BOSE Work Rest API App User Guide

Contents

BOSE Work Rest API App

Introduction

The Bose Videobar devices support representational state transfer application programming interface (REST API) for network management and monitoring. This guide provides instructions for enabling and configuring REST API on Videobar devices, and it provides a detailed description of the supported variables and operations. Configuration items and operations are grouped in these categories:

- system
- behavior
- usb
- audio
- camera
- audioframing
- bluetooth
- network (VBI)
- wifi
- telemetry (VBI)

The API Command Reference section provides the following information for each object:

- Name/Description Name of the object and description of its use.
- Actions Actions that can be performed on the object. The action can
- be one or more of the following: get, put, delete, post.
- Range of Values Acceptable values for the object.
- Default Value Default value of the object. This is the value that is used if you revert the device to factory defaults.
  All values are specified as strings.

Trademark Notices

Privacy Information

Your privacy is important to Bose so we've developed a Privacy Policy that covers how we collect, use, disclose, transfer, and store your personal information.
PLEASE READ THIS PRIVACY POLICY CAREFULLY TO UNDERSTAND HOW WE HANDLE YOUR INFORMATION. IF YOU DO NOT AGREE TO THIS PRIVACY POLICY, PLEASE DO NOT USE THE SERVICES.

Enabling and Configuring the REST API

To enable access to the REST API on a device, use the Bose Work Configuration app, the Bose Work Management app, or the Web UI. Access the Network> API settings. Enable API access and specify an API username and password. You will need these API credentials to use any of the REST API commands. Please reference the application user guides for more information.

Testing the REST API

You can test the Videobar REST API by using the Swagger OpenAPI interface that is embedded in the device. To access this interface the Videobar must be connected to an IP network via its wired or WiFi interface, and your host PC must be on the same network or a network that can access the device via HTTPS.
Connect your PC to the Videobar via the USB interface. Start the Bose Work Configuration app and sign in to access admin controls. Choose the Network > API page and click the link:
REST API Documentation (Web UI)
If you are not connected to the device via USB and your PC is on the same network, you can access the REST API via your browser by browsing to the following address:
https://<videobar-ip-address>/doc-api

REST API Commands

The Videobar REST API interface uses command IDs in each of the four HTTP methods supported: get, put, delete, and post.
Below is a description of the four methods followed by a table describing the methods supported for each of the commands.

GET

The "get" method accepts a single command ID or multiple comma-delimited IDs. For example, to get the audio.micMute state, the command ID is 2. The URL is like this:
https://192.168.1.40/api?query=2

The response body is as follows, with a value of "O" indicating the mic is not muted:
{"2": {"status": "success", "value": "0"}}

To query for multiple values, separate multiple command IDs with a comma. For example, you could query for audio.micMute (ID=2) and system.firmwareVersion (ID=l6) li ke this:
https://192.168.1.40/api?query=2,16

Note: Do not include spaces between multiple IDs.
The result would be:

{“2”: {“status”: “success”, “value”: “0”}, “16”: {“status”: “success”, “value”: “1.2.13_fd6cc0e”}}

PUT

A “put” command uses a JSON body format with the key being “data” and the value being ID:value pairs.
For example, to set the audio.loudspeakerVolume (ID=3) to 39, the “https://192.168.1.40/ api” body is:
{“data”:”{“3″:”39″}”}

The response is:
{“3”: {“status”: “success”, “code”: “0xe000”}}

Here is an example setting multiple values:
{“data”:”{“2″:”1″,”3″:”70″}”}

The response is:
{“2”: {“status”: “success”, “code”: “0xe000”}, “3”: {“status”: “success”, “code”: “0xe000”}}

Response “code” values can be any of the following:

- 0xe000 : Success
- 0xe001 : Success – No change in value
- 0xe002 : Error – Invalid property
- 0xe003 : Error – Invalid property value
- 0xe004 : Error – Invalid property action
- 0xe005 : Error – Message malformed
- 0xe006 : Error – Access denied

POST

A “post” is similar to “put” and is used for actions, such as toggle mic mute and speaker volume up/down. You specify the command ID and use an empty string for the value.
For example, to increase the speaker volume one tick, use audio.loudspeakerVolumeUp (ID=4) with the body format like this:
{“data”:”{“4″:””}”}

The response body is:
{“4”: {“status”: “success”, “code”: “0xe000”}}
The possible response “code” values are the same those listed for the PUT command.

DELETE

The “delete” command format is similar to “get”, and the response body is similar to “put”. Using delete will set the value back to its default.
For example, to set the audio.loudspeaker volume (ID=3) to its default value, the URL is like this:
https://192.168.1.40/api?delete=3

The response body is:
{“3”: {“status”: “success”, “code”: “0xe000”}}

You would need to issue a “get” to retrieve the new value, which in this case is 50. For example:

Command:
https://192.168.1.40/api?query=3

Response:
{“3”: {“status”: “success”, “value”: “50”}}
The possible response “code” values are the same those listed for the PUT command

Videobar REST API Command Reference

Name/ Description	Actions	Cmd ID	Range of Values	Default Value
system.reboot Reboots the system.	post	32	N/A	N/A
system.serialNumber Serial number of the device.	get	10	string (17 chars)	ooooooxooooooooxx
system.firmwareVersion Version of the firmware running on the device. This is set automatically on syst em firmware upgrade.	get	16	string (1-16 chars)	0.0.0
system.model Model of this device.	get	D6	string (1-22 chars)	Not set
system.name Name of the device so it can be uniquel y identified.	get put delete	25	string (1-22 chars)	Not set
system.room Room location of the device	get put delete	26	string (0-128 chars)	Not set
system.floor Floor location of the device.	get put delete	27	string (0-128 chars)	Not set
system.building Building location of the device.	get put delete	28	string (0-128 chars)	Not set
system.gpiMuteStatus (VBI) GPI mute status (on/off).	get	C7	110	(Supported in VBI) 0

Name/ Description	Actions	Cmd ID	Range of Values	Default Value
system.maxOccupancy Room maximum occupancy of the device.	get put delete	DF	string (0-128 chars)	Not set
behavior.ethernetEnabled (VBI) Turns on/off the system Ethernet interface.	get put delete	38	110	(Supported in VBI) 1
behavior.bluetoothEnabled Turns on/off the system Bluetooth.	get put delete	3A	110	1
behavior.wifiEnabled Turns on/off the system WiFi.	get put delete	3B	110	1
behavior.hdmiEnabled (VBI) Turns on/off the HDMI.	get put delete	C9	110	(Supported in VBI) 0
usb.connectionStatus USB cable connection status; 0 when disconnected.	get	36	110	0
usb.callStatus Call status from the host connected to USB port of the system.	get	37	110	0
audio.micMute Mutes/unmutes the system microphone.	get put	2	110	0
audio.micMuteToggle Toggles the mute state of the system microphone.	post	15	N/A	N/A
audio.loudspeakerMute Mutes/unmutes the system loudspeaker.	post	34	N/A	N/A

audio.loudspeakerMuteToggle Toggles the mute state of the system lo udspeaker.	post	34	N/A	N/A
audio.loudspeakerVolume Sets the system loudspeaker volume.	get put delete	3	0-100	50
audio.loudspeakerVolumeUp Increases the system loudspeaker volu me by one step.	post	4	N/A	N/A
audio.loudspeakerVolumeDown Decreases the system loudspeaker vol ume by one step.	post	5	N/A	N/A
camera.zoom The camera’s current zoom value.	get put delete	6	1-10	1
camera.pan The camera’s current pan value.	get put delete	7	-10-10	0
camera.tilt The camera’s current tilt value.	get put delete	8	-10-10	0
camera.zoom In Zooms camera in by one step.	post	9	N/A	N/A
camera.zoomOut Zooms camera out by one step.	post	OA	N/A	N/A
camera.pan Left Pans camera left by one step.	post	OB	N/A	N/A
camera.pan Right Pans camera right by one step.	post	oc	N/A	N/A
camera.tiltUp Tilts camera up by one step.	post	OD	N/A	N/A
camera.tiltDown Tilts camera down by one step.	post	OE	N/A	N/A

camera.homePreset Camera home preset in pan tilt zoom order	get put delete	56	<pan><space> <tilt><space> <zoom>	0 01
camera.firstPreset Camera first preset in pan tilt zoom order.	get put delete	57	<pan><space> <tilt><space> <zoom>	0 01
camera.second Preset Camera second preset in pan tilt zoom order.	get put delete	58	<pan><space> <tilt><space> <zoom>	0 01
camera.savePresetHome Saves to the home preset the current PTZ values.	post	12	N/A	N/A
camera.savePresetFirst Saves to the first preset the current PTZ values.	post	17	N/A	N/A
camera.savePresetSecond Saves to the second preset the current PTZ values.	post	18	N/A	N/A

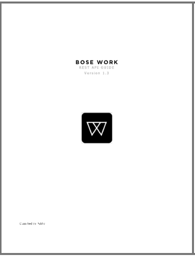
Name/ Description	**Actions**	**Cmd ID**	**Range of Values**	**Default Value**
camera.apply ActivePreset Applies the active preset to the PTZ settings.	post	OF	N/A	N/A
camera.active Preset This is the active preset. Note, at camera start or restart the active preset is set to Home.	get put delete	13	11213	1
camera.state Camera state. When active, camera is streaming video. When inactive, camera is not streaming. When upgrading, camera is upgrading firmware.	get	60	activeI inactiveI upgrading	inactive

autoframing.state Turn on/off the camera autoframing feature.	get put delete	19	110	0
bluetooth.pairingStateToggle Toggle the pairing state from on/off to off/on.	post	C6	N/A	N/A
bluetooth.pairingState Bluetooth pairing state. The on state will allow pairing with the device for a fixed interval. Once the pairing interval is over, the state will change to off.	get put	14	110	0
bluetooth.state Bluetooth and BLE state. The on state will indicate that Bluetooth and BLE are on; the off state will indicate that the Bluetooth and BLE are off.	get	67	110	0
bluetooth.paired Paired device name.	get	6A	string (0-128 chars)	Not set
bluetooth.connected Paired device connection status.	get	6B	110	0
bluetooth.streamState Stream status of Bluetooth.	get	C2	110	0
bluetooth.callState Status of Bluetooth call.	get	6C	110	0
bluetooth.disconnect Disconnect Bluetooth device.	post	E4	11213	N/A
network.dhcpState DHCP state. When DHCP state is on, network will be configured through DHCP. When DHCP state is off, static values are used.	get put delete	74	110	1

network.ip (VBI) Static IP address when DHCP state is o ff.	get put delete	75		(Supported in VBI) 0.0.0. 0
network.state (VBI) State of the Ethernet module.	get	7F	idlel failure! associationl configurati onl readyl disconnect! online	(Supported in VBI) ready

Name/ Description	Actions	Cmd ID	Range of Values	Default Value
network.mac (VBI) MAC address of the LAN interface.	get	80		(Supported in VBI) 00:00 :00:00:00:00
wifi.dhcpState DHCP state. When DHCP state is on, WiFi will be configured through DHCP. When DHCP state is off, static values a re used.	get put delete	AI	110	1
wifi.ip Static IP address when DHCP state is o ff.	get put delete	A2		0.0.0.0
wifi.mac MAC address of the WiFi interface.	get	AC		00:00:00:00:00:00
wifi.state State of the WiFi module.	get	BO	idlel failure! associationl configurati onl readyl disconnect! online	idle
telemetry.peopleCount (VBI) The number of people counted by the c amera autoframing algorithm.	get put delete	DA	0-99	(Supported in VBI) 0
telemetry.peoplePresent (VBI) True when any people have been detected by the camera autoframing alg orithm.	get put delete	DC	110	(Supported in VBI) 0

Documents / Resources

 BOSE WORK REST API GUIDE Version 1.3 Classified as Public	**BOSE Work Rest API App** [pdf] User Guide Work, Rest API, App, Work Rest API App

References

- Privacy Policy | Bose

Manuals+,