

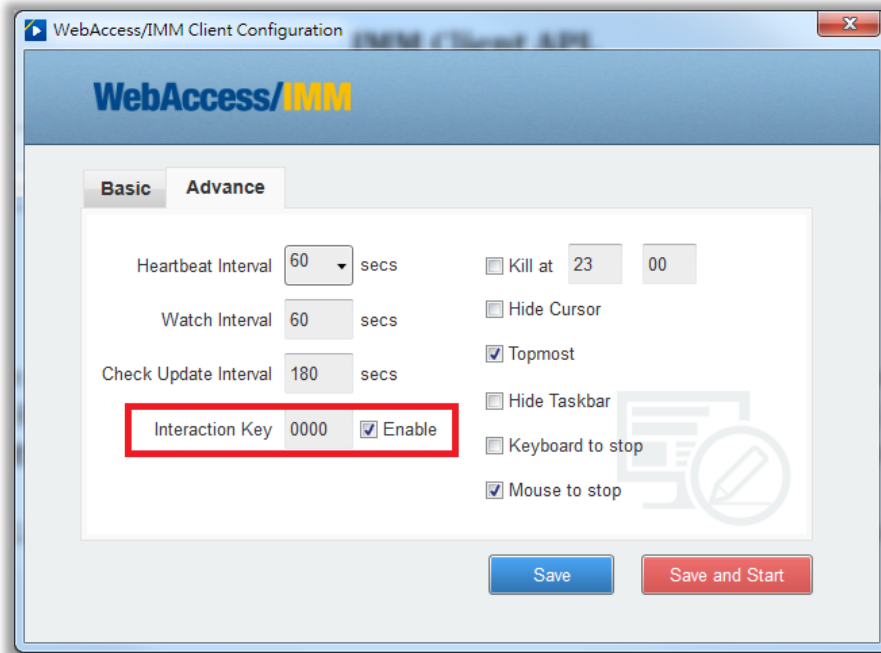
IMM Client API

v1.0	2017-08-22	Ethan Pai
First version		
Apply to IMM-1.7.2		
v1.1	2017-11-14	Ethan Pai
<ol style="list-style-type: none"> 1. Add major property to partition elements in Get Schedule Information API response. 2. Add collision and behavior to playback relative APIs. 3. Add Play Stream API and Play YouTube API. 		
Apply to IMM-1.7.4		
v1.2	2017-11-14	Ethan Pai
1. Add Sensor API and Sensor Status API.		
Apply to IMM-1.7.6		
v1.3	2018-03-09	Ethan Pai
1. Add duration argument to Start Program API.		
Apply to IMM-1.8.2_20180309		
v1.4	2018-05-21	Ethan Pai
<ol style="list-style-type: none"> 1. Add Set Configuration and Get Configuration API. 2. Get Program Information and Get Sensor Status API will require I-Key. 		
Apply to SinageCMS-1.8.4_20180521		
v1.4.1	2018-06-12	Ethan Pai
<ol style="list-style-type: none"> 1. Add position property to program elements and partition elements in Get Schedule Information API and Get Program Information API. 		
[20180612_2] 1. Correct Set Configuration and Get Configuration API instructions.		
Apply to SinageCMS-1.8.4_20180612		
v1.5	2018-06-29	Ethan Pai
1. Push Sensor Status API can trigger multiple events.		
Apply to SinageCMS-1.8.4_20180629		

Introduction

The customer applications can use IMM client API through HTTP port 8085 to inquire program and media information, scheduling and status. It can also be used to control the IMM client play

User can set the IMM client status to enable this interactive function or not to avoid undesired external controlling process. The default setting of the IMM interactive API is “Enabled” and default passcode is “0000”. This setting only applies to the interactive function but not the query function. The client setting looks below:



User can use UDP port 8083 to send out broadcast packet to detect IMM client in the given network domain. Details please refer to the “Probe IMM Clients API” section.

APIs			
Name	Protocol/Port	I-Key	Description
Probe IMM Clients	UDP / 8083	N	Detect IMM clients in the network domain
Get Schedule Information	HTTP / 8085	N	Acquire scheduled program, play area, play list and media information
Get Program Information	HTTP / 8085	Y	Acquire information of ongoing program
Get Program Preview	HTTP / 8085	N	Acquire preview of the program
Get Media Preview	HTTP / 8085	N	Acquire preview of media
Play	HTTP / 8085	Y	Start playing
Stop	HTTP / 8085	Y	Stop playing
Start Program	HTTP / 8085	Y	Stop playing scheduled program and start playing designated program
Start Schedule	HTTP / 8085	Y	Resume schedule
Start Partition	HTTP / 8085	Y	start playing of designated partition
Stop Partition	HTTP / 8085	Y	Stop playing of designated partition
Set Label	HTTP / 8085	Y	Set text label to a text block
Play Media	HTTP / 8085	Y	Play media in a designated partition
Play Web	HTTP / 8085	Y	Play website in a designated partition
Play Stream	HTTP / 8085	Y	Play streaming media in a designated partition
Play YouTube	HTTP / 8085	Y	Play YouTube in a designated partition

Get Event Script	HTTP / 8085	N	Acquire event script
Push Sensor Status	HTTP / 8085	Y	Acquire output from sensor and save in the memory
Get Sensor Status	HTTP / 8085	Y	Acquire sensor status
Get Configuration	HTTP / 8085	Y	Acquire configuration of the IMM client
Set Configuration	HTTP / 8085	Y	Set configuration and restart IMM client

I-Key = require enabling interaction function and interaction key.

Probe IMM Clients

Broadcast UTF-8 string "IMM" via UDP port 8083 for probing IMM clients on local area networks.

Description

Detect IMM clients in the network domain

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response code="0" message="Success">
  <imm_client>
    <version>WebAccess/IMM Client 1.7.2</version>
    <id>112233445566</id>
    <name>Demo Client</name>
    <status>playing</status>
    <schedule>201708101346 Program</schedule>
    <program> Program </program>
    <free_space>1.17 TB</free_space>
    <server_ip>127.0.0.1</server_ip>
    <server_port>80</server_port>
    <server_status>ready</server_status>
    <mqtt_status>connected</mqtt_status>
  </imm_client>
</response>
```

Get Schedule Information

GET interaction/schedule.xml

Description

Acquire scheduled program, play area, play list and media information

Example

http://127.0.0.1:8085/interaction/schedule.xml

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response code="0" message="Success">
  <schedule name="Program" activation_time="2017-08-10 13:46" creation_time="2017-08-10 13:46:32">
    <program id="13" name="PDF" position="0,0,1920,1080"/>
    <program id="35" name="Program" position="0,0,1920,1080">
      <partition id="1" name="interaction_1" type="interaction" major="true"
default_playlist_id="2" position="0,0,500,600">
        <playlist id="1" name="Playlist_1">
          <media id="403" name="CF0.mp4" type="video" content="00000403.mp4" />
          <media id="442" name="IM0.jpg" type="image" content="00000442.jpg" />
        </playlist>
      </partition>
    </program>
  </schedule>
</response>
```

```
<media id="422" name="SWF0.swf" type="flash" content="00000422.swf" />
<media id="" name="" type="web" content="http://www.google.com/" />
</playlist>
<playlist id="2" name="Playlist_2">
  <media id="225" name="CF1.mp4" type="video" content="00000225.mp4" />
  <media id="452" name="IM1.jpg" type="image" content="00000452.jpg" />
  <media id="423" name="SWF1.swf" type="flash" content="00000423.swf" />
  <media id="" name="" type="web" content="http://www.mobile01.com/" />
</playlist>
</partition>
<partition id="2" name="TEX BLOCK" type="label" text="TEXT" position="0,800,60,200"/>
</program>
</schedule>
</response>
```

Possible Errors

ScheduleNotFound

Get Program Information

GET interaction/program.xml?interaction_key=<interaction_key>

Description

Acquire information of ongoing program

Example

http://127.0.0.1:8085/interaction/program.xml

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response code="0" message="Success">
  <program id="35" name="Program" is_breaking="false" position="0,0,1920,1080">
    <partition id="1" name="interaction_1" type="interaction" current_playlist_id="2"
position="0,0,500,600"/>
    <partition id="2" name="Text block" type="label" text="TEXT" position="0,800,60,200"/>
  </program>
</response>
```

Possible Errors

ClientNotPlayState

ProgramNotFound

Get Program Preview

GET interaction/program/<program_id>.jpg

Description

Acquire preview of the program

Example

http://127.0.0.1:8085/interaction/program/35.jpg

Get Media Preview

GET interaction/media/<media_id>.jpg

Description

Acquire preview of media. Only video or image has review

Example

http://127.0.0.1:8085/interaction/media/403.jpg

Play

GET interaction/action?type=play
&interaction_key=<interaction_key>

Description

Start playing. If the IMM client is playing, then it will return success. If the client is not playing, then it will initiate the play of the IMM client.

Example

```
http://127.0.0.1:8085/interaction/action?type=play&interaction_key=0000
```

Possible Return Values

Success

AuthenticationError

InteractionDisabled

ClientNotStandbyState

Stop

```
GET interaction/action?type=stop  
&interaction_key=<interaction_key>
```

Description

Stop playing

Example

```
http://127.0.0.1:8085/interaction/action?type=stop&interaction_key=0000
```

Possible Return Values

Success

AuthenticationError

InteractionDisabled

ClientNotPlayState

Start Program

```
GET interaction/action?type=start_program  
&interaction_key=<interaction_key>  
&program_id=<program_id>&
```

[Optional]

```
&behavior=repeat|at_most_once|exactly_once|at_least_once
```

```
(obsolete)&force=false|true
```

```
&collision=ignore|force|queue
```

```
&queue_timeout=<queue_timeout_in_seconds>
```

&duration=<duration_in_seconds>

Description

Stop playing scheduled program and start playing designated program. If the designated program is playing, then this command will restart the program.

[Optional]

behavior (default=repeat)

repeat: Not to block the next partition command while playing. After playing not to switch back to default playing list.

at_most_once: Not to block the next partition command while playing. After finish switch back to the default playing list and continue the planned program.

exactly_once: Block the next partition command while playing. Switch back to the default playing list and continues the planned program.

at_least_once: Block the partition command while playing, not to switch to the default playing list.

(obsolete)force (default=false)

true: Ignore the previous partition command "behavior" parameter and execute the command.

collision (default=ignore)

ignore: When the execution is blocked by the previous command, abort execution.

force: When the execution is blocked by the previous command, force execution.

queue: When the execution is blocked by the previous command, que to execute after the previous command. There's only one command allowed in the que. When two commands enter queuing, then the later one replaces the previous one.

queue_timeout (default=5)

seconds: The queue time out time. When the queue command is out of this time, then the command will not be executed.

duration (default=10)

seconds: Play time

Example

http://127.0.0.1:8085/interaction/action?type=start_program&interaction_key=000&program_id=13&behavior=exactly_once&collision=queue&queue_timeout=10&duration=10

Possible Return Values

Success

ActionQueued

AuthenticationError

InteractionDisabled

ProgramNoInteractionPartition

ProgramNotFound

LastPlaybackNotFinished

Start Schedule

```
GET interaction/action?type=start_schedule
&interaction_key=<interaction_key>
```

Description

Resume schedule

If the planned schedule is already playing, then return success. If the designated program is not playing, then this command will initiate the program. If the client is not in the playing mode, the command initiates the playing.

Example

```
http://127.0.0.1:8085/interaction/action?type=start_schedule&interaction_key=
0000
```

Possible Return Values

Success

AuthenticationError

InteractionDisabled

Start Partition

```
GET interaction/action?type=start_partition
&interaction_key=<interaction_key>
&program_id=<program_id>
&partition_id=<partition_id>
&playlist_id=<playlist_id>
```

[Optional]

```
&behavior=repeat|at_most_once|exactly_once|at_least_once
```

```
(obsolete)&force=false|true
```

```
&collision=ignore|force|queue
```

```
&queue_timeout=<queue_timeout_in_seconds>
```

Description

start playing of designated partition

[Optional]

behavior (default=repeat)

repeat: Not to block the next partition command while playing. After playing not to switch back to default playing list.

at_most_once: Not to block the next partition command while playing. After finish switch back to the default playing list and continue the planned program.

exactly_once: Block the next partition command while playing. Switch back to the default playing list and continues the planned program.

at_least_once: Block the partition command while playing, not to switch to the default playing list.

(**obsolete**)force (default=false)

true: Ignore the previous partition command "behavior" parameter and execute the command.

collision (default=ignore)

ignore: When the execution is blocked by the previous command, abort execution.

force: When the execution is blocked by the previous command, force execution.

queue: When the execution is blocked by the previous command, que to execute after the previous command. There's only one command allowed in the que. When two commands enter queuing, then the later one replaces the previous one.

queue_timeout (default=5)

seconds: The queue time out time. When the queue command is out of this time, then the command will not be executed.

Example

```
http://127.0.0.1:8085/interaction/action?type=start_partition&interaction_key=0000&program_id=35&partition_id=1&playlist_id=1&behavior=exactly_once&collision=queue&queue_timeout=10
```

Possible Return Values

Success

ActionQueued

AuthenticationError

InteractionDisabled

ClientNotPlayState

ProgramNotFound

ParameterNoProgramId

ProgramNotMatched

ParameterNoPartitionId

PartitionNotFound

ParameterNoPlaylistId
PlaylistNotFound
LastPlaybackNotFinished
ParameterErrBehavior

Stop Partition

GET interaction/action?type=stop_partition

&interaction_key=<interaction_key>

&program_id=<program_id>

&partition_id=<partition_id>

[Optional]

(**obsolete**)&force=false|true

&collision=ignore|force|queue

&queue_timeout=<queue_timeout_in_seconds>

Description

Stop playing of designated partition

[Optional]

(**obsolete**)force (default=false)

true: Ignore the previous partition command "behavior" parameter and execute the command.

collision (default=ignore)

ignore: When the execution is blocked by the previous command, abort execution.

force: When the execution is blocked by the previous command, force execution.

queue: When the execution is blocked by the previous command, que to execute after the previous command. There's only one command allowed in the que. When two commands enter queuing, then the later one replaces the previous one.

queue_timeout (default=5)

seconds: The queue time out time. When the queue command is out of this time, then the command will not be executed.

Example

http://127.0.0.1:8085/interaction/action?type=stop_partition&interaction_key=0000&program_id=35&partition_id=1&collision=queue&queue_timeout=10

Possible Return Values

Success

ActionQueued

AuthenticationError
InteractionDisabled

ClientNotPlayState
ProgramNotFound
ParameterNoProgramId
ProgramNotMatched
ParameterNoPartitionId
PartitionNotFound
LastPlaybackNotFinished

Set Label

```
GET http://127.0.0.1:8085/interaction/action?type=set_label
&interaction_key=<interaction_key>
&program_id=<program_id>
&partition_id=<partition_id>
&text=<text>
```

Description

Set text label to a text block

Example

```
http://127.0.0.1:8085/interaction/action?type=set_label&interaction_key=0000&
program_id=35&partition_id=1&text=HelloWorld!
```

Possible Return Values

Success

AuthenticationError
InteractionDisabled

ClientNotPlayState
ProgramNotFound
ParameterNoProgramId
ProgramNotMatched
ParameterNoPartitionId
PartitionNotFound
ParameterNoText

Play Media

```
GET http://127.0.0.1:8085/interaction/action?type=play_media
&interaction_key=<interaction_key>
&program_id=<program_id>
&partition_id=<partition_id>
&media_id=<media_id>
```

[Optional]

```
&behavior=repeat|at_most_once|exactly_once|at_least_once
(obsolete)&force=false|true
&collision=ignore|force|queue
&queue_timeout=<queue_timeout_in_seconds>
&duration=<duration_in_seconds>
```

Description

Play media in a designated partition

[Optional]

behavior (default=repeat)

- repeat: Not to block the next partition command while playing. After playing not to switch back to default playing list.
- at_most_once: Not to block the next partition command while playing. After finish switch back to the default playing list and continue the planned program.
- exactly_once: Block the next partition command while playing. Switch back to the default playing list and continues the planned program.
- at_least_once: Block the partition command while playing, not to switch to the default playing list.

(**obsolete**)force (default=false)

- true: Ignore the previous partition command "behavior" parameter and execute the command.

collision (default=ignore)

- ignore: When the execution is blocked by the previous command, abort execution.
- force: When the execution is blocked by the previous command, force execution.
- queue: When the execution is blocked by the previous command, que to execute after the previous command. There's only one command allowed in the que. When two commands enter queuing, then the later one replaces the previous one.

queue_timeout (default=5)

- seconds: The queue time out time. When the queue command is out of this time, then the command will not be executed.

duration (default=10)

Seconds: Playing time, not application to video.

Example

```
http://127.0.0.1:8085/interaction/action?type=play_media&interaction_key=0000
&program_id=35&partition_id=1&media_id=422&behavior=exactly_once&collision=qu
eue&queue_timeout=10&duration=10
```

Possible Return Values

Success

ActionQueued

AuthenticationError

InteractionDisabled

Failure

ClientNotPlayState

ProgramNotFound

ParameterNoProgramId

ProgramNotMatched

ParameterNoPartitionId

PartitionNotFound

ParameterNoUrl

LastPlaybackNotFinished

ParameterErrBehavior

Play Web

```
GET http://127.0.0.1:8085/interaction/action?type=play_web
&interaction_key=<interaction_key>
&program_id=<program_id>
&partition_id=<partition_id>
&url=<url>
```

[Optional]

```
&behavior=repeat|at_most_once|exactly_once|at_least_once
```

```
(obsolete)&force=false|true
```

```
&collision=ignore|force|queue
```

```
&queue_timeout=<queue_timeout_in_seconds>
```

```
&duration=<duration_in_seconds>
```

Description

Play website in a designated partition

[Optional]

behavior (default=repeat)

- repeat: Not to block the next partition command while playing. After playing not to switch back to default playing list.
- at_most_once: Not to block the next partition command while playing. After finish switch back to the default playing list and continue the planned program.
- exactly_once: Block the next partition command while playing. Switch back to the default playing list and continues the planned program.
- at_least_once: Block the partition command while playing, not to switch to the default playing list.

(**obsolete**)force (default=false)

- true: Ignore the previous partition command "behavior" parameter and execute the command.

collision (default=ignore)

- ignore: When the execution is blocked by the previous command, abort execution.
- force: When the execution is blocked by the previous command, force execution.
- queue: When the execution is blocked by the previous command, que to execute after the previous command. There's only one command allowed in the que. When two commands enter queuing, then the later one replaces the previous one.

queue_timeout (default=5)

- seconds: The queue time out time. When the queue command is out of this time, then the command will not be executed.

duration (default=10)

- seconds: Playing time

Example

```
http://127.0.0.1:8085/interaction/action?type=play_web&interaction_key=0000&program_id=35&partition_id=1&url=http://www.google.com/&behavior=exactly_once&collision=queue&queue_timeout=10&duration=10
```

Possible Return Values

Success

ActionQueued

AuthenticationError

InteractionDisabled

Failure

ClientNotPlayState

ProgramNotFound

ParameterNoProgramId

ProgramNotMatched
ParameterNoPartitionId
PartitionNotFound
ParameterNoUrl
LastPlaybackNotFinished
ParameterErrBehavior

Play Stream

```
GET http://127.0.0.1:8085/interaction/action?type=play_stream
&interaction_key=<interaction_key>
&program_id=<program_id>
&partition_id=<partition_id>
&url=<url>
```

[Optional]

```
&behavior=repeat|at_most_once|exactly_once|at_least_once
(obsolete)&force=false|true
&collision=ignore|force|queue
&queue_timeout=<queue_timeout_in_seconds>
&duration=<duration_in_seconds>
```

Description

Play streaming media in a designated partition

[Optional]

behavior (default=repeat)

repeat: Not to block the next partition command while playing. After playing not to switch back to default playing list.

at_most_once: Not to block the next partition command while playing. After finish switch back to the default playing list and continue the planned program.

exactly_once: Block the next partition command while playing. Switch back to the default playing list and continues the planned program.

at_least_once: Block the partition command while playing, not to switch to the default playing list.

(**obsolete**)force (default=false)

true: Ignore the previous partition command "behavior" parameter and execute the command.

collision (default=ignore)

ignore: When the execution is blocked by the previous command, abort execution.

force: When the execution is blocked by the previous command, force execution.

queue: When the execution is blocked by the previous command, que to execute after the previous command. There's only one command allowed in the que. When two commands enter queuing, then the later one replaces the previous one.

queue_timeout (default=5)

seconds: The queue time out time. When the queue command is out of this time, then the command will not be executed.

duration (default=10)

seconds: Playing time

Example

```
http://127.0.0.1:8085/interaction/action?type=play_stream&interaction_key=0000&program_id=35&partition_id=1&url=rtsp://127.0.0.1:8554/movie&behavior=exactly_once&collision=queue&queue_timeout=10&duration=10
```

Possible Return Values

Success

ActionQueued

AuthenticationError

InteractionDisabled

Failure

ClientNotPlayState

ProgramNotFound

ParameterNoProgramId

ProgramNotMatched

ParameterNoPartitionId

PartitionNotFound

ParameterNoUrl

LastPlaybackNotFinished

ParameterErrBehavior

Play YouTube

```
GET http://127.0.0.1:8085/interaction/action?type=play_youtube
&interaction_key=<interaction_key>
&program_id=<program_id>
&partition_id=<partition_id>
&url=<url>
```


[Optional]

&behavior=~~repeat~~|at_most_once|exactly_once|at_least_once

(~~obsolete~~)&force=~~false~~|true

&collision=ignore|force|queue

&queue_timeout=<queue_timeout_in_seconds>

Description

Play YouTube in a designated partition

[Optional]

behavior (default=repeat)

repeat: Not to block the next partition command while playing. After playing not to switch back to default playing list.

at_most_once: Not to block the next partition command while playing. After finish switch back to the default playing list and continue the planned program.

exactly_once: Block the next partition command while playing. Switch back to the default playing list and continues the planned program.

at_least_once: Block the partition command while playing, not to switch to the default playing list.

(~~obsolete~~)force (default=false)

true: Ignore the previous partition command "behavior" parameter and execute the command.

collision (default=ignore)

ignore: When the execution is blocked by the previous command, abort execution.

force: When the execution is blocked by the previous command, force execution.

queue: When the execution is blocked by the previous command, que to execute after the previous command. There's only one command allowed in the que. When two commands enter queuing, then the later one replaces the previous one.

queue_timeout (default=5)

seconds: The queue time out time. When the queue command is out of this time, then the command will not be executed.

Example

http://127.0.0.1:8085/interaction/action?type=play_youtube&interaction_key=0000&program_id=35&partition_id=1&url=https://www.youtube.com/watch?v=8CmRIp3y9j4&behavior=exactly_once&collision=queue&queue_timeout=10

Possible Return Values

Success

ActionQueued

AuthenticationError

InteractionDisabled

Failure

ClientNotPlayState

ProgramNotFound

ParameterNoProgramId

ProgramNotMatched

ParameterNoPartitionId

PartitionNotFound

ParameterNoUrl

LastPlaybackNotFinished

ParameterErrBehavior

Get Event Script

GET interaction/eventscript.xml

Description

Acquire event script

Example

http://127.0.0.1:8085/interaction/eventscript.xml

Response

```
<?xml version="1.0" encoding="utf-8"?>
<eventscrip name="MyEventScript" >
  <eventmapping id="11" priority="3" program_id="16" behavior="repeat" collision="queue"
queue_timeout="10">
    <event name="Young Man" logic_concat="and">
      <sensor name="face" output="gender" op="eq" value="male" />
      <sensor name="face" output="age" op="lt" value="30" />
    </event>
  </eventmapping>
  <eventmapping id="12" priority="1" program_id="17" behavior="exactly_once"
collision="queue" queue_timeout="10">
    <event name="Young Woman" logic_concat="and">
      <sensor name="face" output="gender" op="eq" value="female" />
      <sensor name="face" output="age" op="lt" value="30" />
    </event>
  </eventmapping>
  <eventmapping id="13" priority="2" program_id="54" partition_id="1" playlist_id="2"
behavior="exactly_once" collision="queue" queue_timeout="10">
    <event name="EVENT NAME 1">
      <sensor name="rfid" output="id" op="eq" value="001" />
    </event>
  </eventmapping>
  <eventmapping id="14" priority="4" program_id="54" partition_id="1" playlist_id="3"
behavior="exactly_once" collision="queue" queue_timeout="10">
    <event name=" EVENT NAME 2">
      <sensor name="rfid" output="id" op="eq" value="002" />
    </event>
  </eventmapping>
</eventscrip>
```

Push Sensor Status

```
GET interaction/action?type=sensor
&interaction_key=<interaction_key>
&name=<sensor_name>
&output=<sensor_output_name>
&value=<sensor_output_value>
```

Description

Acquire output from sensor and save in the memory

Remark

When the "value" is empty string, the output will be automatic removed from the memory.

Example

```
http://127.0.0.1:8085/interaction/action?type=sensor&interaction_key=0000&name=face&output=age&value=20
http://127.0.0.1:8085/interaction/action?type=sensor&interaction_key=0000&name=face&output=gender&value=male
```

Possible Return Values

When the sensor initiates the event to switch Program, then return Start Program API response and parameter of eventmapping_id

When the sensor initiates the event to switch Playlist, then return Start Partition API response and parameter of eventmapping_id

When the sensor initiates multiple event, then returns 9 – Multiple responses exist, with next levels of individual event returns.

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<response code="9" message="Multiple responses exist">
  <response code="0" message="Success" eventmapping_id="333" />
  <response code="7" message="Action has been queued" eventmapping_id="328" />
</response>
```

NothingHappened: Sensor status receives but no event is initiated.

Get Sensor Status

```
GET interaction/sensor.xml?interaction_key=<interaction_key>
```

Description

Acquire sensor status

Example

```
http://127.0.0.1:8085/interaction/sensor.xml
```

Response

```
<response code="0" message="Success">
```

```
<sensor name="face" output="age" value="23"/>
<sensor name="face" output="gender" value="male"/>
</response>
```

Possible Return Values

Success

Failure

Get Configuration

GET interaction/configuration.xml?interaction_key=<interaction_key>

Description

Acquire configuration of the IMM client

Example

http://127.0.0.1:8085/interaction/configuration.xml&interaction_key=0000

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response code="0" message="Success">
  <imm_client>
    <version>WISE-PaaS/SignageCMS Client 1.8.2</version>
    <id>112233445566</id>
    <name>Demo Client</name>
    <status>playing</status>
    <schedule>Interactive Program</schedule>
    <program>Interactive Main Page</program>
    <free_space>94.38 GB</free_space>
    <server_ip>127.0.0.1</server_ip>
    <server_port>80</server_port>
    <server_status>ready</server_status>
    <mqtt_status>connected</mqtt_status>
  </imm_client>
</response>
```

Possible Return Values

InteractionDisabled

AuthenticationError

Set Configuration

GET interaction/action?type=config&interaction_key=<interaction_key>

&id=<id>

&server_ip=<server_ip>

&server_port=<server_port>

Description

Set configuration and restart IMM client IMM client ◦

Example

http://127.0.0.1:8085/

interaction/action?type=config&interaction_key=<interaction_key>

&id=<id>

&server_ip=<server_ip>

&server_port=<server_port>

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response code="0" message="Success">
  <imm_client>
    <version>WISE-PaaS/SignageCMS Client 1.8.4</version>
    <id>112233445566</id>
    <name>Demo Client</name>
    <status>playing</status>
    <schedule>Interactive Program</schedule>
    <program>Interactive Main Page</program>
    <free_space>94.38 GB</free_space>
    <server_ip>127.0.0.1</server_ip>
    <server_port>80</server_port>
    <server_status>ready</server_status>
    <mqtt_status>connected</mqtt_status>
  </imm_client>
</response>
```

Possible Return Values

InteractionDisabled

AuthenticationError

Error Codes

Name	Code	Description
Success	0	Success
Failure	1	Failure
AuthenticationError	2	Authentication error
ClientNotPlayState	3	Client not in play state
ClientNotStandbyState	4	Client not in standby state
InteractionDisabled	5	Interaction disabled
LastPlaybackNotFinished	6	Last Playback is not finished
ActionQueued	7	Action has been queued
NothingHappened	8	No events were triggered
ScheduleNotFound	1001	Schedule not found"
ProgramNotFound	1002	Program not found
PartitionNotFound	1003	Partition not found
PlaylistNotFound	1004	Playlist not found
ActionTypeNotFound	1005	Action type not found
MediaNotFound	1006	Media not found
ProgramNotMatched	2002	Program not matched
ParameterNoProgramId	3002	Action type needed
ParameterNoPartitionId	3003	Program ID needed

ParameterNoPlaylistId	3004	Partition ID needed
ParameterNoActionType	3005	Playlist ID needed
ParameterNoText	3006	Text needed
ParameterNoUrl	3007	URL needed
ParameterNoMediaId	3008	Media ID needed
ParameterErrBehavior	3009	Error behavior

Server Status Values

Value	Description
disconnected	Client is unable to connect to the server.
ready	Server is ready.
unrecognizedclient	Server is unable to recognize this client.
unauthorized	Server license error.

MQTT Status Values

Value	Description
disconnected	MQTT is disconnected from the server.
connecting	MQTT is trying to connect to the server.
connected	MQTT is connected to the server.

Client Status Values

Value	Description
standby	Client is standby, not playing.
playing	Client is playing.

Media types

image
video
web
html
flash