

Kommander WebSocket Protocol Documentation

Applicable to all Kommander series and customized versions. Latest update corresponds to software version **v6.4.12**.

Tel: +86 177 5686 4211

Website: www.kommander.com.cn/en

Hunan Bowan Technology Co., Ltd.

Contents

1. Revision History	5
2. Establishing Connections	5
3. General Data Header Format	6
4. Specific Communication Messages	7
4.1.7 Mute	11
4.1.8 Volume	12
4.1.9 Resource Thumbnail	12
4.1.10 Play	13
4.1.11 Pause	13
4.1.12 Stop	14
4.1.13 Play Specific Scenario	14
4.1.14 Invoke Scenario by Index	14
4.1.15 Drag-and-Drop File (Add/Replace)	15
4.1.16 Update Program File List (Push/Request)	15
4.1.17 Get Visible Canvas Info	17
4.1.18 Modify File Position	17
4.1.19 Delete File	18
4.1.20 Next Page	19
4.1.21 Update Screen List	19
4.1.22 Update Scenario by Resource Name	20
4.1.23 Save As Playback Scenario	20
4.1.24 Clear All Files on Canvas	21
4.1.25 Scenario Mode Change	21
4.1.26 Get Scenario Mode	21
4.1.27 Layer Adjustment	22
4.1.28 Output Pre-edits to Real-Time	22
4.1.29 Copy Output to Pre-edit	22
4.1.30 Get Lottery State	23

4.1.31 Update Lottery State.....	23
4.1.32 Canvas Zoom Scale Change.....	23
4.1.33 Set User Mode	23
4.1.34 Get User Mode	24
4.1.35 Get Scenario Usage Flags.....	24
4.1.36 Get Maximum Texture Size	24
4.1.37 Set File Playback State	25
4.1.38 Set File Mute	25
4.1.39 Get File Mute State	25
4.1.40 Set Browser URL	26
4.1.41 Update State	26
4.1.42 Set Currently Selected File	26
4.1.43 Get Currently Selected File.....	26
4.1.44 Get Central Control Public Configuration	27
4.1.45 Get Central Control Image	27
4.1.46 Get Central Control Module List	27
4.1.47 Program File Seek	28
4.1.48 Get Program Schedule.....	28
4.1.49 Get Program File Volume	29
4.1.50 Set Program File Volume.....	29
4.1.51 Shutdown or Reboot	29
4.1.52 Clear Cache	30
4.1.53 Replace Window with Specified Media	30
4.1.54 Set Window Position.....	30
4.1.55 Enable/Disable All Monitors	31
4.1.56 Open Window with Specified Media	31
4.1.57 Add Teleprompter Content.....	32
4.1.58 Set Teleprompter Loop Mode	32
4.1.59 Set Teleprompter Current Line	33
4.1.60 Update Subtitle Content	33

4.1.61 Get File Playback State	33
4.2 Push Notifications to External Endpoints	34
4.2.5 Update Resource Thumbnail	37
4.2.6 Update Lottery State	37
4.2.7 Update Visible Canvas Information.....	37
4.2.8 Update User Mode	38
4.2.9 Update Scenario Usage Flags	38
4.2.10 Media Library Changed	39
4.2.11 Update Black Screen Status	39
4.2.12 Set File Mute	39
4.2.13 Update User Mode.....	40
4.2.14 Update Browser URL.....	40
4.2.15 Update State Changed	41
4.2.16 Update Currently Selected File	41
4.2.17 Update Central Control Public Configuration	41
4.2.18 Update File Playback State.....	42
4.2.19 Update Schedule	42
4.2.20 Update File Volume	43

1. Revision History

Content	Modifier/Date
Initial document creation	Ren Xian / 2021/03/29

2. Establishing Connections

2.1 Kommander as the Server

- **Port:** 1702
- **Protocol:** WebSocket
- **Host:** Local IP
- **Example:** `ws://127.0.0.1:1702`

2.2 Kommander Connecting to External Endpoints

2.2.1 Connecting to a Platform

Parameters:

Parameter Name	Data Type	Description
<code>from</code>	String	Identifies the terminal type. Use the fixed value "kommander".
<code>params</code>	JSON String	Additional parameters (see below).

Params Structure:

Parameter Name	Data Type	Description
<code>name</code>	String (32)	Server name.

Example:

```
ws://127.0.0.1:7200?from=kommander&params={"name":"kommander_001"}
```

Note:

- Implement heartbeat checks using a **ping-pong mechanism**. Avoid reconnecting unless a disconnection is confirmed to prevent command delivery issues.

3. General Data Header Format

3.1 Client-Sent Data

Parameter Name	Data Type	Description
<i>identificationID</i>	String	Generated by the client. Kommander echoes this value. Optional for push notifications.
<i>KommanderMsg</i>	String	Message type (e.g., <i>KommanderMsg_Authentication</i> for login).
<i>params</i>	JSON	Payload (JSON object, array, or value). See Section 4 for details. Empty if unused.

3.2 Server Response or Push Notification

Parameter Name	Data Type	Description
<i>identificationID</i>	String	Echoed from the client. Optional for push notifications.
<i>KommanderMsg</i>	String	Message type.
<i>code</i>	Integer	1 for success, other values indicate failure.
<i>data</i>	JSON Object	Response payload (varies by command). Empty if unused.

Notes:

- Subsequent protocols describe only the *data* field; the other three fields are mandatory.
- Logging is recommended.** Use the *code* field to determine retry requirements for failed commands.
- The software logs all command transactions for audit and analysis.

4. Specific Communication Messages

4.1 Client Requests

4.1.1 Kommander Login

Protocol Name: *KommanderMsg_Authentication*

Request Parameters:

Parameter	Type	Description
<i>username</i>	String	User account name.
<i>password</i>	String	User password.
<i>ip</i>	String	IP address of the client.
<i>deviceid</i>	String	Unique device identifier (6-character random string).
<i>seize</i>	Bool	Whether to preempt existing connections.
<i>bUseRtmpPushStream</i>	Bool	Whether to enable RTMP streaming.

Success Response:

Parameter	Type	Description
<i>serverName</i>	String	Server name.
<i>playUrl</i>	String	Stream playback URL.
<i>protocolVersion</i>	String	Protocol version.
<i>productName</i>	String	Product name.
<i>showPreediting</i>	String	Version control flag.
<i>showUpdate</i>	Bool	Version control flag.

Parameter	Type	Description
<i>supportMode</i>	Int	Permission level.
<i>version</i>	String	Software version.

Failure Response:

Parameter	Type	Description
<i>error</i>	Int	Error code.
<i>msg</i>	String	Error description.
<i>deviceid</i>	String	Client-provided device identifier.
<i>protocolVersion</i>	String	Protocol version.
<i>productName</i>	String	Product name.
<i>showPreediting</i>	Bool	Version control flag.
<i>showUpdate</i>	Bool	Version control flag.

4.1.2 Login Mode

Protocol Name: *KommanderMsg_LoginMode*

Request Parameters:

Parameter	Type	Description
<i>type</i>	Int	Mode identifier: <i>0x01</i> (Kommander), <i>0x02</i> (Control Panel), <i>0x04</i> (Lite).

Response: None.

4.1.3 Project Information

Protocol Name: *KommanderMsg_ProjectInfo*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>projectName</i>	String	Name of the active project.

4.1.4 Media Library Information

Protocol Name: *KommanderMsg_GetMediaLibs*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>MediaLibs</i>	Array	Media library data array, including materials, playlists, and scenarios.

MediaLibs Structure:

Parameter	Type	Description
<i>MediaGroup</i>	String	Media group description.
<i>MediaGroupType</i>	Int	Group type (see <i>E_MediaGroupType</i>).
<i>data</i>	Array	Media items or subgroups.

data Structure:

1) When *isGroup* is true:

Parameter	Type	Description
<i>isGroup</i>	Bool	Indicates a subgroup.
<i>name</i>	String	Subgroup name.
<i>data</i>	Array	Nested media items (recursive structure).

2) When *isGroup* is false:

Parameter	Type	Description
<i>isGroup</i>	Bool	Indicates a media item.
<i>type</i>	Int	Media type (see <i>E_MediaType</i>).
<i>name</i>	String	Media name.
<i>id</i>	String	Unique GUID of the media.
<i>isInsert</i>	Bool	Whether the media is an interruptible insert (fixed <i>false</i> for <i>MGT_DEFAULT</i>).

Example:

```

json
{
  "MediaLibs": [
    {
      "MediaGroup": "Material",
      "MediaGroupType": 1,
      "data": [
        {
          "isGroup": false,
          "id": "{bc89c8f7-6eb3-4667-8e47-ad07eb92e64a}",
          "name": "Plan1",
          "type": 1,
          "isInsert": false
        }
      ]
    }
  ]
}

```

```
{
  "isGroup": true,
  "name": "Folder",
  "data": [ /* Sub-items */ ]
}
]
```

4.1.5 Get Black Screen Status

Protocol Name: *KommanderMsg_GetBlackScreen*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>blackscreenSwitching</i>	Bool	<i>true</i> : Black screen transition in progress; <i>false</i> : Transition complete.
<i>blackscreen</i>	Bool	<i>true</i> : Black screen enabled.

4.1.6 Set Black Screen Status

Protocol Name: *KommanderMsg_UpdateBlackScreen*

Request Parameters:

Parameter	Type	Description
<i>blackscreen</i>	Bool	<i>true</i> : Enable black screen.

Response: None (asynchronous operation; wait for status updates from the PC).

4.1.7 Mute

Protocol Name: *KommanderMsg_Mute*

Request Parameters:

Parameter	Type	Description
<i>mute</i>	Bool	<i>true</i> to enable mute (optional).

Response Parameters:

Parameter	Type	Description
<i>mute</i>	Bool	Current mute status (<i>true</i> = muted). If the request is empty, returns the current state.

4.1.8 Volume

Protocol Name: *KommanderMsg_Volume*

Request Parameters:

Parameter	Type	Description
<i>volume</i>	Int	Volume level (0–100, optional).

Response Parameters:

Parameter	Type	Description
<i>volume</i>	Int	Current volume (0–100). If the request is empty, returns the current level.

4.1.9 Resource Thumbnail

Protocol Name: *KommanderMsg_GetThumbnaillImage*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Resource GUID.

Response Parameters:

Parameter	Type	Description
<i>id</i>	String	Resource GUID.
<i>image</i>	String	Base64-encoded image data.

4.1.10 Play

Protocol Name: *KommanderMsg_Play*

Request Parameters:

Parameter	Type	Description
<i>onlySetReal</i>	Bool	<i>true</i> triggers real-time execution; <i>false</i> follows canvas settings.

Response: None.

4.1.11 Pause

Protocol Name: *KommanderMsg_Pause*

Request Parameters:

Parameter	Type	Description
<i>onlySetReal</i>	Bool	<i>true</i> triggers real-time execution; <i>false</i> follows canvas settings.

Response: None.

4.1.12 Stop

Protocol Name: *KommanderMsg_Stop*

Request Parameters:

Parameter	Type	Description
<i>onlySetReal</i>	Bool	<i>true</i> triggers real-time execution; <i>false</i> follows canvas settings.

Response: None.

4.1.13 Play Specific Scenario

Protocol Name: *KommanderMsg_PlayPrePlan*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Scenario GUID.
<i>onlySetReal</i>	Bool	<i>true</i> triggers real-time execution; <i>false</i> follows canvas settings.

Response Parameters:

Parameter	Type	Description
<i>id</i>	String	Scenario GUID.
<i>onlySetReal</i>	Bool	Echoes the request parameter.

4.1.14 Invoke Scenario by Index

Protocol Name: *KommanderMsg_IndexInvokePrePlan*

Request Parameters:

Parameter	Type	Description
<i>groupIndex</i>	Int	Group index of the scenario.
<i>index</i>	Int	Scenario index within the group.
<i>onlySetReal</i>	Bool	<i>true</i> triggers real-time execution; <i>false</i> follows canvas settings.

Response: None.

4.1.15 Drag-and-Drop File (Add/Replace)

Protocol Name: *KommanderMsg_DropFile*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Resource GUID.
<i>x-ratio</i>	Double	Normalized X-coordinate (0.0–1.0).
<i>y-ratio</i>	Double	Normalized Y-coordinate (0.0–1.0).

Failure Response:

Parameter	Type	Description
<i>error</i>	String	<i>0</i> = Unknown error; <i>1</i> = Exceeds display limit; <i>2</i> = Unsupported file.
<i>msg</i>	String	Error description.

4.1.16 Update Program File List (Push/Request)

Protocol Name: *KommanderMsg_UpdateProgFileList*

Request Parameters: None.

Response Parameters: Array of:

Parameter	Type	Description
<i>name</i>	String	Media name.
<i>id</i>	String	Media GUID.
<i>bMute</i>	Bool	Mute status.
<i>mediaObject</i>	JSON Array	Resource details (see below).
<i>dstRect</i>	JSON Object	Position data (see below).

***mediaObject* Structure:**

Parameter	Type	Description
<i>name</i>	String	Resource name.
<i>id</i>	String	Resource GUID.
<i>type</i>	Int	Resource type (see <i>E_MediaType</i>).
<i>url</i>	String	URL (required for web resources).

***dstRect* Structure:**

Parameter	Type	Description
<i>leftTop.x</i>	Double	Top-left X-coordinate.
<i>leftTop.y</i>	Double	Top-left Y-coordinate.
<i>rightTop.x</i>	Double	Top-right X-coordinate.

Parameter	Type	Description
<i>rightTop.y</i>	Double	Top-right Y-coordinate.
<i>leftBottom.x</i>	Double	Bottom-left X-coordinate.
<i>leftBottom.y</i>	Double	Bottom-left Y-coordinate.
<i>rightBottom.x</i>	Double	Bottom-right X-coordinate.
<i>rightBottom.y</i>	Double	Bottom-right Y-coordinate.

4.1.17 Get Visible Canvas Info

Protocol Name: *KommanderMsg_GetViewCanvasInfo*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>cavasXOffset</i>	Int	X-axis offset of the visible canvas's top-left corner.
<i>cavasYOffset</i>	Int	Y-axis offset of the visible canvas's top-left corner.
<i>canvasViewWidth</i>	Int	Visible canvas width.
<i>canvasViewHeight</i>	Int	Visible canvas height.
<i>canvasWidth</i>	Int	Total canvas width.
<i>canvasHeight</i>	Int	Total canvas height.
<i>CanvasNegativeX</i>	Int	Maximum negative X-coordinate.
<i>CanvasNegativeY</i>	Int	Maximum negative Y-coordinate.

4.1.18 Modify File Position

Protocol Name: *KommanderMsg_filePositionChanged*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.
<i>leftTop.x</i>	Double	Top-left X-coordinate.
<i>leftTop.y</i>	Double	Top-left Y-coordinate.
<i>rightTop.x</i>	Double	Top-right X-coordinate.
<i>rightTop.y</i>	Double	Top-right Y-coordinate.
<i>leftBottom.x</i>	Double	Bottom-left X-coordinate.
<i>leftBottom.y</i>	Double	Bottom-left Y-coordinate.
<i>rightBottom.x</i>	Double	Bottom-right X-coordinate.
<i>rightBottom.y</i>	Double	Bottom-right Y-coordinate.

Response: None.

4.1.19 Delete File

Protocol Name: *KommanderMsg_DeleteProgFile*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.

Response: None.

4.1.20 Next Page

Protocol Name: *KommanderMsg_NextPage*

Request Parameters:

Parameter	Type	Description
<i>isGlobalTurnPage</i>	Bool	<i>true</i> = global page turn; <i>false</i> = single file/page turn.
<i>onlySetReal</i>	Bool	<i>true</i> applies changes only to real-time output.
<i>id</i>	String	GUID of the target file (required if <i>isGlobalTurnPage</i> is <i>false</i>).

Response: None.

4.1.21 Update Screen List

Protocol Name: *KommanderMsg_UpdateScreenList*

Request Parameters: None.

Response Parameters: Array of:

Parameter	Type	Description
<i>name</i>	String	Screen name.
<i>dstRect</i>	JSON Object	Position coordinates (see structure below).

***dstRect* Structure:**

Parameter	Type	Description
<i>leftTop.x</i>	Double	Top-left X-coordinate.
<i>leftTop.y</i>	Double	Top-left Y-coordinate.
<i>rightTop.x</i>	Double	Top-right X-coordinate.

Parameter	Type	Description
<i>rightTop.y</i>	Double	Top-right Y-coordinate.
<i>leftBottom.x</i>	Double	Bottom-left X-coordinate.
<i>leftBottom.y</i>	Double	Bottom-left Y-coordinate.
<i>rightBottom.x</i>	Double	Bottom-right X-coordinate.
<i>rightBottom.y</i>	Double	Bottom-right Y-coordinate.

4.1.22 Update Scenario by Resource Name

Protocol Name: *KommanderMsg_UpdatePreplan*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Resource name.

Response: None.

4.1.23 Save As Playback Scenario

Protocol Name: *KommanderMsg_AddPreplan*

Request Parameters:

Parameter	Type	Description
<i>name</i>	String	New scenario name. If empty, the PC auto-generates a name.

Success Response:

Parameter	Type	Description
<i>type</i>	Int	Scenario type (see <i>E_MediaType</i>).
<i>name</i>	String	Scenario name.
<i>id</i>	String	Scenario GUID.

Failure Response:

Parameter	Type	Description
<i>error</i>	Int	0 = Unknown error; 1 = Duplicate name.
<i>msg</i>	String	Error description.

4.1.24 Clear All Files on Canvas

Protocol Name: *KommanderMsg_ClearFileOnCanvas*

Request Parameters: None.

Response: None.

4.1.25 Scenario Mode Change

Protocol Name: *KommanderMsg_SetPretidingMode*

Request Parameters:

Parameter	Type	Description
<i>realMode</i>	Bool	<i>true</i> = real-time mode.

Response: None.

4.1.26 Get Scenario Mode

Protocol Name: *KommanderMsg_GetPretidingMode*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>realMode</i>	Bool	<i>true</i> = real-time mode.

4.1.27 Layer Adjustment

Protocol Name: *KommanderMsg_LayerAdjust*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.
<i>layerAdjust</i>	Int	<i>1</i> = Bring to front; <i>2</i> = Send to back; <i>3</i> = Move up; <i>4</i> = Move down.

Response: None.

4.1.28 Output Pre-edits to Real-Time

Protocol Name: *KommanderMsg_OutPutToEchoWindow*

Request Parameters: None.

Response: None.

4.1.29 Copy Output to Pre-edit

Protocol Name: *KommanderMsg_CopyOutPutToEdit*

Request Parameters: None.

Response: None.

4.1.30 Get Lottery State

Protocol Name: *KommanderMsg_GetLotteryState*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>state</i>	String	-1 = No active lottery; 0 = Stopped; 2 = Lottery in progress.

4.1.31 Update Lottery State

Protocol Name: *KommanderMsg_UpdateLotteryState*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>state</i>	String	-1 = No active lottery; -2 = Lottery switch triggered (no status update); 4 = No eligible participants; 5 = All prizes drawn.

4.1.32 Canvas Zoom Scale Change

Protocol Name: *KommanderMsg_CanvasScaleChanged*

Request Parameters:

Parameter	Type	Description
<i>scaleAdd</i>	Int	1 = Zoom in; -1 = Zoom out.

Response: None.

4.1.33 Set User Mode

Protocol Name: *KommanderMsg_UpdateUserMode*

Request Parameters:

Parameter	Type	Description
<i>userMode</i>	Int	Sets the operational mode (primarily for video wall controller configuration).

Response: None.

4.1.34 Get User Mode

Protocol Name: *KommanderMsg_GetUserMode*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>userMode</i>	Int	Current operational mode (primarily for video wall controller configuration).

4.1.35 Get Scenario Usage Flags

Protocol Name: *KommanderMsg_GetPrePlanUsingMark*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>outPuttingPrePlanId</i>	String	GUID of the currently outputting scenario.
<i>editingPrePlanId</i>	String	GUID of the scenario in pre-edit mode.

4.1.36 Get Maximum Texture Size

Protocol Name: *KommanderMsg_GetMaxTexture*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>textureWidth</i>	Int	Maximum texture width (16,000 pixels).
<i>textureHeight</i>	Int	Maximum texture height (16,000 pixels).

4.1.37 Set File Playback State

Protocol Name: *KommanderMsg_SetProgFilePlayState*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Program file GUID.
<i>playState</i>	Int	Playback state: 1 = Play, 2 = Pause, 3 = Stop.

Response: None.

4.1.38 Set File Mute

Protocol Name: *KommanderMsg_SetProgFileMute*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.
<i>mute</i>	Bool	<i>true</i> = Mute; <i>false</i> = Unmute.

Response: None.

4.1.39 Get File Mute State

Protocol Name: *KommanderMsg_GetProgFileMute*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>mute</i>	Bool	Current mute state (<i>true</i> = Muted).

4.1.40 Set Browser URL

Protocol Name: *KommanderMsg_SetBrowserUrl*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Media GUID (not file GUID).
<i>url</i>	String	Target URL.

Response: None.

4.1.41 Update State

Protocol Name: *KommanderMsg_UpdateStateChanged*

Request Parameters: None.

Response: None.

4.1.42 Set Currently Selected File

Protocol Name: *KommanderMsg_SetCurSelectFile*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Program file GUID. Empty = Deselect.

Response: None.

4.1.43 Get Currently Selected File

Protocol Name: *KommanderMsg_GetCurSelectFile*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>id</i>	String	Selected file GUID. Empty = No selection.

4.1.44 Get Central Control Public Configuration

Protocol Name: *KommanderMsg_GetCentreControlPublicConfig*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>name</i>	String	Base64-encoded image (system name).
<i>logo</i>	String	Base64-encoded image (logo).

4.1.45 Get Central Control Image

Protocol Name: *KommanderMsg_GetModuleImage*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>name</i>	String	Base64-encoded image (module name).
<i>logo</i>	String	Base64-encoded image (module logo).

4.1.46 Get Central Control Module List

Protocol Name: *KommanderMsg_GetModuleList*

Request Parameters: None.

Response Parameters:

Parameter	Type	Description
<i>name</i>	String	Base64-encoded image (module name).

Parameter	Type	Description
<i>logo</i>	String	Base64-encoded image (module logo).

4.1.47 Program File Seek

Protocol Name: *KommanderMsg_ProgFileSeek*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Program file GUID.
<i>pos</i>	String	Seek position (longlong value converted to string).

Response: None.

4.1.48 Get Program Schedule

Protocol Name: *KommanderMsg_GetSchedule*

Request Parameters: None.

Response Parameters: Array of:

Parameter	Type	Description
<i>content</i>	String	Command content.
<i>preplanName</i>	String	Scenario name included in the task.
<i>taskName</i>	String	Task name.
<i>exectime</i>	String	Execution time (format: YYYY-MM-DD HH:mm:ss).
<i>endTime</i>	String	End time (format: YYYY-MM-DD HH:mm:ss).
<i>bHaveDateLimite</i>	Bool	<i>false</i> = <i>startDate</i> and <i>endDate</i> are effective; <i>true</i> = no date limits.
<i>startDate</i>	String	Start date (format: YYYY-MM-DD).
<i>endDate</i>	String	End date (format: YYYY-MM-DD).
<i>bHaveWeekLimite</i>	Bool	<i>false</i> = <i>specifiedWeek</i> is effective; <i>true</i> = no weekly limits.

Parameter	Type	Description
<i>specifiedWeek</i>	Int	Bitmask for days: <i>0x01</i> = Monday, <i>0x02</i> = Tuesday, <i>0x04</i> = Wednesday, <i>0x08</i> = Thursday, <i>0x10</i> = Friday, <i>0x20</i> = Saturday, <i>0x40</i> = Sunday.

4.1.49 Get Program File Volume

Protocol Name: *KommanderMsg_GetProgFileVolume*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Program file GUID.

Response Parameters:

Parameter	Type	Description
<i>volume</i>	Int	Volume level (0–100).

4.1.50 Set Program File Volume

Protocol Name: *KommanderMsg_SetProgFileVolume*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Program file GUID.
<i>volume</i>	Int	Volume level (0–100).

Response: None.

4.1.51 Shutdown or Reboot

Protocol Name: *KommanderMsg_ShutDown*

Request Parameters:

Parameter	Type	Description
<i>shutDown</i>	Int	0 = Shutdown; 1 = Reboot.

Response: None.

4.1.52 Clear Cache

Protocol Name: *KommanderMsg_CleanCache*

Request Parameters: Array of:

Parameter	Type	Description
<i>id</i>	Int	Database file ID (primary key of <i>med_file</i>).
<i>name</i>	String	File name.

Response: None.

4.1.53 Replace Window with Specified Media

Protocol Name: *KommanderMsg_ReplaceWindowsForWindows*

Request Parameters:

Parameter	Type	Description
<i>mediald</i>	String	Media GUID.
<i>windowId</i>	String	Window GUID or index ID.
<i>mode</i>	Int	Playback mode: 0 = Loop, 1 = Pause on last frame, 2 = Stop, 3 = Jump to next scenario, -1 = Default.
<i>isGuid</i>	Bool	true = ID is a GUID; false = index ID.

Response: None.

4.1.54 Set Window Position

Protocol Name: *KommanderMsg_SetFilePosition*

Request Parameters:

Parameter	Type	Description
<i>id</i>	Int	Window ID.
<i>posX</i>	Int	X-coordinate of the center.
<i>posY</i>	Int	Y-coordinate of the center.

Response: None.

4.1.55 Enable/Disable All Monitors

Protocol Name: *KommanderMsg_EnableAllMonitor*

Request Parameters:

Parameter	Type	Description
<i>bOpen</i>	Bool	<i>true</i> = Enable; <i>false</i> = Disable.

Response: None.

4.1.56 Open Window with Specified Media

Protocol Name: *KommanderMsg_OpenWindows*

Request Parameters:

Parameter	Type	Description
<i>mediaId</i>	Int	Media ID.
<i>posX</i>	Int	X-coordinate of the center.
<i>posY</i>	Int	Y-coordinate of the center.
<i>nWidth</i>	Int	Window width (defaults to media resolution if empty).
<i>nHeight</i>	Int	Window height (defaults to media resolution if empty).
<i>playModel</i>	Int	Play mode: 1 = Fixed duration (seconds), 2 = Fixed loops.

Parameter	Type	Description
<i>count</i>	Int	Duration (seconds) or loop count.
<i>endMode</i>	Int	Action after completion: 0 = Stop, 1 = Pause on last frame, 2 = Jump to next scenario, 3 = Jump to specified scenario, 5 = Close window.
<i>prePlan</i>	String	Scenario ID (required if <i>endMode</i> is 3).
<i>clone</i>	Bool	<i>true</i> = Clone window; <i>false</i> = Do not clone.

Response: None.

4.1.57 Add Teleprompter Content

Protocol Name: *KommanderMsg_TeleprompterContent*

Request Parameters:

Parameter	Type	Description
<i>windows</i>	Int	Teleprompter window ID.
<i>isAppend</i>	Bool	<i>true</i> = Append content; <i>false</i> = Replace.
<i>content</i>	JSON Array	Content array (see structure below).

Content Structure:

Parameter	Type	Description
<i>text</i>	String	Display text.
<i>time</i>	Int	Display duration (seconds).

Response: None.

4.1.58 Set Teleprompter Loop Mode

Protocol Name: *KommanderMsg_SetTeleprompterLoop*

Request Parameters:

Parameter	Type	Description
<i>windows</i>	Int	Teleprompter window ID.
<i>bLoop</i>	Bool	<i>true</i> = Enable loop.

Response: None.

4.1.59 Set Teleprompter Current Line

Protocol Name: *KommanderMsg_SetTeleprompterLine*

Request Parameters:

Parameter	Type	Description
<i>windows</i>	Int	Teleprompter window ID.
<i>nLine</i>	Int	Line number to display.

Response: None.

4.1.60 Update Subtitle Content

Protocol Name: *KommanderMsg_SetSubTitleContent*

Request Parameters:

Parameter	Type	Description
<i>windows</i>	Int	Subtitle window ID.
<i>append</i>	Int	1 = Prepend, 2 = Append, 3 = Overwrite.
<i>content</i>	String	Subtitle text.
<i>count</i>	Int	Maximum allowed entries.

Response: None.

4.1.61 Get File Playback State

Protocol Name: *KommanderMsg_GetProgFilePlayState*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.

Response Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.
<i>curPos</i>	Int	Current playback position (seconds).
<i>duration</i>	Int	Total duration (may differ from media file due to trimming).
<i>state</i>	Int	0 = Paused; 1 = Playing.

4.2 Push Notifications to External Endpoints

4.2.1 Update Visible Canvas Information

Protocol Name: *KommanderMsg_UpdateViewCanvasInfo*

Request Parameters:

Parameter	Type	Description
<i>canvasXOffset</i>	Int	X-axis offset of the visible canvas's top-left corner.
<i>canvasYOffset</i>	Int	Y-axis offset of the visible canvas's top-left corner.
<i>canvasViewWidth</i>	Int	Visible canvas width.
<i>canvasViewHeight</i>	Int	Visible canvas height.
<i>canvasWidth</i>	Int	Total canvas width.
<i>canvasHeight</i>	Int	Total canvas height.
<i>CanvasNegativeX</i>	Int	Maximum negative X-coordinate.

Parameter	Type	Description
<i>CanvasNegativeY</i>	Int	Maximum negative Y-coordinate.

Response: None.

4.2.2 Logout

Protocol Name: *KommanderMsg_DisconnectWebSocket*

Request Parameters:

Parameter	Type	Description
<i>discode</i>	Int	Disconnection reason: 0 = Unknown error, 1 = PC-initiated disconnection, 2 = Application termination, 3 = Connection preempted.
<i>deviceid</i>	Int	Preempting device ID (included only if <i>discode</i> is 3).

Response: None.

4.2.3 Update Media Information

Protocol Name: *KommanderMsg_UpdateMediaLibs*

Request Parameters:

Parameter	Type	Description
<i>MediaLibs</i>	Array	Media library data array, including materials, scenarios, and playlists.

MediaLibs Structure:

Parameter	Type	Description
<i>MediaGroup</i>	String	Media group description.

Parameter	Type	Description
<i>MediaGroupType</i>	Int	Group type (see <i>E_MediaGroupType</i>).
<i>data</i>	Array	Media items or subgroups.

data Structure:

1) When *isGroup* is *true*:

Parameter	Type	Description
<i>isGroup</i>	Bool	Indicates a subgroup.
<i>name</i>	String	Subgroup name.
<i>data</i>	Array	Nested media items (recursive structure for <i>MGT_DEFAULT</i> type).

2) When *isGroup* is *false*:

Parameter	Type	Description
<i>isGroup</i>	Bool	Indicates a media item.
<i>type</i>	Int	Media type (see <i>E_MediaType</i>).
<i>name</i>	String	Media name.
<i>id</i>	String	Media GUID.
<i>isInsert</i>	Bool	Indicates interruptible playback (fixed <i>false</i> for <i>MGT_DEFAULT</i>).

Response: None.

4.2.4 Scenario Mode Change

Protocol Name: *KommanderMsg_SetPretidingMode*

Request Parameters:

Parameter	Type	Description
<i>realMode</i>	Bool	<i>true</i> = Real-time mode; <i>false</i> = Pre-edit mode.

Response: None.

4.2.5 Update Resource Thumbnail

Protocol Name: *KommanderMsg_UpdateThumbnailImage*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Resource GUID.
<i>image</i>	String	Base64-encoded image.

Response: None.

4.2.6 Update Lottery State

Protocol Name: *KommanderMsg_UpdateLotteryState*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	Resource GUID.
<i>image</i>	String	Base64-encoded image.

Response: None.

4.2.7 Update Visible Canvas Information

Protocol Name: *KommanderMsg_UpdateViewCavasInfo*

Request Parameters:

Parameter	Type	Description
<i>cavasXOffset</i>	Int	X-axis offset of the visible canvas's top-left corner.
<i>cavasYOffset</i>	Int	Y-axis offset of the visible canvas's top-left corner.
<i>canvasViewWidth</i>	Int	Visible canvas width.
<i>canvasViewHeight</i>	Int	Visible canvas height.

Response: None.

4.2.8 Update User Mode

Protocol Name: *KommanderMsg_UpdateUserMode*

Request Parameters:

Parameter	Type	Description
<i>userMode</i>	Int	Sets the operational mode (primarily for video wall controller configuration).

Response: None.

4.2.9 Update Scenario Usage Flags

Protocol Name: *KommanderMsg_UpdatePrePlanUsingMark*

Request Parameters:

Parameter	Type	Description
<i>outPuttingPrePlanId</i>	String	GUID of the currently outputting scenario.

Parameter	Type	Description
<i>editingPrePlanId</i>	String	GUID of the scenario in pre-edit mode.

Response: None.

4.2.10 Media Library Changed

Protocol Name: *KommanderMsg_MediaLibsChanged*

Request Parameters: None.

Response: None.

4.2.11 Update Black Screen Status

Protocol Name: *KommanderMsg_UpdateBlackScreen*

Request Parameters:

Parameter	Type	Description
<i>blackscreenSwitching</i>	Bool	<i>true</i> = Black screen transition in progress; <i>false</i> = Transition complete.
<i>blackscreen</i>	Bool	<i>true</i> = Black screen enabled.

Response: None.

4.2.12 Set File Mute

Protocol Name: *KommanderMsg_UpdateUserMode*

Request Parameters:

Parameter	Type	Description
<i>userMode</i>	Int	Sets the operational mode (primarily for video wall controller configuration).

Response: None.

Note:

- **Potential Conflict:** The protocol name *KommanderMsg_UpdateUserMode* is reused here, which may indicate a typo. The correct protocol name for muting files should likely be *KommanderMsg_SetProgFileMute*.

4.2.13 Update User Mode

Protocol Name: *KommanderMsg_UpdateUserMode*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.
<i>mute</i>	Bool	<i>true</i> = Mute; <i>false</i> = Unmute.

Response: None.

4.2.14 Update Browser URL

Protocol Name: *KommanderMsg_SetBrowserUrl*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.

Parameter	Type	Description
<i>url</i>	String	New URL to load.

Response: None.

4.2.15 Update State Changed

Protocol Name: *KommanderMsg_UpdateStateChanged*

Request Parameters:

Parameter	Type	Description
<i>isUpdating</i>	Bool	<i>true</i> = Update in progress; <i>false</i> = Idle.

Response: None.

4.2.16 Update Currently Selected File

Protocol Name: *KommanderMsg_SetCurSelectFile*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID. Empty = Deselect.

Response: None.

4.2.17 Update Central Control Public Configuration

Protocol Name: *KommanderMsg_UpdateCentreControlPublicConfig*

Request Parameters:

Parameter	Type	Description
<i>name</i>	String	Base64-encoded image (name).
<i>logo</i>	String	Base64-encoded image (logo).

Response: None.

4.2.18 Update File Playback State

Protocol Name: *KommanderMsg_UpdateProgFilePlayState*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.
<i>curPos</i>	Int	Current playback position (seconds).
<i>duration</i>	Int	Total duration (may differ from media file due to trimming).
<i>state</i>	Int	0 = Paused; 1 = Playing.

Response: None.

4.2.19 Update Schedule

Protocol Name: *KommanderMsg_UpdateSchedule*

Request Parameters:

Parameter	Type	Description
<i>content</i>	String	Command content.
<i>preplanName</i>	String	Scenario name included in the task.

Parameter	Type	Description
<i>taskName</i>	String	Task name.
<i>exectime</i>	String	Execution time (format: YYYY-MM-DD HH:mm:ss).
<i>endTime</i>	String	End time (format: YYYY-MM-DD HH:mm:ss).
<i>bHaveDateLimite</i>	Bool	<i>false</i> = <i>startDate</i> and <i>endDate</i> are effective; <i>true</i> = no date limits.
<i>startDate</i>	String	Start date (format: YYYY-MM-DD).
<i>endDate</i>	String	End date (format: YYYY-MM-DD).
<i>bHaveWeekLimite</i>	Bool	<i>false</i> = <i>specifiedWeek</i> is effective; <i>true</i> = no weekly limits.
<i>specifiedWeek</i>	Int	Bitmask for days: <i>0x01</i> = Monday, <i>0x02</i> = Tuesday, <i>0x04</i> = Wednesday, <i>0x08</i> = Thursday, <i>0x10</i> = Friday, <i>0x20</i> = Saturday, <i>0x40</i> = Sunday.

Response: None.

4.2.20 Update File Volume

Protocol Name: *KommanderMsg_UpdateProgFileVolume*

Request Parameters:

Parameter	Type	Description
<i>id</i>	String	File GUID.
<i>volume</i>	Int	Volume level (1–100).

Response: None.