordingWithDirNID",
      "@Name": "TestCase"
    }
  ]},
"@fileTitle": "RecordingTitle",
"@id": " ",
"@dirNID": "dirNID1"
},
"@sessionId": " ",
"@id": " ",
"@sessionName": " ",
"SrcPort": {
  "@portNID": " ",
  "@portTitle": "tx1",
  "@id": " "
}
}}
```

Example 4.53. Record from an existing Src to file

```
        {"Record": {
"@name": " ",
"DstFile": {
  "MetaData": {"Tag": [
    {
      "content": 1299792256,
      "@Name": "ID"
    },
    {
      "content": "RecordExistingSource",
      "@Name": "TestCase"
    }
  ]},
"@fileTitle": "RecordingTitle",
"@id": " ",
"@dirNID": " "
},
"@srcEntityId": "srcentityId1",
"@sessionId": "id1",
"@id": "conn1"
}}
```

Example 4.54. Append to Recording Based on Title

```
        {"Record": {
"@name": " ",
"DstFile": {
  "@action": "Append",
  "@key": "Title",
  "@fileTitle": "fileTitle1",
  "@id": " "
},
"@sessionId": " ",
"@id": " ",
"SrcPort": {
  "@portNID": " ",
  "@portTitle": "tx1",
  "@id": " "
}
}}
```

Example 4.55. Append to Recording Based on MetaData

```
        {"Record": {
"@name": " ",
"DstFile": {
  "@action": "Append",
  "@key": "MetaData",
  "MetaData": {"Tag": [
    {
      "content": 1299792256,
      "@Name": "ID"
    },
    {
      "content": "AppendToRecording",
      "@Name": "TestCase"
    }
  ]},
  "@keyMetaDataTagName": "ID",
  "@fileTitle": "fileTitle1",
  "@id": " "
},
"@srcEntityId": " ",
"@sessionId": " ",
"@id": " ",
"SrcPort": {
  "@portNID": " ",
  "@portTitle": "tx1",
  "@id": " "
}
}}
```

Example 4.56. Record Request with Profile NID

```
        {"Record": {
"@name": "",
"DstFile": {
  "MetaData": {"Tag": [
    {
      "content": 1299792256,
      "@Name": "ID"
    },
    {
      "content": "BasicRecordingWithDirNID",
      "@Name": "TestCase"
    }
  ]},
"@fileTitle": "RecordingTitle",
"@id": "",
"@dirNID": "dirNID1"
},
"Profile": {
  "@profileTitle": "",
  "@profileNID": "NID1",
  "@id": ""
},
"@sessionId": "",
"@id": "",
"@sessionName": "",
"SrcPort": {
  "@portNID": "",
  "@portTitle": "tx1",
  "@id": ""
}
}}
```

Example 4.57. Record Request with Profile Title

```
        {"Record": {
"@name": "",
"DstFile": {
  "MetaData": {"Tag": [
    {
      "content": 1299792256,
      "@Name": "ID"
    },
    {
      "content": "BasicRecordingWithDirNID",
      "@Name": "TestCase"
    }
  ]},
"@fileTitle": "RecordingTitle",
"@id": "",
"@dirNID": "dirNID1"
},
"Profile": {
  "@profileTitle": "V2D_LAN",
  "@profileNID": "",
  "@id": ""
},
"@sessionId": "",
"@id": "",
"@sessionName": "",
"SrcPort": {
  "@portNID": "",
  "@portTitle": "tx1",
  "@id": ""
}
}}
```


Example 4.58. Record from an existing Src with profile id

```
        {"Record": {
"@name": "",
"DstFile": {
  "MetaData": {"Tag": [
    {
      "content": 1299792256,
      "@Name": "ID"
    },
    {
      "content": "BasicRecordingWithDirNID",
      "@Name": "TestCase"
    }
  ]},
"@fileTitle": "RecordingTitle",
"@id": "",
"@dirNID": "dirNID1"
},
"@profileId": "profileID",
"Profile": {
  "@profileTitle": "",
  "@profileNID": "",
  "@id": ""
},
"@sessionId": "",
"@id": "",
"@sessionName": "",
"SrcPort": {
  "@portNID": "",
  "@portTitle": "tx1",
  "@id": ""
}
}}
```

Example 4.59. Advanced Record Request including grpFileId and StorePort spec

[Not Implemented]

```
{ "Record": {  
  "DstFile": {  
    "@dirTitle": "",  
    "@fileTitle": "",  
    "StorePort": {  
      "@portNID": "",  
      "@id": ""  
    },  
    "@grpFileId": "",  
    "@id": "",  
    "@dirNID": ""  
  },  
  "@connGroupId": "",  
  "@sessionId": "",  
  "@id": "",  
  "SrcPort": {  
    "@portNID": "",  
    "@id": ""  
  }  
}  
}}
```

Example 4.60. Record Request Response

```
{ "Session": {
  "ProfileList": { "Profile": {
    "@name": "V2D_LAN",
    "@profileNID": "default.mediastreamprofilelist.mspl_elcb45a9-f433-40b2-8156-",
    "@id": "3980fb57-5fd7-4398-be26-2ca9056b0d63"
  } },
  "@type": "LIVE",
  "@timelineId": "c29ccf26-74c4-4e5a-9d3e-f2818e54cd1d.StreamTrack",
  "GrpFileList": "",
  "DirList": "",
  "ConnectionList": { "Connection": {
    "@windowId": "set_file_metada_test_1299792256",
    "@name": "set_file_metada_test_1299792256",
    "@dstEntityId": "47bdcfe0-b7a1-4e73-a1b9-01318f99e654",
    "@srcEntityId": "021cf69d-b6e4-4f5e-88e2-56930830d243",
    "@type": "RECORD",
    "@state": "WAITING_FOR_SOURCE-DEST_CONTACTED",
    "@canvasId": "set_file_metada_test_1299792256",
    "@id": "set_file_metada_test_1299792256"
  } },
  "EntityList": { "Entity": [
    {
      "@errorCode": "",
      "@name": "tx1",
      "Port": {
        "@trickPlay": "",
        "@resourceTitle": "tx1",
        "@resourceNID": "default.mediastreamsrcportresourceList.mssprl_e22023ce-",
      },
    },
  ]
}
```

Example 4.61. Record Request Response (Continued)

```
"@errorData": "",
"@type": "SrcPort",
"@gc": false,
"@presence": "available",
"@state": "",
"@id": "021cf69d-b6e4-4f5e-88e2-56930830d243"
},
{
  "@errorCode": "",
  "@name": "set_file_metada_test_1299792256",
  "@errorData": "",
  "@type": "RecFile",
  "File": {
    "@playLength": 0,
    "@clipId": "47bdcfe0-b7a1-4e73-a1b9-01318f99e654",
    "@action": "",
    "@key": "",
    "MetaData": {"Tag": [
      {
        "content": 1299792256,
        "@Name": "ID"
      },
      {
        "content": "AddMediaStreamFile",
        "@Name": "TestCase"
      }
    ]},
    "@dirId": "",
    "@grpFileId": "",
    "@resourceTitle": "set_file_metada_test_1299792256",
    "@mediaGroupId": "",
    "@keyMetaDataKeyName": "",
    "@resourceNID": "default.mediastreamfileresourcelist.msfrl_725d55ff-30ef"
  },
  "@gc": false,
  "@presence": "available",
  "@state": "",
  "@id": "47bdcfe0-b7a1-4e73-a1b9-01318f99e654"
}
]}}
```

Example 4.62. Record Request Response (Continued)

```

        "Properties": {
          "@dateCreated": 1299792256254,
          "@ownerUserJID": "srcuser1@localhost/srcpc1",
          "@playlength": "",
          "@sessionConfigNID": ""
        },
        "Groups": { "Tag": {
          "@Name": "Location",
          "@Value": "Sunnyvale"
        } },
        "@name": "srcuser1 Mar 10 21:24",
        "TimelineList": "",
        "@recordTimelineId": "c29ccf26-74c4-4e5a-9d3e-f2818e54cd1d.RecordTrack",
        "@gc": true,
        "@id": "set_file_metada_test_1299792256",
        "@state": "Started"
      }
    }
  }
}

```

SwitchSrc

Used to switch the src for an existing connections.

Use the SrcEntityId to specify the source to switch to.

Example 4.63. SwitchSrc Request**Table 4.9. SwitchSrc Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```

    { "SwitchSrc": {
      "@sessionId": "recordSessionID",
      "@newSrcEntityId": "src1",
      "@connId": "strmc2"
    } }
  }
}

```

Example 4.64. SwitchSrc Response

```
{ "Connection": {  
  "@dstEntityId": "myDst1234",  
  "@srcEntityId": "src1",  
  "@id": "strmc2"  
}}
```

ResumeRecord

Example 4.65. ResumeRecord Request**Table 4.10. ResumeRecord Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{ "ResumeRecord": { "@sessionId": "xyz123" } }
```

Example 4.66. ResumeRecord Response

```
{
  "Timeline": {
    "Last": {
      "@requestedAction": "Resume",
      "@TimeScale": 1,
      "@WC": 1252016479279,
      "@Offset": 406769,
      "@TC": ""
    },
    "Start": {
      "@parentTimelineId": "",
      "@WC": 1252016072510,
      "@Offset": 0,
      "@TC": ""
    },
    "@connGroupId": "",
    "@sessionId": "xyz123",
    "@id": "",
    "@connId": ""
  }
}
```

SkipBack

The "skipOffset" is always a "nowOffset". This is the equivalent of issuing "Resume" with a negative "nowOffset"

Example 4.67. SkipBack Request**Table 4.11. SkipBack Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{
  "SkipBack": {
    "@timeScale": "",
    "@sessionId": "xyz123",
    "@skipOffset": 2000
  }
}
```

Example 4.68. SkipBack Response

```
{
  "Timeline": {
    "Last": {
      "@requestedAction": "SkipBack",
      "@TimeScale": 1,
      "@WC": 1252016400432,
      "@Offset": 325922,
      "@TC": ""
    },
    "Start": {
      "@parentTimelineId": "",
      "@WC": 1252016072510,
      "@Offset": 0,
      "@TC": ""
    },
    "@connGroupId": "",
    "@sessionId": "xyz123",
    "@id": "",
    "@connId": ""
  }
}
```

SkipFwd

The "skipOffset" is always a "nowOffset". This is the equivalent of issuing "Resume" with a positive "nowOffset"

Example 4.69. SkipFwd Request**Table 4.12. SkipFwd Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{
  "SkipFwd": {
    "@timeScale": "",
    "@sessionId": "xyz123",
    "@skipOffset": 1000
  }
}
```


Example 4.70. SkipFwd Response

```
{ "Timeline": {
  "Last": {
    "@requestedAction": "SkipFwd",
    "@TimeScale": 1,
    "@WC": 1252016400432,
    "@Offset": 325922,
    "@TC": ""
  },
  "Start": {
    "@parentTimelineId": "",
    "@WC": 1252016072510,
    "@Offset": 0,
    "@TC": ""
  },
  "@connGroupId": "",
  "@sessionId": "xyz123",
  "@id": "",
  "@connId": ""
}}
```

PauseRecord

Example 4.71. PauseRecord Request**Table 4.13. PauseRecord Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{ "PauseRecord": { "@sessionId": "xyz123" } }
```

Example 4.72. PauseRecord Response

```
{ "Timeline": {
  "Last": {
    "@requestedAction": "Pause",
    "@TimeScale": 0,
    "@WC": 1252016452472,
    "@Offset": 379962,
    "@TC": ""
  },
  "Start": {
    "@parentTimelineId": "",
    "@WC": 1252016072510,
    "@Offset": 0,
    "@TC": ""
  },
  "@connGroupId": "",
  "@sessionId": "xyz123",
  "@id": "",
  "@connId": ""
}}
```

Pause

Example 4.73. Pause Request**Table 4.14. Pause Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{"Pause": {"@sessionId": "xyz123"}}
```

Example 4.74. Pause Response

```
{
  "Timeline": {
    "Last": {
      "@requestedAction": "Pause",
      "@TimeScale": 0,
      "@WC": 1252016119868,
      "@Offset": 47358,
      "@TC": ""
    },
    "Start": {
      "@parentTimelineId": "",
      "@WC": 1252016072510,
      "@Offset": 0,
      "@TC": ""
    },
    "@connGroupId": "",
    "@sessionId": "xyz123",
    "@id": "",
    "@connId": ""
  }
}
```

Catchup

This is the equivalent of issuing "Resume" with a liveOffset=0

Example 4.75. Catchup Request**Table 4.15. Catchup Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{
  "Catchup": {
    "@sessionId": ""
  }
}
```

Example 4.76. Catchup Response

```
{ "Timeline": {
  "Last": {
    "@requestedAction": "Catchup",
    "@TimeScale": 1,
    "@WC": 1252016160137,
    "@Offset": 87627,
    "@TC": ""
  },
  "Start": {
    "@parentTimelineId": "",
    "@WC": 1252016072510,
    "@Offset": 0,
    "@TC": ""
  },
  "@connGroupId": "",
  "@sessionId": "xyz123",
  "@id": "",
  "@connId": ""
}}
```

Session Participant API

API calls for Session Participants

Invite

[Not Implemented]

Issue an Invite to explicitly invite another user to a session

Example 4.77. Simple Invite by userId or jid

Table 4.16. Simple Invite by userId or jid

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session

Authorization	Basic (Required)
---------------	------------------

```
{ "Invite": {
  "@sessionId": "",
  "@jid": "",
  "@id": "",
  "@userId": ""
}}
```

Example 4.78. Invite by userId(who) and/or deviceId(from where)

```
{ "Invite": {
  "@sessionId": "",
  "@id": "",
  "@userId": "",
  "@deviceId": ""
}}
```

Join

In response to the Join - the system will send back an "MUC invite" to an MUC room. The client should respond to that by doing a standard "XMPP MultiUserChat Join" to enter the MUC room.

- Join is once per session with or without invite.
- An invite is required to join "Invitation Only" Sessions (the default)
- Sessions that have been "Published" can be joined without an explicit "Invite"

Example 4.79. Join without invite Request

Table 4.17. Join without invite Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{"Join": {"@sessionId": ""}}
```

Example 4.80. Join without invite Response

```
{"Session": ""}
```

Example 4.81. Join without invite Response - MUC Room Invite Message

```
{"Session": ""}
```

Example 4.82. Join with invite

```
{"Join": {  
  "@inviteId": "",  
  "@sessionId": ""  
}}
```

Kick

[Not Implemented]

Issue a Kick to explicitly force a user to leave a session

Example 4.83. Kick**Table 4.18. Kick**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session

Authorization	Basic (Required)
---------------	------------------

```
{ "Kick": {  
  "@userName": "",  
  "@sessionId": "",  
  "@jid": "",  
  "@id": "",  
  "@userId": "",  
  "@deviceId": ""  
}}
```

Bookmark Operations

API calls for Bookmarks

GetBookmark

Example 4.84. GetBookmark Request

Table 4.19. GetBookmark Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

Example 4.85. GetBookmark Request

```
{ "GetBookmark": {  
  "@whereSessionInfo": "",  
  "@whereTCRange": "",  
  "@sortBy": "ascending/descending",  
  "@whereNID": "",  
  "@sortOrder": "",  
  "@startFrom": "",  
  "@countToFetch": "",  
  "@whereFileNID": "",  
  "@whereTextSubStr": "",  
  "@selectFormat": "BASIC/EXTENDED"  
}}
```

- *Either whereSessionInfo or whereFileNID Required*

*whereSessionInfo is key:value pairs e.g. "@whereSessionInfo":
"sessionId:,connGroupId:,connId:,entityId"*

whereFileNID is the NID of the file which contains bookmarks.

Example 4.86. GetBookmark Response - BASIC Format

```
{ "GetBookmark": {  
  "@total": "",  
  "Bookmark": [  
    {  
      "@NID": "",  
      "@offset": "",  
      "@fileNID": "",  
      "@dateCreated": "",  
      "@title": "",  
      "@owner": "",  
      "@TC": ""  
    },  
    {  
      "@NID": "",  
      "@offset": "",  
      "@fileNID": "",  
      "@dateCreated": "",  
      "@title": "",  
      "@owner": "",  
      "@TC": ""  
    },  
    {  
      "@NID": "",  
      "@offset": "",  
      "@fileNID": "",  
      "@dateCreated": "",  
      "@title": "",  
      "@owner": "",  
      "@TC": ""  
    }  
  ],  
  "@from": ""  
}}
```

Example 4.87. GetBookmark Response - EXTENDED Format

```

{ "GetBookmark": {
  "@total": " ",
  "Bookmark": {
    "@NID": " ",
    "Description": " ",
    "@offset": " ",
    "MetaData": { "Tag": { "@Name": " " } },
    "@fileNID": " ",
    "@dateCreated": " ",
    "@title": " ",
    "@owner": " ",
    "@TC": " "
  },
  "@from": " "
}}

```

AddBookmark

The location of the bookmark is derived by the App Server based on the "Time" of the specified Connection (or ConnGroup). All the fields in the RequestXML are optional. If an 'id' is not specified - it will be generated by the system. Bookmarks can be added without a title or comment and then updated with all the info later.

Example 4.88. AddBookmark Request**Table 4.20. AddBookmark Request**

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session

Authorization	Basic (Required)
---------------	------------------

```
{ "AddBookmark": {
  "@whereSessionInfo": "",
  "Bookmark": {
    "Description": "",
    "@offset": "",
    "MetaData": { "Tag": { "@Name": "" } },
    "@title": "",
    "@TC": ""
  },
  "@whereFileNID": ""
}}
```

- *Either whereSessionInfo or whereFileNID Required*

whereSessionInfo is key:value pairs e.g. "@whereSessionInfo": "sessionId:,connGroupId:,connId:,entityId"

whereFileNID is the NID of the file to add bookmark to.

Example 4.89. AddBookmark Response

```
{ "AddBookmark": {
  "@NID": "",
  "@title": "",
  "Bookmark": {
    "Description": "",
    "@offset": "",
    "MetaData": { "Tag": { "@Name": "" } },
    "@fileNID": "",
    "@dateCreated": "",
    "@owner": "",
    "@TC": ""
  }
}}
```

DeleteBookmark

Example 4.90. DeleteBookmark Request

Table 4.21. DeleteBookmark Request

Request	HTTP POST
---------	-----------

Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{ "DeleteBookmark": {
  "@whereSessionInfo": "",
  "@whereTCRange": "",
  "@whereDateRange": "",
  "@whereNID": "",
  "@whereFileNID": ""
}}
```

- *whereNID* or *whereSessionInfo* or *whereFileNID* Required

whereNID is the NID of the bookmark to delete.

whereTCRange or *whereDateRange* format: "From:2014-11-17 00:00:00.0000,To:2014-11-19 15:00:00.0000"

Example 4.91. DeleteBookmark Response

```
{ "DeleteBookmark": {
  "@total": "",
  "Bookmark": [
    { "@NID": "" },
    { "@NID": "" },
    { "@NID": "" }
  ]
}}
```

NewBookmark

Return new bookmark for given session or file nid by filling the time code and offset values. (Note: this call don't add bookmark in database. To add bookmark in database user has to use AddBookmark along with this). []

Example 4.92. NewBookmark Request

Table 4.22. NewBookmark Request

Request	HTTP POST
---------	-----------

Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{ "NewBookmark": {
  "@whereSessionInfo": "sessionId:,connGroupId:,connId:,entityId",
  "@whereFileNID": ""
}}
```

- *Either whereSessionInfo or whereFileNID Required*

Example 4.93. NewBookmark Response

```
{ "NewBookmark": { "Bookmark": {
  "Description": "",
  "@offset": "",
  "MetaData": { "Tag": { "@Name": "" } },
  "@title": "",
  "@TC": ""
}}}
```

UpdateBookmark

Example 4.94. UpdateBookmark Request

Table 4.23. UpdateBookmark Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500
Query Parameters	None
Resource Path	/Session

Authorization	Basic (Required)
---------------	------------------

```
{ "UpdateBookmark": {
  "@whereNID": "",
  "Bookmark": {
    "Description": "",
    "@offset": "",
    "MetaData": { "Tag": { "@Name": "" } },
    "@title": "",
    "@TC": ""
  }
}}
```

- *whereNID* Required

Example 4.95. UpdateBookmark Response

```
{ "UpdateBookmark": { "Bookmark": {
  "@NID": "",
  "Description": "",
  "@offset": "",
  "MetaData": { "Tag": { "@Name": "" } },
  "@fileNID": "",
  "@dateCreated": "",
  "@title": "",
  "@owner": "",
  "@TC": ""
}}}
```

Authorization

External Authorization

Authorize Request

Auth user can reply back with his decision allow/deny/revoke using Authorize request

Example 4.96. Authorize Request

Table 4.24. Authorize Request

Request	HTTP POST
Accept	"application/json"
Status	200, 401, 500

Query Parameters	None
Resource Path	/Session
Authorization	Basic (Required)

```
{ "Authorize": {  
  "@action": "ALLOW/DENY/REVOKE",  
  "@sessionId": "",  
  "@reason": "",  
  "@connId": ""  
}}
```

Example 4.97. Authorize Response

```
{ "Authorize": {  
  "AuthConnectionInfo": ""  
  <!-- Check AuthConnectionInfo element in ConnectionUpdate -->  
}}
```

Chapter 5. Session Notifications

EntityPresence Notification

Every time an entity joins or leaves the room a presence update is sent into the room

Example 5.1. EntityPresence Notification

```
{ "EntityPresence": {  
  "@entityId": "",  
  "@value": "",  
  "@sessionId": "",  
  "@jid": ""  
}}
```

ConnState Notification

Every time a connection state changes - a notification is sent into the room. If value is Error - then code and msg are valid The list of possible connections states are mentioned below

Example 5.2. ConnState Notification: CONNECTED

```
{ "ConnState": {  
  "@value": "<see below>",  
  "@sessionId": "",  
  "Timeline": "",  
  "@connId": ""  
}}
```

The list of possible connections states are mentioned below
[desc not present in the notification]

The connection states are also reflected in the Connection
element within the Session

```
{ "ConnStateList": { "ConnState": [  
  {  
    "@desc": "Stream has been created",  
    "@value": "CREATED"  
  },  
  {  
    "@desc": "Waiting for authorization",  
    "@value": "AUTHORIZING"  
  },  
  {  
    "@desc": "Waiting for the source/dest resource to acquire a contactJID",  
    "@value": "CHECKING_AVAILABILITY"  
  },  
  {  
    "@desc": "Waiting for destination to join the room",  
    "@value": "WAITING_FOR_DEST"  
  },  
  {  
    "@desc": "Destination notified of stream",  
    "@value": "WAITING_FOR_SOURCE-DEST_CONTACTED"  
  },  
  {  
    "@desc": "Destination waiting for source",  
    "@value": "WAITING_FOR_SOURCE-DEST_ACKNOWLEDGED"  
  },  
  {  
    "@desc": "Destination waiting for streamURL",  
    "@value": "WAITING_FOR_STREAMURL"  
  },  
]}
```

Example 5.3. ConnState Notification: CONNECTED(Continued)

```
    {
      "@desc": "Destination waiting for stream started notification",
      "@value": "WAITING_FOR_STREAMSTART"
    },
    {
      "@desc": "Stream started",
      "@value": "STREAMING"
    },
    {
      "@desc": "Stream paused",
      "@value": "STREAM_PAUSED"
    },
    {
      "@desc": "Authorization denied",
      "@value": "DENIED_XAUTH"
    },
    {
      "@desc": "Authorization revoked",
      "@value": "REVOKED_XAUTH"
    },
    {
      "@desc": "End of Media",
      "@value": "END_OF_MEDIA"
    },
    {
      "@desc": "Error while setup of stream",
      "@value": "SETUP_ERROR"
    },
    {
      "@desc": "Error when starting the stream",
      "@value": "START_ERROR"
    },
    {
      "@desc": "Source setup error",
      "@value": "SOURCE_ERROR"
    },
    {
      "@desc": "Media error",
      "@state": "MEDIA_ERROR"
    }
  ]}]}
```

The Timeline element is same as the Session Timeline mentioned earlier,
but now specific to the scope of the Connection

Example 5.4. ConnState Notification: Error

```
{ "ConnState": {  
  "@errorCode": "NNNN",  
  "@errorData": "Error Description",  
  "@value": "ERROR",  
  "@sessionId": "",  
  "@connId": ""  
}}
```

SessionState Notification

Every time a Session state changes - a notification is sent into the room. If value is Error - then code and msg are valid The possible values in the SessionState Notification are CREATED and DELETED

Example 5.5. SessionState Notification

```
{ "SessionState": {  
  "@value": "",  
  "@sessionId": ""  
}}
```

Timeline Notification

A Timeline is a collection of 3 attributes that serves as the basic unit for sharing Time/Offset information among Entities. It allows for Time/Offset exchange without being subject to the delays in message transmission/processing. Using it, one Entity can derive/track the Current Offset of another entity (to the extent that their WCs are in Sync).

Timelines exist Hierarchically for:

- the session
- all connection groups
- all connections
- all entities.

A Basic Timeline correlates:

- lastWC = the WC when the last action that affected this Timeline happened
- lastOffset = the Offset of this Entity w.r.t. its START at that time
- TimeScale = the rate at which the Offset has been changing since then - reflects the lastAction: eg: Play = 1, Stop = 0, Rewind = -1, 15xFF = 15

To track fully Time/Offset, the Timeline includes Info about the Start of the Entity.

- StartWC = The WC at the time this Entity was Started. This is used to derive the Entity's LIVE Time at any moment.
- StartOffset = The difference between the 0 offset of this Entity and the 0 Offset of it's current Parent Entity This is used to translate a parent Entity Offset to a child Entity Offset.
- PVR actions essentially specify a new "target" Timeline for the scope they are issued at.
- These then result in timeline change requests for entities that make up that scope
- When the entites achieve this timeline, they send timeline update notifications into the session.
- If all goes well, these will match the requested timeline.
- The other event that causes Timeline Notifications is when Connection grouping/hierarchy changes.

If the difference between the Start and Last WC is equal to the difference between the Start and Last Offset then you are "Live/CaughtUp"

Example 5.6. Timeline Notification

```
{ "Timeline": {
  "Last": {
    "@requestedAction": "Pause",
    "@TimeScale": 0,
    "@WC": 7444674,
    "@Offset": 9678687,
    "@TC": ""
  },
  "Start": {
    "@parentTimelineId": "",
    "@WC": 2341234,
    "@Offset": 3456,
    "@TC": ""
  },
  "@connGroupId": "",
  "@sessionId": "",
  "@id": "",
  "@connId": ""
}}
```

Appendix A. Data Model

RX Display Config/Status Parameter Listing

- *Background enable*="true/false" [Required]
- *Background zIndex*=" " [Required]
- *Background content*="0xRRGGBB/LOCALVIDEO" [Required]
- *Window id*=" " [Required]
- *Window enable*="true/false" [Required]
- *Window zIndex*=" " [Required]
- *Window content*="SPLASH/REMOTEVIDEO" [Required]
- *vpos*="ALIGNTOP/ALIGNMIDDLE/ALIGNBOTTOM/NNNpx" [Required]
- *hpos*="ALIGNLEFT/ALIGNMIDDLE/ALIGNRIGHT/NNNpx" [Required]
- *hres*="NNNpx" [Required]
- *vres*="NNNpx" [Required]
- *aspect*="SourceWidth/SourceHeight/Auto/None" [Required]
- *transparency*="NNNpx" [Required]

RX DisplayConfig

```
{ "Canvas": {  
  "Background": {  
    "@content": "LOCALVIDEO",  
    "@zIndex": 0,  
    "@enable": true  
  },  
  "Window": {  
    "@content": "REMOTEVIDEO",  
    "@zIndex": 1,  
    "Geometry": {  
      "@transparency": "20%",  
      "@hres": "75%",  
      "@vpos": "ALIGNTOP",  
      "@aspect": "Auto",  
      "@vres": "75%",  
      "@hpos": "ALIGNLEFT"  
    },  
    "@id": "id",  
    "@enable": true  
  }  
}
```

```
}}
```

RX DisplayStatus

```
{ "Canvas": {
  "@hres": "1280px",
  "Display": {
    "@hres": "1280px",
    "@lut": "15 [VESA 1280x1024@60 Hz]",
    "@vres": "1024px",
    "@id": ""
  },
  "@vres": "1024px",
  "Background": {
    "@content": "LOCALVIDEO",
    "@zIndex": 0,
    "@enable": true
  },
  "Window": {
    "Source": {
      "@hres": "960px",
      "@hscale": "75%",
      "@lut": "Server Splash Screen 1280x1024@60.0",
      "@vres": "768px",
      "@vscale": "75%"
    },
    "@content": "REMOTEVIDEO",
    "@zIndex": 1,
    "Geometry": {
      "@transparency": "20%",
      "@hres": "960px",
      "@vpos": "0px",
      "@vres": "768px",
      "@hpos": "0px"
    },
    "@id": "",
    "@enable": true
  }
}
```

Remote Operated Client Display Config/Status Parameter Listing

- *Background enable="true/false"* [Required]

- *Background* *zIndex*="" [Required]
- *Background* *content*='OS/Image/Session' [Required]
- *ImageContent* *imageUrl*="" [Required]
- *ImageContent* *color*="" [Required]
- *SessionContent* *enableConnPanel*='true/false' [Required]
- *SessionContent* *enableBookmarkPanel*='true/false' [Required]
- *SessionContent* *enableCtrlPanel*='true/false' [Required]
- *SessionContent* *enableHeaderPanel*='true/false' [Required]
- *SessionContent* *enablePlayerPanel*='true/false' [Required]
- *SessionContent* *sessionId*='' [Required]
- *SessionContent* *content*="0xRRGGBB/LOCALVIDEO" [Required]
- *Layout* *id*="" [Required]
- *Layout* *type*="Custom/AutoGrid/AutoStrip" [Required]
- *Layout* *nRow*="" [Required if type='AutoGrid']
- *Layout* *nCol*="" [Required if type='AutoGrid']
- *Layout* *colWidth*="" [Required if type='AutoGrid/AutoStrip']
- *Layout* *rowHeight*="" [Required if type='AutoGrid/AutoStrip']
- *Layout* *hpad*="" [Required if type='AutoGrid/AutoStrip']
- *Layout* *vpad*="" [Required if type='AutoGrid/AutoStrip']
- *Layout* *stripPos*="" [Required if type='AutoStrip']
- *Layout* *nStrip*="" [Required if type='AutoStrip']
- *Layout* *border*="show/hide/autohide" [Required if type='Custom']
- *Layout* *borderStyle*="[L:T:R:B]/[css]" [Required if type='Custom']
- *Layout* *header*="hide/show/sutohide" [Required if type='Custom']
- *Layout* *headerStyle*="basic/extended/[css]" [Required if type='Custom']
- *Layout* *active*="true/false" [Required]
- *Window* *id*="" [Required]
- *Window* [Required]
- *Window* *content*='MediaThumbnail/MediaStream' [Required]
- *Window* *zIndex*="" [Required]

- *Window content*="SPLASH/REMOTEVIDEO" [Required]
- *ConnectionContent MediaStreamIOPortTypeTitle*="" [Required]
- *ConnectionContent MediaStreamIOPortElementID*='' [Required]
- *ConnectionContent sessionId*='' [Required]
- *ConnectionContent connId*='' [Required]
- *ConnectionContent srcKbmCtrl*='' [Not Supported]
- *ThumbnailContent MediaStreamIOPortTypeTitle*="" [Required]
- *ThumbnailContent MediaStreamIOPortElementID*='' [Required]
- *ThumbnailContent imageURL*="" [Required]
- *ThumbnailContent refreshInterval*='' [Required]
- *Properties fps* [Required]
- *Properties maximize*='true/false' [Required]
- *Properties mute*='true/false' [Required]
- *Properties remoteKbmCtrl*='true/false' [Required]
- *Properties scale* [Required]
- *Properties visibility*='true/false' [Required]
- *Properties volume* [Required]
- *Properties aspect* [Required]
- *Properties border* [Required]
- *Properties hpos* [Required]
- *Properties vpos* [Required]
- *Properties hres* [Required]
- *Properties vres* [Required]
- *Properties layoutPos* [Required]
- *Properties scrollHPos* [Required]
- *Properties scrollVPos* [Required]
- *Properties transparency* [Required]

Remote Operated Client DisplayConfig


```
{
  "Display": {
    "Panel": {
      "@content": "",
      "Geometry": {
        "@transparency": "",
        "@hres": "100%",
        "@border": "",
        "@zIndex": "",
        "@vpos": 0,
        "@aspect": "",
        "@vres": "100%",
        "@hpos": 0
      },
      "@id": "",
      "@enable": ""
    },
    "@hres": "AUTO",
    "@vres": "AUTO",
    "@configurable": ""
  },

  "Canvas": {
    "Layout": {
      "@nRow": "",
      "@border": "hide",
      "@header": "hide",
      "@type": "",
      "@hpad": "",
      "@borderStyle": "[8px]",
      "@nStrip": "",
      "@colWidth": "",
      "@active": true,
      "@headerStyle": "basic",
      "@stripPos": "",
      "@rowHeight": "",
      "@nCol": "",
      "@id": "LayoutId1",
      "@vpad": ""
    },

    "@hres": 1280,
    "@vres": 1024,
    "Background": {
      "@content": "",
      "ImageContent": {
        "@imageURL": "",
        "@color": ""
      },
      "SessionContent": {
        "@enablePlayerPanel": "",

```

```
        "@enableCtrlPanel": "",
        "@enableBookmarkPanel": "",
        "@sessionId": "",
        "@enableConnPanel": "",
        "@enableHeaderPanel": ""
    },
    "@zIndex": "",
    "@enable": ""
},
"@id": "",
"Window": {
    "@content": "",
    "ThumbnailContent": {
        "@imageUrl": "",
        "@MediaStreamIOPortElementID": "",
        "@MediaStreamIOPortTypeTitle": "",
        "@refreshInterval": ""
    },
    "ConnectionContent": {
        "@hres": "",
        "@srcKbmCtrl": "",
        "@MediaStreamIOPortElementID": "",
        "@vres": "",
        "@MediaStreamIOPortTypeTitle": "",
        "@sessionId": "",
        "@connId": ""
    },
    "@id": "Win1",
    "Properties": {
        "@scrollHPos": 0,
        "@scale": "fitWindow",
        "@border": "",
        "@hres": "100%",
        "@maximize": false,
        "@zIndex": 100,
        "@aspect": "",
        "@scrollVPos": 0,
        "@transparency": 0,
        "@remoteKbmCtrl": false,
        "@vpos": 0,
        "@layoutId": "LayoutId1",
        "@vres": "100%",
        "@layoutPos": "",
        "@visibility": true,
        "@hpos": 0,
        "@fps": 15,
        "@mute": false,
        "@volume": "50%"
    }
}
}
}
```

Remote Operated Client DisplayStatus

```
{
  "Display": {
    "Panel": {
      "@content": "",
      "Geometry": {
        "@transparency": "",
        "@hres": "100%",
        "@border": "",
        "@zIndex": "",
        "@vpos": 0,
        "@aspect": "",
        "@vres": "100%",
        "@hpos": 0
      },
      "@id": "",
      "@enable": ""
    },
    "@hres": "AUTO",
    "@vres": "AUTO",
    "@configurable": ""
  },
  "Canvas": {
    "Layout": {
      "@nRow": "",
      "@border": "hide",
      "@header": "hide",
      "@type": "",
      "@hpad": "",
      "@borderStyle": "[8px]",
      "@nStrip": "",
      "@colWidth": "",
      "@active": true,
      "@headerStyle": "basic",
      "@stripPos": "",
      "@rowHeight": "",
      "@nCol": "",
      "@id": "LayoutId1",
      "@vpad": ""
    },
    "@hres": 1280,
    "@vres": 1024,
    "Background": {
      "@content": "",
      "ImageContent": {
        "@imageURL": "",
        "@color": ""
      },
      "SessionContent": {
        "@enablePlayerPanel": ""
      }
    }
  }
}
```

```

        "@enableCtrlPanel": "",
        "@enableBookmarkPanel": "",
        "@sessionId": "",
        "@enableConnPanel": "",
        "@enableHeaderPanel": ""
    },
    "@zIndex": "",
    "@enable": ""
},
"@id": "",
"Window": {
    "@content": "",
    "ThumbnailContent": {
        "@imageUrl": "",
        "@MediaStreamIOPortElementID": "",
        "@MediaStreamIOPortTypeTitle": "",
        "@refreshInterval": ""
    },
    "ConnectionContent": {
        "@hres": "",
        "@srcKbmCtrl": "",
        "@MediaStreamIOPortElementID": "",
        "@vres": "",
        "@MediaStreamIOPortTypeTitle": "",
        "@sessionId": "",
        "@connId": ""
    },
    "@id": "Win1",
    "Properties": {
        "@scrollHPos": 0,
        "@scale": "fitWindow",
        "@border": "",
        "@hres": "100%",
        "@maximize": false,
        "@zIndex": 100,
        "@aspect": "",
        "@scrollVPos": 0,
        "@transparency": 0,
        "@remoteKbmCtrl": false,
        "@vpos": 0,
        "@layoutId": "LayoutId1",
        "@vres": "100%",
        "@layoutPos": "",
        "@visibility": true,
        "@hpos": 0,
        "@fps": 15,
        "@mute": false,
        "@volume": "50%"
    }
}
}
}
}

```

Appendix B. REST API - Basic Tutorial

The following tutorial walks you through a basic set of calls used in the API and how to execute them.

REST Tester

Prerequisites for the tutorial:

- Login to admin UI
- Add user: user1
- Add device(webclient): webclient

We will use the BarcoRESTdiagnostics tool for demonstrating the use of the API.

BarcoRESTdiagnostics: <http://<msip>/clients/gui/player/resttester/BarcoRESTdiagnostics.html>

Important fields in the tester are:

- IPaddress: Enter your Management Server IP Address
- User: Enter the username for an existing user.
- Password: Enter password for above user
- Resource: Resource is the web client resource existing in the MS
- Service version: Service version is the current service version of your MS.
- Login: Link to login the user with above details
- Logout: Link for user logout
- URL: Enter REST URL here `http://<msip>/controller/<resource_path>?<query_params>`
- Accept Header: The format in which you expect your response from server (e.g. application/json)
- HTTP Method: Allows to choose the method type for your REST request.
- SUBMIT: Used to submit your REST requests
- Request: Hidden box, which is visible when request method is POST. Add your request body here for POST/PUT requests.
- Response: Displays the response for your REST request.
- Notifications: Displays the notifications being sent by server.

Service Login

Open BarcoRESTdiagnostics.html in a browser window.

Following are the details to be filled out in the REST Tester page for executing the Service Login request. Click on the 'Login' link after filling in the below details.

- IPaddress: MS ip address
- User: Enter the username for an existing user.
- Password: Enter password for above user
- Resource: Resource is the web client resource existing in the MS
- Service version: Service version is the current service version of your MS.

Service Login Response is available in the Response text field and a Long Polling channel is established for the User. Long-polling is achieved by calling HTTP GET on a custom 'X-Barco-notification-channel' response header. **To see more details about how a Long polling notification channel is setup, please refer to the ServiceLogin API call in this document.**

GetPort - Sources

Fetching the list of source ports from the system.

Following are the details to be filled out in the REST Tester page for executing the GetPort request

- URL: `http://10.51.51.72/controller/Resource/Port?type=SrcPort`
- METHOD: HTTP GET
- Accept Header: `application/json`

Upon hitting SUBMIT, a response will appear in the Response text field. From the response, please note down title for one of the source ports. For e.g. "tx1"

GetPort - Destination

Fetching the list of destination ports from the system.

Following are the details to be filled out in the REST Tester page for executing the GetPort request

- URL: `http://10.51.51.72/controller/Resource/Port?type=DstPort`
- METHOD: HTTP GET
- Accept Header: `application/json`

Upon hitting SUBMIT, a response will appear in the Response text field. From the response, please note down title for one of the destination ports. For e.g. "rx1"

View

View a source stream on the destination.

Following are the details to be filled out in the REST Tester page for executing the View request. Enter the request body in the Request field which appears on selection of the HTTP POST method from drop down. Hit SUBMIT to send the request.

- URL: `http://10.51.51.72/controller/Session`
- METHOD: HTTP POST
- Accept Header: `application/json`
- REQUEST BODY:

```
{ "View": {  
  "SrcPort": {  
    "@portTitle": "tx1" },  
  "DstPort": {  
    "@portTitle": "rx1" } } }
```

From the response, please note down session ID. Should be found as an attribute "@id" at the bottom of Session response

StopSession

Stop the session started by the View call

Following are the details to be filled out in the REST Tester page for executing the Stop Session request. Enter the request body in the Request field which appears on selection of the HTTP POST method from drop down. Hit SUBMIT to send the request.

- URL: `http://10.51.51.72/controller/Session`
- METHOD: HTTP POST
- Accept Header: `application/json`
- REQUEST BODY:

```
{"StopSession": {"@sessionId": ""}}
```

Please use the sessionId from the View response

Record

Start a recording session

Following are the details to be filled out in the REST Tester page for executing the Record request Enter the request body in the Request field which appears on selection of the HTTP POST method from drop down. Hit SUBMIT to send the request.

- URL: `http://10.51.51.72/controller/Session`
- METHOD: HTTP POST
- Accept Header: `application/json`
- REQUEST BODY:

```
{ "Record": {  
  "SrcPort": {  
    "@portTitle": "tx1" },  
  "DstFile": {  
    "@fileTitle": "tx1record" } } }
```

Please use the sessionId from the Record response and run the StopSession(see above) to stop the recording

GetFile - Recordings

Fetch the recording files

Following are the details to be filled out in the REST Tester page for executing the GetFile request. Hit SUBMIT to send the request.

- URL: `http://10.51.51.72/controller/Resource/File`
- METHOD: HTTP GET
- Accept Header: `application/json`

Please note down the fileNID for one of the files from the GetFile response. e.g. "default.mediastreamfileresourcelist.msfrl_cb2bbd33-285b-4c1a-babd-20e7bb19df07"

Playback

Start a playback session

Following are the details to be filled out in the REST Tester page for executing the Playback request Enter the request body in the Request field which appears on selection of the HTTP POST method from drop down. Hit SUBMIT to send the request.

- URL: `http://10.51.51.72/controller/Session`
- METHOD: HTTP POST
- Accept Header: `application/json`
- REQUEST BODY:

```
{ "Playback": {  
  "SrcFile": {  
    "@fileNID": "default.mediastreamfileresourcelist.msfrl_eb2bbd33-285b-4c  
    "DstPort": {  
      "@portTitle": "rx1" } } } }
```

Please use the sessionId from the Playback response and run the StopSession(see above) to stop the recording

View live source using mediaURL

View Live source From SDK Player

Following actions requires REST Tester and sdkPlayer (**BarcoSDK/player/src/barco/BarcoPlayer.html**). The URL for the sdkplayer will be **`http://<webserver_ip>/<basePath>/BarcoSDK/player/src/barco/BarcoPlayer.html`**

1. Enter `http://<MSIP>/controller/Resource/Port?type=SrcPort&selectFormat=BASICUR`
2. destPortTitle should be the resource_username using in above field 2 & 4 e.g. web-client_barco. **Please refer to Example 2.17 for Note regarding URLs**
3. Select the HTTP Method as GET
4. Hit SUBMIT.
5. Should see a list of ports that user is allowed to see in the Response field.
- 6. Copy this 'mediaURL' from response and paste in sample sdk player test box**
- 7. Keep the BarcoRESTdiagnostics loggedin for LIVE to work.**
8. Sample Response:

```
{  
  "Resource": {  
    "Port": {  
      "Groups": {  
        "Tag": [  
          {  
            "@Name": "Location",  
            "@Value": "Sunnyvale"  
          },  
          {  
            "@Name": "Department",  
            "@Value": "Engineering"  
          }  
        ]  
      },  
      "@NID": "default.mediastreamsrcportresourcelist.mssprl_c3dfb990-6b80-4408-
```



```
    "@streamType": "RTP",
    "@portType": "StreamSrc",
    "@streamingURL": "http://10.51.51.18/mediaroom/index.m3u8?dirID=/data1&med
    "@hls": true,
    "@portId": 1,
    "@presence": "unavailable",
    "@mediaURL": "http://10.51.51.65/clients/player?requestName=View&SrcPort.p
    "@state": "Unmanaged",
    "@title": "axis",
    "Permissions": {
      "@add": false,
      "@modify": false,
      "@delete": false,
      "@streamto": false,
      "@viewfrom": true,
      "@read": true
    }
  },
  "@total": 1,
  "@from": 1
}
```

Playback recorded file using mediaURL

Playback recorded file From SDK Player

Following actions requires REST Tester and sdkPlayer (**BarcoSDK/player/src/barco/BarcoPlayer.html**). The URL for the sdkplayer will be **http://<webserver_ip>/<basePath>/BarcoSDK/player/src/barco/BarcoPlayer.html**

1. Enter **http://<MSIP>/controller/Resource/File?selectFormat=BASICURL&dstPortTitle**
2. destPortTitle should be the resource_username using in above field 2 & 4 e.g. web-client_barco. **Please refer to Example 2.17 for Note regarding URLs**
3. Select the HTTP Method as GET
4. Hit SUBMIT.
5. Should see a list of files that user is allowed to see in the Response field.
- 6. Copy this 'mediaURL' from response and paste in sample sdk player test box**
- 7. Keep the BarcoRESTdiagnostics loggedin for LIVE to work.**
8. Sample Response:

```
{
  "Resource": {
    "File": {
      "@NID": "default.mediastreamfileresourcelist.msfrl_4a6f4f7c-028a-4fba-b852
      "@clipID": "aeb29c42-701a-4a79-990f-ecb6a04cd203",
      "Permissions": {
        "@playback": true,
        "@export": true,
        "@modify": true,
        "@delete": true,
        "@read": true
      }
    }
  }
}
```

```
    },
    "@size":14831321,
    "@readOnly":false,
    "@bypassAutoStorageRules":false,
    "Groups":{
      "Tag":{
        "@Name":"","
        "@Value":""
      }
    },
    "@streamType":"RTP",
    "@thumbnailURL":"http://10.51.51.9/mediaportal/thumbnail?dirID=/data&tim
    "@hls":false,
    "MetaData":"__EMPTY_STRING_",
    "@dateCreated":1396414984489,
    "@fileType":"MediaClip",
    "@dateModified":1396414984489,
    "@playlength":19333,
    "@mediaURL":"http://10.51.51.65/clients/player?requestName=Playback&SrcFi
    "@title":"rtptx2",
    "@state":"Ready",
    "@sessionConfigNID":"","
    "@mediaGroupID":"","
    "Download":{
      "Media":{
        "@format":"","
        "@state":"NotReady",
        "@stale":false
      }
    }
  },
  "@total":1,
  "@from":1
}
```

Download Media

Prepare File for download

We first subscribe the user for notifications and then prepare file for download.

1. Enter `http://<MSIP>/controller/Resource/File?selectFormat=EXTENDEDURL&subscrip`
2. Select the HTTP Method as GET
3. Hit SUBMIT.
4. Should see a list of files that user is allowed to see in the Response field.
5. Enter `http://<MSIP>/controller/Resource/File/PrepareForDownload` in the URL field.
6. Select the HTTP Method as POST
7. In the Request text area, add your request body. E.g.

```
{ "PrepareFileForDownload": {  
  "@byNID": "default.mediastreamfileresourcelist.msfrl_959ecf85-022b-4146-8641-6e",  
  "@format": "Standard" }}
```

byNID is the 'NID' for the RTP file you want to download (**get this from the GetFile Response**). **check the permissions element in the getFile response for verifying the permissions on the file. Should have export permissions for the file to allow download.** e.g.

```
"Permissions": {  
  "@playback": true,  
  "@export": true,  
  "@modify": false,  
  "@delete": false,  
  "@read": true  
},
```

format is 'Proprietary' (Barco Native format) or 'Standard' (Zip with mp4)

8. Hit SUBMIT.

9. Will get a downloadURL in file response in the Notifications field, once the file is ready for download.

10. Use any browser to download media using downloadURL

Create Media using Cutpoints

Create Media using Cutpoints

1. Enter `http://<MSIP>/controller/Resource/File/CreateMediaFromCutList` in the URL
2. Select the HTTP Method as POST
3. In the Request text area, add your request body. E.g.

```
{  
  "CreateMediaFromCutList": {  
    "@byNID": "default.mediastreamfileresourcelist.msfrl_959ecf85-022b-4146-8641-6e",  
    "@cutList": "0:30000,40000:60000,70000:120000",  
    "@title": "newFileTitle",  
    "MetaData": {  
      "Tag": {  
        "content": "Value",  
        "@Name": ""  
      }  
    },  
    "Description": "testing"  
  }  
}
```

byNID is the 'NID' for the RTP file you want to download (**get this from the GetFile Response**). cutList is a comma separated list of start:end offsets in the media in milliseconds. E.g. 0:30000 is 30 seconds of media from start offset. Title is the name for the new media being created.

4. Hit SUBMIT.

5. Will get a 201 response for request accepted.

8. Will get a file notification in the Notifications field, when the created media is ready

API

Following is a list of pointers to the API call documentation over REST. Note that the request bodies mentioned as sample are not minimal. Some of the elements/attributes could be optional.

Table B.1.

ServiceLogin	Refer Example 1.1 Service Login Request
GetPort - Sources	Refer Example 2.16 Get Port Request
GetPort - Destination	Refer Example 2.16 Get Port Request
View	Refer Example 4.23 Basic View Request by Title
StopSession	Refer Example 4.21 Stop Session Request
Record	Refer Example 4.45 Basic Record Request
GetFile - MediaClips	Refer Example 2.53 Get File Request
Playback	Refer Example 4.35 Basic Playback Request

NOTE: BarcoRESTdiagnostics tool is qualified for Google Chrome and IE-9. When loading the page in IE-9 please ensure you are NOT running the page in compatibility view mode. The BarcoRESTdiagnostics.html is a pure JavaScript implementation using XMLHttpRequests and does not use a JQuery or any other 3rd party library. To debug or view the communication logs use the browsers development tools.

Release Notes

Summary of changes from 4.0 to 4.1

Added Requests

Prepare File: prepare a file to download for Proprietary/Standard/Both formats
Create Media From Cutlist: Creates Media from existing media using start and end
Trim File with Cutlist: Trim out sections of an existing media file using start
Clear Download Cache: Remove cache memory , which is occupied while downloading
Download Media: Download the provided media file to the media store, you can use
Inject Media: Uploads the provided media file to the media store

Updated Requests

GetFile: Added new selectFormat BASICCONNECTIONINFO which will return connection

GetDir: Added new selectFormat UPLOADURL which will return upload URL. Which is

Added Notifications

FileUpdate: Added file Add/Update/Delete notifications
FileConnectionUpdate: Added connection Add/Delete notification
FileConnectionStateUpdate: Added file connection state update notification

Summary of changes from 4.1 to 4.2

Added Requests

Get DisplayProfile: The list of DisplayProfile objects that the Client has access to

Updated Requests

GetFile: Added new query parameter whereDownloadState
GetDir: Added new query parameter dstPortTitle
DeleteFile: Added new query parameter moveToRecycleBin
GetDevice: Updated to allow filter for whereType="Proxy-Device/WebClient-Device"

Updated Accept header to "application/xml" for Session and Bookmark API's (Start, Stop, Get, Set)

Create Media From Cutlist: Cutlist values are validated against certain rules in the API

Inject Media: Added support to enable HLS streaming
Play: Slowmotion is not supported