

# Stream parameters monitoring with REST API

- [Overview](#)
- [Receiving common stream information](#)
  - [REST API](#)
    - [REST queries and response states](#)
    - [Parameters](#)
- [Receiving stream metrics only](#)
  - [Metrics available](#)
  - [REST API](#)
    - [REST queries and response states](#)
    - [Parameters](#)

## Overview

WCS supports realtime stream publishing and playing parameters monitoring with REST API

## Receiving common stream information

The `/stream/find` REST API query allows to find a stream by filter and get common information about streams found. Since build [5.2.923](#) stream metrics can also be requested.

## REST API

REST query should be HTTP/HTTPS POST request as follows:

- HTTP: `http://streaming.flashphoner.com:8081/rest-api/stream/find`
- HTTPS: `https://streaming.flashphoner.com:8444/rest-api/stream/find`

Where

- `streaming.flashphoner.com` - WCS server address
- `8081` - REST / HTTP port of WCS server
- `8444` - HTTPS port of WCS server

- rest-api - URL mandatory part
- /stream/find - REST query used

## REST queries and response states

REST query	REST query body example	REST response example	Response states
/stream/find	<pre>{   "name": "stream1",   "published": true,   "display": ["metrics"] }</pre>	<pre>[   {     "appKey": "flashStreamingApp",     "sessionId": "/5.129.23.83:51309/95.191.130.39:1935",     "mediaSessionId": "6af0baf1-f014-48d6-b285-c975951a402f",     "name": "stream1",     "published": true,     "hasVideo": true,     "hasAudio": true,     "status": "PUBLISHING",     "sdp": "...",     "audioCodec": "mpeg4-generic",</pre>	<p>200 - Stream data received</p> <p>404 - Stream not found</p> <p>500 - Query parameters error</p>

REST query

REST query body  
example

```
"videoCodec": "H264",  
  
"record":  
false,  
  
"width":  
640,  
  
"height":  
360,  
  
"bitrate":  
0,  
  
"minBitrate":  
0,  
  
"maxBitrate":  
0,  
  
"quality":  
0,  
  
"history":  
false,  
  "gop":  
0,  
  "fps":  
0,  
  
"audioBitrate": 0,  
  
"codecImpl":  
: "",  
  
"transport":  
: "UDP",  
  
"cvoExtension": false,  
  
"createDate":  
:  
1616730493789,  
  
"mediaType":  
:  
"publish",
```

Response states

**REST query****REST query body example**

```
"mediaProvider":  
"Flash",  
  
"metrics":  
{  
  
"VIDEO_SYNC"  
": 43688,  
  
"VIDEO_K_FRAMES"  
": 22,  
  
"AUDIO_SYNC"  
": 43712,  
  
"VIDEO_NACK"  
": 0,  
  
"AUDIO_RATE"  
": 129408,  
  
"AUDIO_LOST"  
": 0,  
  
"VIDEO_LOST"  
": 0,  
  
"VIDEO_CODE"  
C": 119,  
  
"VIDEO_B_FRAMES"  
": 0,  
  
"VIDEO_PLI"  
": 0,  
  
"AUDIO_CODE"  
C": 96,  
  
"VIDEO_RATE"  
": 1187440,  
  
"VIDEO_WIDT"  
H": 640,  
  
"VIDEO_GOP_  
SIZE": 48,  
  
"VIDEO_HEIG"  
HT": 360,
```

**Response states**

REST query	REST query body example		Response states
		<pre data-bbox="815 282 1067 573"> "VIDEO_FPS" : 24,  "VIDEO_P_FR AMES" : 1027 } } ]</pre>	

## Parameters

### Parameter

### Description

### Example

display

Additional stream parameters to be in response

["metrics"]

mediaSessionId

Media session Id

6af0baf1-f014-48d6-b285-c975951a402f

name

Stream name

stream1

published

Published or playing

true

hasVideo

Is there video in the stream?

true

hasAudio

Is there video in the stream?

true

status

Stream status

PUBLISHING

sdp

Stream SDP

audioCodec

Audio codec

mpeg4-generic

videoCodec

Video codec

H264

record

Stream is recording

false

width

Stream width at publishing moment

640

height

Stream height at publishing moment

360

transport

WebRTC stream transport used

UDP

cvoExtension

WebRTC picture orientation extension is used

false

createDate

Stream creation date

1616730493789

mediaType

Media session type: publish or play

publish

mediaProvider

Publishing or playing stream provider (Flash for RTMP)

Flash

minBitrate

Transcoder parameters created by [REST API](#)

maxBitrate

quality

gop

fps

audioBitrate

codecImpl

## Receiving stream metrics only

### Metrics available

Metric	Description
VIDEO_HEIGHT	Video height
VIDEO_WIDTH	Video width
VIDEO_RATE	Video bitrate
VIDEO_SYNC	Video synchronization

VIDEO_FPS	Video frame rate per second
VIDEO_NACK	NACK requests count
VIDEO_PLI	PLI packets count
VIDEO_CODEEC	Video codec
VIDEO_K_FRAMES	Video key frames (I-frames) count
VIDEO_P_FRAMES	Video P frames count
VIDEO_B_FRAMES	Video B frames count
VIDEO_GOP_SIZE	Last group of frames size between key frames
VIDEO_LOST	Video lost packets count
AUDIO_SYNC	Audio synchronization
AUDIO_RATE	Audio bitrate
AUDIO_LOST	Lost audio packets count
AUDIO_CODEEC	Audio codec

## REST API

REST query should be HTTP/HTTPS POST request as follows:

- HTTP: <http://streaming.flashphoner.com:8081/rest-api/stream/metrics>
- HTTPS: <https://streaming.flashphoner.com:8444/rest-api/stream/metrics>

Where

- streaming.flashphoner.com - WCS server address
- 8081 - REST / HTTP port of WCS server
- 8444 - HTTPS port of WCS server
- rest-api - URL mandatory part
- /stream/metrics - REST query used

## REST queries and response states

REST query	REST query body example	REST response example	Response states

## REST query

/stream/metrics

```
{  
  "mediaSessionId" :  
  "72b76eb8-  
3c01-47f3-  
b308-  
a0516d51249  
b"  
}
```

or

```
{  
  "name" :  
  "testStream"  
}
```

## Response states

200 - Stream metrics received

404 - Stream not found

```
{  
  "VIDEO_B_FRAMES" : 0,  
  "VIDEO_WIDTH" : 1920,  
  "VIDEO_SYNC" :  
  1583463093448,  
  "AUDIO_RATE" : 31832,  
  "VIDEO_PLI" : 0,  
  "VIDEO_HEIGHT" : 1080,  
  "AUDIO_SYNC" :  
  1583463093415,  
  "VIDEO_FPS" : 36,  
  "AUDIO_CODEC" : 111,  
  "VIDEO_P_FRAMES" :  
  3989,  
  "VIDEO_RATE" : 684352,  
  "VIDEO_CODEC" : 119,  
  "VIDEO_K_FRAMES" : 173,  
  "VIDEO_NACK" : 1,  
  "VIDEO_LOST" : 1,  
  "AUDIO_LOST"
```

REST query	REST query body example	<pre>": 130 }</pre>	Response states
------------	-------------------------	---------------------	-----------------

## Parameters

Parameter	Description	Example
mediaSessionId	Media session identifier	72b76eb8-3c01-47f3-b308-a0516d51249b
name	Stream name	testStream