Cinegy Air 26.2 HTTP API Manual

© Cinegy GmbH

Built 18.11.2026

Table of Contents

Cinegy Air HTTP API to Control Layers (Devices) 1
1 Main URL Definition 1
2 Commands 2
3 Status 3
4 List and Items 4
Cinegy Air HTTP API to Control CG Engine 5
5 Concept and Common Model 5
6 Technical Details 6
7 CG Variables 8
8 CG Web Call 9
Telemetry 10
9 Metrics 10
10 SRT connections statistics 11

Cinegy Air HTTP API to Control Layers (Devices)

Chapter 1 Main URL Definition

Main URL is `http://NAME:PORT/DEVICE`

where:

- **NAME** is the server name or IP address,
- **PORT** is 5521 plus the instance number,
- **DEVICE** is the device name, possible values are:
  - video
  - titler_0
  - logo
  - cg_0 … cg_8
  - cg_logo
  - gfx_0 … gfx_8
  - gfx_logo
  - gfx_0 … gfx_8
  - gfx_logo
  - audio

All names are in lowercase letters, uppercase is "macro". Can be dependent on the engine version and configuration (gfx layers, ovl layers).



The audio device is designed for overlaying audio over video file.

Chapter 2 Commands

Command URL is `http://NAME:PORT/DEVICE/command`

The request method should be **POST**, the request body should be XML with command request and **Content-Type** in the request header should be **text/xml**.

General request XML is:

```
<?xml version="1.0" encoding="utf-8"?>
<Request>
  <!-- Here should be one or more command tags -->
</Request>
```

Request tags are:

```
<Cue Id="{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}"/>
<StartCued />
<GPI Commercial="y" Preroll="10" />
<GPI Commercial="n" Preroll="10" />
<Event Device="" Cmd="" Op1="" Op2="" Op3=""
      Name="" Description="" ThirdPartyId=""
      Id="{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}">
  <Op1></Op1>
  <Op2></Op2>
  <Op3></Op3>
</Event>
<SetOutput State="Normal|Black|Bypass|Clean"/>
<ExitSceneLoop/>
<SetVariables>
  <Variable Name="Name1" Value="Val1"/>
  <Variable Name="Name2" Value="Val2"/>
  <!-- More like above -->
</SetVariables>
```

General response is:

```
<?xml version="1.0" encoding="utf-8"?>
<Reply Sucess="n" Status="Error">
  <!-- Here should be one or more response tags like following -->
  <Result Sucess="y" Status="OK" />
  <Result Sucess="n" Status="0x8XXXXXXX" />
</Reply>
```

Upper success is "yes" only if all commands were successfully executed. Syntax error can lead to the partial command execution. Bad command result does not stop the execution.

Chapter 3 Status

Status URL is `http://NAME:PORT/DEVICE/status`

Method is **GET**.

Result XML is:

```
<?xml version="1.0" encoding="utf-8"?>
<Status>
  <Active Id="{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}"/>
  <Cued Id="{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}"/>
  <License State="Licensed|Not Licensed|Demo"/>
  <Output State="Normal|Black|Bypass|Clean"/>
  <Client Connected="y|n" Identity="IdentityString"/>
</Status>
```

Status for active item URL is `http://NAME:PORT/DEVICE/status/active`

Status for cued item URL is `http://NAME:PORT/DEVICE/status/cued`

Method is GET.

Result XML is the same as for items list request.

Chapter 4 List and Items

List URL is `http://NAME:PORT/DEVICE/list`

Method is **GET**.

Result XML is:

```
<?xml version="1.0" encoding="utf-8"?>
<List>
  <Item Id="{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}"
        Name="Name" Description="Desc" ThirdPartyId="3rd ID" SubtitleId="CCC"
        ProxyProgress="0" ProxyError="y"
        ScheduledAt="yyyy-mm-ddThh:mm:ss.mssZ" Duration="hh:mm:ss.mss" IsEmpty="y"
        Clocked="y" ManualEnd="y" JipSkipable="y"
        LoopBody="y" LoopStart="y" ManualStart="y"
        TrickLoop="y" />
  <!-- More like above -->
</List>
```

Item URL is `http://NAME:PORT/DEVICE/list/XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX`

Method is **GET**.

Result XML is:

```
<?xml version="1.0" encoding="utf-8"?>
<Item Id="{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}"
      Name="Name" Description="Desc" ThirdPartyId="3rd ID" SubtitleId="CCC"
      AvailId="AA" EpgId="EEEE"
      ProxyProgress="0" ProxyError="y"
      ScheduledAt="yyyy-mm-ddThh:mm:ss.mssZ" Duration="hh:mm:ss.mss" IsEmpty="y"
      Clocked="y" ManualEnd="y" JipSkipable="y"
      LoopBody="y" LoopStart="y" ManualStart="y"
      TrickLoop="y" />
```

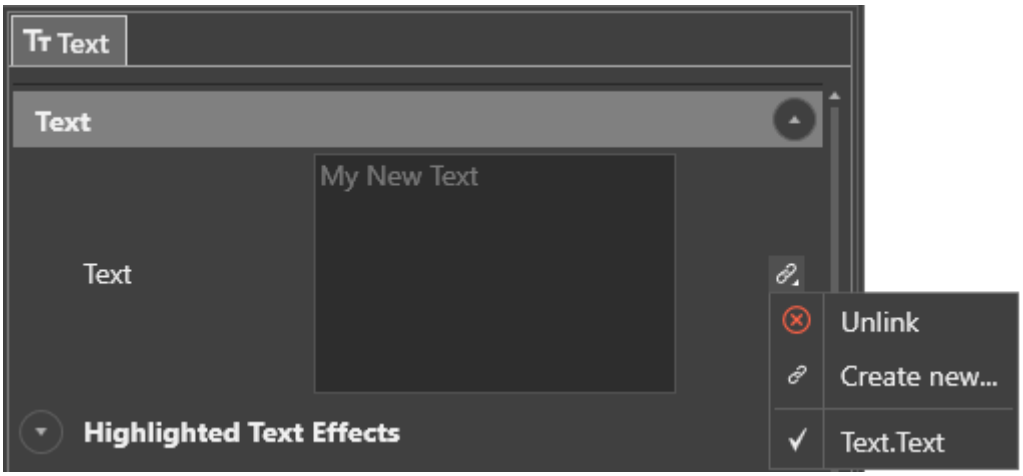
All "yes" values are present only if they are "yes", and absent if they are "no". Absent proxy status means that proxy is disabled or not relevant for this device. Other absent values actually mean that they are empty strings. "SheduledAt" and "Duration" are always filled, the first is just approximation made by plain list analysis but actual time depends on loops and manual status.

Cinegy Air HTTP API to Control CG Engine

Chapter 5 Concept and Common Model

Cinegy Title allows you to use not only static data and texts in your CG templates, but enables you to control such data dynamically, when a Cinegy Title template is already activated and being sent on air.

To enable this feature, on the template preparation stage link the parameter or data you want to be able to change dynamically, to the "Variable" in the Cinegy Title Editor application:

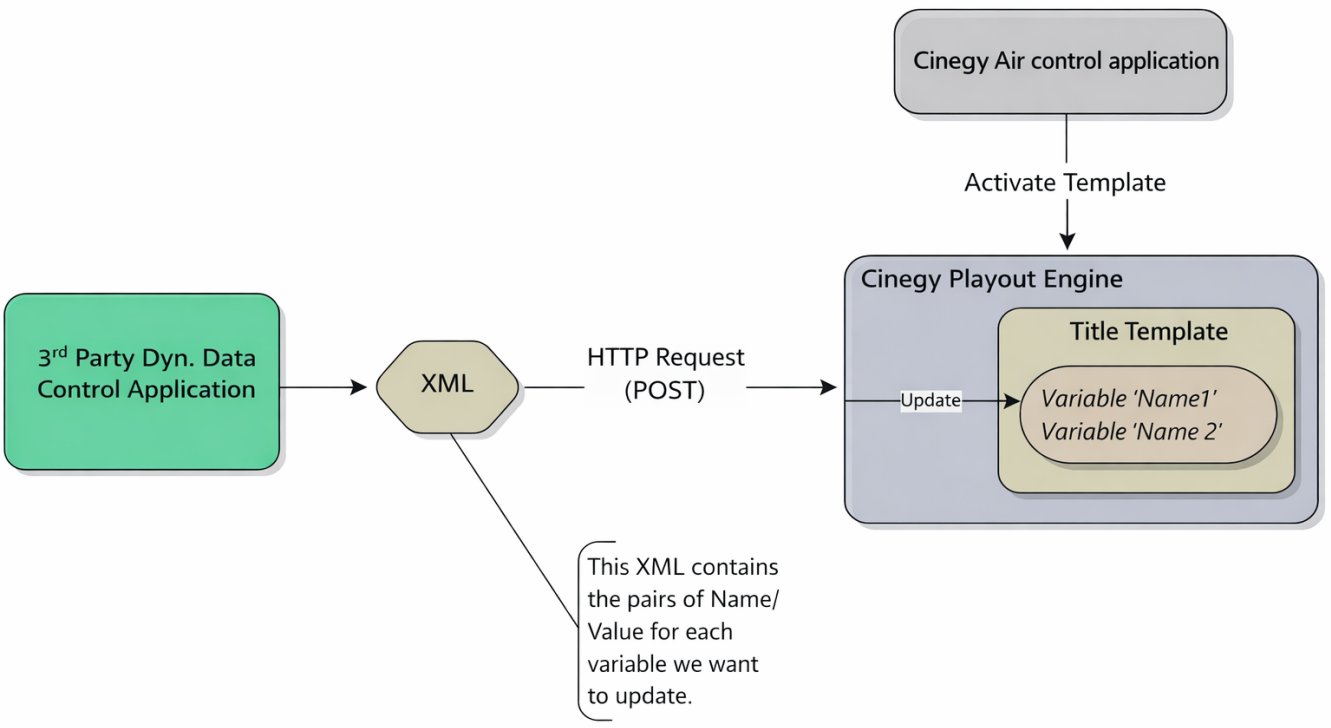


When the Cinegy Title template is created with variables, it may be started in a usual way, by using the secondary event in the Cinegy Air control application. Then any "Variable" can be dynamically updated via a simple web request (which may be initiated from any third-party application written on C#, or Visual Basic). A new value (text, color, etc.) will then be applied to the template objects, which are linked to this variable.

Chapter 6 Technical Details

As described above, the so-called 'HTTP request' should be sent to the Cinegy Playout engine in order to alter the variable (or variable list) in the Cinegy Title template.

The following diagram shows the common process graphically:



The third-party application then sends the HTTP 'POST' request to the Cinegy Playout engine, where the Cinegy Title template is activated, sending the XML text as 'body'. This XML contains the Name–Value pairs for the variables that need to be updated.

The HTTP request is sent to the following URL:

`http://<machine-name>:<port>/postbox`

where

<machine-name> – name or IP address of the Cinegy Air server;

<port> - port number that defines the Cinegy Playout engine instance number (as more than one service can be started on the same Cinegy Air server) and is calculated according to the following table:

Service Instance Number	Port
0	5521
1	5522
…	…
15	5536

The XML format is like following:

```
<?xml version="1.0" encoding="utf-8"?>
<PostRequest>
  <SetValue Name="TextVar" Type="text" Value="My New Text"/>
  ...
</PostRequest>
```

The example above will set the text 'My New Text' to the variable 'TextVar'. Multiple Variables can be updated in the single HTTP request (more <SetValue> tags should be added into the XML in this case).

Chapter 7 CG Variables

URL is `http://NAME:PORT/postbox`

Method is **POST**.

Request XML is:

```
<PostRequest>
  <SetValue Name="Name1" Type="text|file|float|int|bool|color|trigger" Value="Val1"/>
  <SetValue Name="Name2" Type="text|file|float|int|bool|color|trigger" Value="Val2"/>
  <SetValue Name="Name3" Type="text|file|float|int|bool|color|trigger" Value="Val3"/>
</PostRequest>
```

Response XML is:

```
<PostReply Success="1|0" Status="OK|Error">
  <SetValue Name="Name1" Success="1|0" Status="OK|Syntax error"/>
  <SetValue Name="Name2" Success="1|0" Status="OK|Syntax error"/>
  <SetValue Name="Name3" Success="1|0" Status="OK|Syntax error"/>
</PostReply>
```

Chapter 8 CG Web Call

URL is `http://NAME:PORT/webcallcg`

Method is **POST**.

XML transparently goes to the Cinegy Playout engine and response XML also transparently returns as the result.

Telemetry

Chapter 9 Metrics

URL is `http://NAME:PORT/metrics`

Method is **GET**.

Response XML is:

```
<Metrics StartAt="yyyy-mm-ddThh:mm:ssZ" >
  <At Time="yyyy-mm-ddThh:mm:ssZ" DroppedCount="0" OutputCount="25" NoInputSignal="0"
      AverageReadTime="0.0" ReadErrorRate="0" Heartbeat="650"/>
  <!-- More like above -->
</Metrics>
```

All telemetry values in the **At** tag are values on the interval between the previous and current **At** tag. Currently this interval is 1 second. Maximum history length is 60 items, so it is 1 minute.

DroppedCount – maximum number of frames dropped on output devices.

OutputCount – number of frames sent to the output.

NoInputSignal – number of frames that were not received from current active video input (for live items).

AverageReadTime – average value of milliseconds per frame on main file reading (both heads).

ReadErrorRate – percentage of reading errors.

Heartbeat – how long we wait till the next second in telemetry thread (typical value greatly depends on CPU power and load), may be used to detect total Cinegy Air slowdown.

Chapter 10 SRT connections statistics

URL is http://NAME:PORT/metrics/srt

Method is **GET**.

Response XML is:

```
<Connections>
  <Connection Key="192.168.0.83:51347 <- 5471" MsTimeStamp="52585"
              PktSent="16248" PktRecv="0" PktSndLoss="0" PktRcvLoss="0"
              PktRetrans="0" PktRcvRetrans="0" PktSentACK="0"
              PktRecvACK="4377" PktSentNAK="0" PktRecvNAK="0"
              MbpsSendRate="3.362" MbpsRecvRate="0.000"
              UsSndDuration="41508826" PktReorderDistance="0"
              PktRcvAvgBelatedTime="0.000" PktRcvBelated="0"
              PktSndDrop="0" PktRcvDrop="0" PktRcvUndecrypt="0"
              ByteSent="22097280" ByteRecv="0" ByteRcvLoss="0"
              ByteRetrans="0" ByteSndDrop="0" ByteRcvDrop="0"
              ByteRcvUndecrypt="0" UsPktSndPeriod="2587"
              PktFlowWindow="8153" PktCongestionWindow="8192"
              PktFlightSize="5" MsRTT="0.265" MbpsBandwidth="3.648"
              ByteAvailSndBuf="0" ByteAvailRcvBuf="0" MbpsMaxBW="4.205"
              ByteMSS="1500" PktSndBuf="0" ByteSndBuf="0" MsSndBuf="0"
              MsSndTsbPdDelay="120" PktRcvBuf="0" ByteRcvBuf="0"
              MsRcvBuf="0" MsRcvTsbPdDelay="120" PktReorderTolerance="0"
              PktSentUnique="16248" PktRecvUnique="0" ByteSentUnique="22097280"
              ByteRecvUnique="0"/>
  <!-- More like above -->
</Connections>
```

Every present connection has its own Connection element.

Key – identifies connection and data flow direction, consists of remote IP, remote port, local port, and data flow indicator (->/<-).

The rest of the parameters are from SRT library statistics https://github.com/Haivision/srt/blob/master/docs/API/statistics.md



We do not use accumulated statistics as we never reset statistics values.