

Polycom RMX 500C/1000C XML API Documentation

Table of Content

Polycom RMX 500C/1000C XML API Documentation	1
Introduction	1
API Flow Chart	2
Communicating with the RMX XML API	4
XML Message	6
Types of Schema	6
Schemas Used to Log in to the MCU and to Create and Manage Conferences	8
Function: get alarms list	8
Function: get On-going conference details	8
Function: get On Going conference list	9
Function: get IP network services details	9
Function: get management network services details	9
Function: log in MCU	10
Function: log out MCU	10
Function: get MCU details	11
Function: get MCU time	11
Function: activate the RMX.	11
Function: get license information	12
Function: new a conference, meeting room, profile	12
Function: update a profile, meeting room	12
Function: get meeting room reservation list	13
Function: get profile list	13
Function: get resource details	14
Function: participant connects or disconnects to conference	14
Function: delete participant from conference	14

Function: set end time of conference	15
Function: audio video mute	15
Function: auto layout setting	16
Function: terminate a conference.	16
Function: audio block	17
Function: set conference password	17
Function: set conference entry password	17
Function: set conference billing data	18
Function: set conference contact information	18
Function: add participant to conference	18
Function: video layout setting	19
Function: party video layout setting	19
Function: update a party	20
Function: set lecture mode	20
Function: get a conference participant information	21
Function: get meeting room details	21
Function: get profile details	21
Function: terminate a meeting room	22
Function: terminate a profile	22
Function: get currently connected users list	22
Function: get users list	23
Function: add a user	23
Function: change user password	24
Function: delete a user	24
Function: get recording links list	24
Function: update a recording link	25
Function: get CDR list	25
Function: get LOG list	25
Function: get IVR service list	26
Detailed error code	27

Specific of RMX API	33
Description of 4 streams setting	34
Video Switching Highest Common:	34
CP Mode	34

Introduction

The contents of this document will cover the RMX Series RMX 500C and RMX 1000C; All content is applicable to these two products if it is not specified.

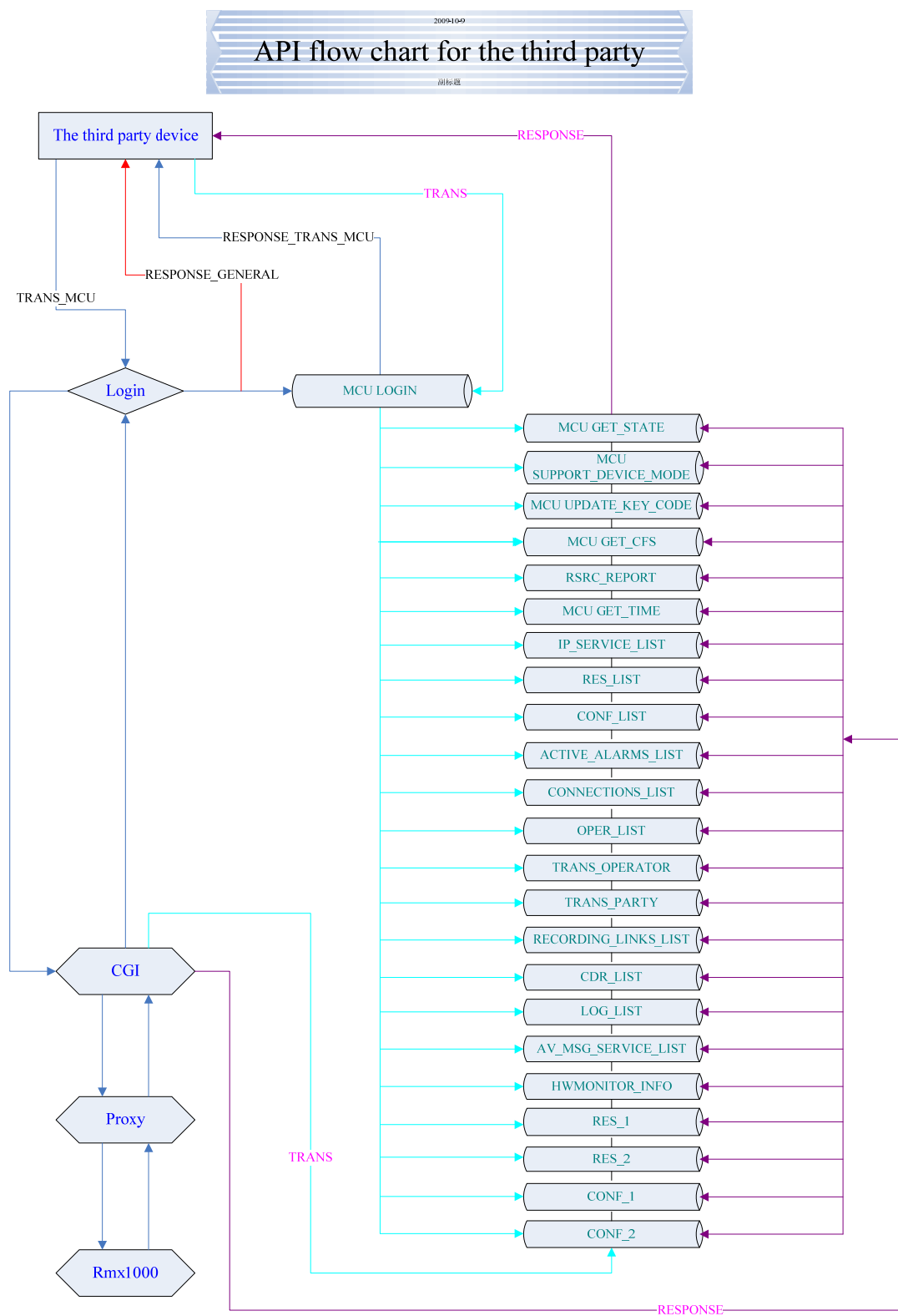
RMX: The short form of the RMX 500C/1000C.

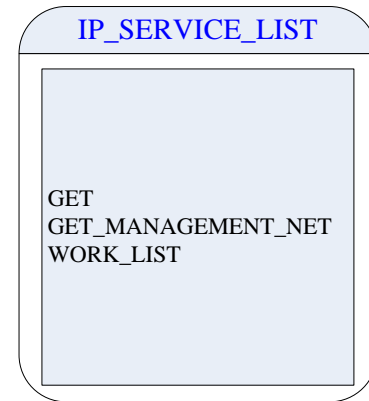
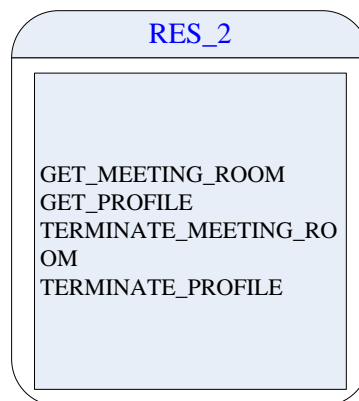
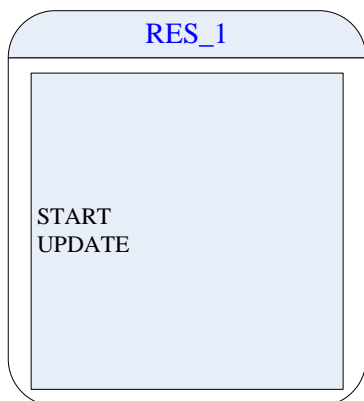
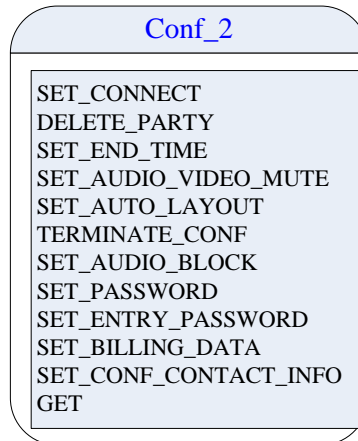
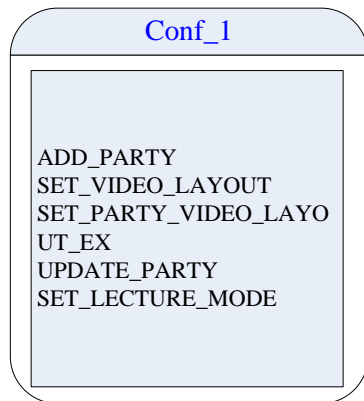
In order to support the third party which can manage RMX2000 & MGC device to integration management RMX, RMX XML API is developed.

The XML API of RMX is a subset of RMX2000 API, it's same in working style, for more information please reference RMX2000 API documents.

This documentation is to clarify that the schemas we have add to API to support the third party use XML message to manage RMX.

API Flow Chart





Communicating with the RMX XML API

Protocol: pure xml over http

Request:

http type: POST

http url: http://ip/

http version: 1.1

http header which is must:

Connection: Keep-Alive

Content-Type: text/xml

http body: pure xml api, example as following:

```
<TRANS_MCU><TRANS_COMMON_PARAMS><MCU_TOKEN>-1</MCU_TOKEN><MCU_USER_TOKEN>-1</MCU_USER_TOKEN></TRANS_COMMON_PARAMS><ACTION><LOGIN><MCU_IP><IP>172.21.111.41</IP><LISTEN_PORT>80</LISTEN_PORT></MCU_IP><USER_NAME>POLYCOM</USER_NAME><PASSWORD>POLYCOM</PASSWORD><STATION_NAME>notdmaalpha1.prx.eng.westminster.polycom.com/198.18.0.1</STATION_NAME></LOGIN></ACTION></TRANS_MCU>
```

Response from RMX: pure xml api, example as following:

```
<RESPONSE_TRANS_MCU><RETURN_STATUS><ID>0</ID><DESCRIPTION>Status OK</DESCRIPTION><YOUR_TOKEN1>0</YOUR_TOKEN1><YOUR_TOKEN2>0</YOUR_TOKEN2><MESSAGE_ID>0</MESSAGE_ID><DESCRIPTION_EX/></RETURN_STATUS><ACTION><LOGIN><MCU_TOKEN>11</MCU_TOKEN><MCU_USER_TOKEN>11</MCU_USER_TOKEN><VERSION_LIST><MCU_VERSION><MAIN>2</MAIN><MAJOR>0</MAJOR><MINOR>0</MINOR><INTERNAL>0</INTERNAL><PRIVATE_DESCRIPTION/></MCU_VERSION><MCMS_VERSION><MAIN>0</MAIN><MAJOR>0</MAJOR><MINOR>0</MINOR><INTERNAL>0</INTERNAL><PRIVATE_DESCRIPTION/></MCMS_VERSION></VERSION_LIST><AUTHORIZATION_GROUP>administrator</AUTHORIZATION_GROUP><API_NUMBER>1000</API_NUMBER><PRODUCT_TYPE>Rmx_1000</PRODUCT_TYPE><HTTP_PORT>0</HTTP_PORT><ENTRY_QUEUE_ROUTING>numeric_id_routing</ENTRY_QUEUE_ROUTING></LOGIN></ACTION></RESP
```


ONSE_TRANS_MCU>

For more information of communicating with the RMX XML API please
reference RMX 2000 XML API document.

XML Message

Types of Schema

The schemas in RMX XML API include two categories with different prefix: prefix trans and prefix response.

Transaction Schemas (Prefix trans)

The transaction schemas are used to retrieve the data of objects in the MCU (get requests), or to act on objects in the MCU (set requests). For example, the schema trans_res1 defines the XML format to start, start repeated, login start and update reservations on the MCU.

The transaction schemas often contain other types of schemas, such as object schemas and common schemas. For example, the trans_res1 schema includes the obj_reservation schema, because it has to include the reservation data when it is sent to start a reservation.

The transaction schemas have a consistent pattern that demonstrates object oriented methodology, as follows:

- 1 The root element is the schema name, which is similar to an MGC API class name.
- 2 The next element is the trans_common_params element, which is similar to a base class. This element mainly describes the MCU on which the action is to operate, and the synchronization method to be used
- 3 The next element is the ACTION element.
- 4 After this there are details of the individual actions that the transaction can make, which are similar to the MGC API class functions. For example, the action in trans_res1 used in RMX XML is START (to start or reserve a conference). There is a description of the parameters that must be specified in order to initiate the action.

Response Schemas (Prefix response_trans)

The response schemas describe the XML format returned by the XML API for the sent transactions.

The response schemas have a consistent pattern, as follows:

- 1 The root element is the schema name.
- 2 The next element is the return status for the transaction.

- 3 The next element is the ACTION element.
- 4 After this is a choice of the actions that this response answers, corresponding to the actions in the trans_xxx schema. Under each action there is a description of the returned data.

In general, each trans_xxx schema has a corresponding response_trans_xxx schema. However, although due to internal performance considerations the trans_res and trans_conf schemas were each split into two schemas (trans_res1 and trans_res2, and trans_conf1 and trans_conf2, respectively), there is only one response schema for each pair of transactions, (response_trans_res and response_trans_conf, respectively).

Each response schema contains a status which is a numeric value, and usually also a description.

Schemas Used to Log in to the MCU and to Create and Manage Conferences

The following schemas are used to log in to the MCU and to create and manage conferences:

Function: get alarms list

Message:

<TRANS_ACTIVE_ALARMS_LIST>

...

<GET>

Description: Used to retrieve details of system alerts.

Response:

<RESPONSE_TRANS_ACTIVE_ALARMS_LIST>

Description: Indicates and contains the requested information.

Function: get On-going conference details

Message:

<TRANS_CONF_2>

...

<GET>

Description: Used to manage On Going conferences.

Response:

<RESPONSE_TRANS_CONF>

Description: Indicates that the requested action was to retrieve the details of a specified On Going Conference, and contains the conference details.

Depending on the value you specify in the OBJ_TOKEN element, you

will receive full conference data, or only object members that were changed since the previous GET request.

Function: get On Going conference list

Message:

< TRANS_CONF_LIST >

...

< GET_LS >

Description: Used to retrieve a list of On Going conferences from MCU.

Response:

< RESPONSE_TRANS_CONF_LIST >

Description: The RESPONSE_TRANS_CONF_LIST element contains the response to the TRANS_CONF_LIST schema, which is used to retrieve a list of On Going conferences.

Function: get IP network services details

Message:

< TRANS_IP_SERVICE_LIST >

...

< GET >

Description: Used to retrieve details of IP network services.

Response:

< RESPONSE_TRANS_IP_SERVICE_LIST >

Description: The RESPONSE_TRANS_IP_SERVICE_LIST element contains the response to the trans_ip_service_list schema, which used to retrieve details of IP Network Services.

Function: get management network services details

Message:

< TRANS_IP_SERVICE_LIST >

...

< GET_MANAGEMENT_NETWORK_LIST >

Description: Retrieves details of all Management Network Services.

Response:

< RESPONSE_TRANS_IP_SERVICE_LIST >

Description: The RESPONSE_TRANS_IP_SERVICE_LIST element contains the response to the trans_ip_service_list schema, which used to retrieve details of all Management Network Services, and contains the requested information.

Function: log in MCU

Message:

< TRANS_MCU >

...

< LOGIN >

Description: Used to connect a user to the MCU (SE200 login to RMX)

Response:

< RESPONSE_TRANS_MCU >

Description: Indicates that the requested action is to log in the MCU, and return the login parameters.

Function: log out MCU

Message:

< TRANS_MCU >

...

< LOGOUT >

Description: Terminates the user connection to the MCU.

Response:

< RESPONSE_TRANS_MCU >

Description: Indicates that the requested action was to log out of the MCU.

Function: get MCU details

Message:

< TRANS_MCU >

...

< GET_STATE >

Description: Used to retrieve information about the MCU.

Response:

< RESPONSE_TRANS_MCU >

Description: Indicates and contains the requested information.

Function: get MCU time

Message:

< TRANS_MCU >

...

< GET_TIME >

Description: Used to retrieve the current time and date from the MCU.

Response:

< RESPONSE_TRANS_MCU >

Description: Indicates and contains the requested information.

Function: activate the RMX.

Message:

< TRANS_MCU>

...

< UPDATE_KEY_CODE>

Description: used to activate the RMX.

Response:

< RESPONSE_TRANS_MCU >

Description: Indicates that the requested action was to activate the RMX.

Function: get license information

Message:

< TRANS_MCU >

...

< GET_CFS >

Description: used to Retrieves license information.

Response:

< RESPONSE_ TRANS_MCU >

Description: Indicates that the requested action was to retrieve license information, and contains the requested information.

Function: new a conference, meeting room, profile

Message:

< TRANS_RES_1 >

...

< START >

Description: Used to setup a new conference, meeting room, profile.

Response:

< RESPONSE_TRANS_RES >

Description: Indicates and contains the conference details, meeting room, profile.

Function: update a profile, meeting room

Message:

< TRANS_RES_1 >

...

< UPDATE >

Description: Used to Update a Profile, Meeting Room

Response:

< RESPONSE_TRANS_RES >

Description: Indicates that the requested action was to update details a profile, meeting room and contains the update details..

Function: get meeting room reservation list

Message:

< TRANS_RES_LIST >

...

< GET_MEETING_ROOM_LIST >

Description: Used to retrieve a list of Meeting Room Reservation summaries.

Response:

< RESPONSE_TRANS_RES_LIST >

Description: Indicates and contains the meeting room reservation summary information and requested information. Depending on the value of the OBJ_TOKEN element, the GET_MEETING_ROOM_LIST element will either retrieve all the Meeting Room reservation summaries, or only those reservations which are new or modified since the last time this request was sent.

Function: get profile list

Message:

< TRANS_RES_LIST >

...

< GET_PROFILE_LIST >

Description: Used to retrieve a list of Profile summaries.

Response:

< RESPONSE_TRANS_RES_LIST >

Description: Indicates and contains the profile summary information and the requested information.

Depending on the value of the OBJ_TOKEN element, the GET_PROFILE_LIST element will either retrieve all the Profile summaries, or only those Profiles which are new or modified since the

last time this request was sent.

Function: get resource details

Message:

```
< TRANS_RSRC_REPORT >

...

< GET_CARMEL_REPORT >
```

Description: Used to retrieve resource reports and to set the resource allocation method.

Response:

```
< RESPONSE_TRANS_RSRC_REPORT >
```

Description: The RESPONSE_TRANS_RSRC_REPORT element contains the response to the TRANS_RSRC_REPORT schema, which is used to retrieve resource reports and to set the resource allocation method.

Function: participant connects or disconnects to conference

Message:

```
< TRANS_CONF_2 >

...

< SET_CONNECT >
```

Description: Used to connect a specific participant to a conference, or disconnects a specific participant from a conference.

Specify the conference ID in the ID element.

Response:

```
< RESPONSE_TRANS_CONF >
```

Description: Indicates that the requested action was to connect a participant to a conference, or disconnect a participant from a conference.

Function: delete participant from conference

Message:

< TRANS_CONF_2 >

...

< DELETE_PARTY >

Description: Used to remove a specified participant from the conference.

Deleting a participant removes the participant's definition from the conference.

Specify the conference ID in the ID element.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to remove a specified participant from a conference.

Function: set end time of conference

Message:

< TRANS_CONF_2 >

...

< SET_END_TIME >

Description: Used to set the end time of an On Going conference.

Specify the conference ID in the ID element.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to set the end time of the On Going conference.

Function: audio video mute

Message:

< TRANS_CONF_2 >

...

< SET_AUDIO_VIDEO_MUTE >

Description: Used to mute or unmute the audio and/or video signal of a specified participant. While the audio and/or video is muted, the

participant receives audio/video signal from other participants, but does not transmit audio/video to them.

Specify the conference ID in the ID element.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to mute or unmute the audio and/or video signal of a specified participant.

Function: auto layout setting

Message:

< TRANS_CONF_2 >

...

< SET_AUTO_LAYOUT >

Description: Used to set Auto Layout activated or inactivated for the conference.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to activate or de-activate the Auto Layout feature.

Function: terminate a conference.

Message:

< TRANS_CONF_2 >

...

< TERMINATE_CONF >

Description: Used to terminate a conference.
Specify the conference ID in the ID element.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to terminate the conference.

Function: audio block

Message:

< TRANS_CONF_2 >

...

< SET_AUDIO_BLOCK >

Description: Used to blocks or unblocks the audio transmission from the conference to a specified participant.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to block or unblock the audio transmission from the conference to a specified participant.

Function: set conference password

Message:

< TRANS_CONF_2 >

...

< SET_PASSWORD >

Description: Used to set the chairperson conference password.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to set the chairperson conference password.

Function: set conference entry password

Message:

< TRANS_CONF_2 >

...

< SET_ENTRY_PASSWORD >

Description: Used to set the conference entry password.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to set conference entry password.

Function: set conference billing data

Message:

< TRANS_CONF_2 >

...

< SET_BILLING_DATA >

Description: Used to set conference billing code.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to set conference billing code.

Function: set conference contact information

Message:

< TRANS_CONF_2 >

...

< SET_CONF_CONTACT_INFO >

Description: Used to set conference contact information.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to set conference contact information.

Function: add participant to conference

Message:

< TRANS_CONF_1 >

...

< ADD_PARTY >

Description: Used to add a participant to an on going conference.

If the participant's connection properties allows the MCU to initiate a connection, the MCU will attempt to automatically connect the added participant to the conference.

Specify the conference ID in the ID element.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to add a participant to the conference.

Function: video layout setting

Message:

< TRANS_CONF_1 >

...

< SET_VIDEO_LAYOUT >

Description: Used to set the video layout of an On Going conference.

You can use this element to change the number of cells in the layout and to force each cell.

Specify the conference ID in the ID element.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to update the video layout for the conference.

Function: party video layout setting

Message:

< TRANS_CONF_1 >

...

< SET_PARTY_VIDEO_LAYOUT_EX >

Description: Used to Set whether the specified participant sees his/her

Personal Layout or the Conference Layout, and forces a specific screen layout.

Specify the conference ID in the ID element.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to set the participant layout type and to force a specified screen layout.

Function: update a party

Message:

< TRANS_CONF_1 >

...

< UPDATE_PARTY >

Description: Updates the parameters of a participant. This element can only be used to update the properties of disconnected participants

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to update the parameters of a participant.

Function: set lecture mode

Message:

< TRANS_CONF_1 >

...

< SET_LECTURE_MODE >

Description: Sets the Lecture mode parameters.

Response:

< RESPONSE_TRANS_CONF >

Description: Indicates that the requested action was to sets the Lecture mode

Function: get a conference participant information

Message:

< TRANS_PARTY >

...

< GET >

Description: used to retrieve information about a conference participant.

Response:

< RESPONSE_ TRANS_PARTY >

Description: contains the response to the trans_party schema, which is used to retrieve information about a conference participant.

Function: get meeting room details

Message:

< TRANS_RES_2 >

...

< GET_MEETING_ROOM >

Description: Retrieves details of a specified Meeting Room.

Response:

< RESPONSE_TRANS_RES >

Description: Indicates that the requested action was to retrieve details of a specific Meeting Room and contains the requested information.

Function: get profile details

Message:

< TRANS_RES_2 >

...

< GET_PROFILE >

Description: Retrieves details of a specified profile.

Response:

< RESPONSE_TRANS_RES >

Description: Indicates that the requested action was to retrieve details of a specific profile and contains the requested information.

Function: terminate a meeting room

Message:

< TRANS_RES_2 >

...

< TERMINATE_MEETING_ROOM >

Description: Deletes a Meeting Room.

Response:

< RESPONSE_TRANS_RES >

Description: Indicates that the requested action was to delete a Meeting Room.

Function: terminate a profile

Message:

< TRANS_RES_2 >

...

< TERMINATE_PROFILE >

Description: Deletes a profile.

Response:

< RESPONSE_TRANS_RES >

Description: Indicates that the requested action was to delete a profile.

Function: get currently connected users list

Message:

< TRANS_CONNECTIONS_LIST >

...

< GET >

Description: Retrieves information about users who are currently connected to the MCU

Response:

< RESPONSE_TRANS_CONNECTIONS_LIST >

Description: The RESPONSE_TRANS_CONNECTIONS_LIST element contains the response to the trans_connections_list schema, which is used to retrieve information about users who are currently connected to the MCU.

Function: get users list

Message:

< TRANS_OPER_LIST >

...

< GET_OPER_LIST >

Description: Retrieves a list of users and their details

Response:

< RESPONSE_TRANS_OPER_LIST >

Description: The RESPONSE_TRANS_OPER_LIST element contains the response to the trans_oper_list schema, which is used to retrieve details of users.

Function: add a user

Message:

< TRANS_OPERATOR >

...

< NEW_OPERATOR >

Description: Adds a new user.

Response:

< RESPONSE_TRANS_OPERATOR >

Description: Indicates that the requested action was to add a new user.

Function: change user password

Message:

< TRANS_OPERATOR >

...

< CHANGE_PASSWORD >

Description: Changes the password of an existing user.

Response:

< RESPONSE_TRANS_OPERATOR >

Description: Indicates that the requested action was to change the password of an existing user.

Function: delete a user

Message:

< TRANS_OPERATOR >

...

< DELETE_OPERATOR >

Description: Deletes a user.

Response:

< RESPONSE_TRANS_OPERATOR >

Description: Indicates that the requested action was to delete a user.

Function: get recording links list

Message:

< TRANS_RECORDING_LINKS_LIST >

...

< GET >

Description: Retrieves a list of recording links and their details

Response:

< RESPONSE_TRANS_RECORDING_LINKS_LIST >

Description: Indicates that the requested action was to retrieve details of recording links.

Function: update a recording link

Message:

< TRANS_RECORDING_LINKS_LIST >

...

< UPDATE >

Description: update a recording link.

Response:

< RESPONSE_ TRANS_RECORDING_LINKS_LIST >

Description: Indicates that the requested action was to update a recording link.

Function: get CDR list

Message:

< TRANS_CDR_LIST >

...

< GET >

Description: The TRANS_CDR_LIST element is used to retrieve a list of CDR summaries from the MCU.

Response:

< RESPONSE_ TRANS_CDR_LIST >

Description: The RESPONSE_TRANS_CDR_LIST element contains the response to the trans_cdr_list schema, which is used to retrieve a list of CDR summaries from the MCU.

Function: get LOG list

Message:

< TRANS_LOG_FILE_LIST >

...

< GET >

Description: The TRANS_LOG_LIST element is used to retrieve a list of log file summaries.

Response:

< RESPONSE_ TRANS_LOG_FILE_LIST >

Description: The RESPONSE_TRANS_LOG_LIST element contains the response to the trans_cdr_list schema, which is used to retrieve a list of log file summaries.

Function: get IVR service list

Message:

< TRANS_AV_MSG_SERVICE_LIST >

...

< GET >

Description: used to retrieve information about Conference IVR Services.

Response:

< RESPONSE_ TRANS_AV_MSG_SERVICE_LIST >

Description: contains the response to the trans_av_msg_service_list schema, which is used to retrieve information about Conference IVR Services.

Detailed error code

Description: Detailed error code for the third party.

CODE	Description
10001	Permission denied.
10002	Invalid message transmission. The web interface sent an unreadable command to the RMX1000.
10003	Invalid GUID. Unable to add user.
10004	The GUID has been duplicated. Unable to add user.
10005	Account already exists.
10006	Maximum users already configured – users must be deleted to add more.
10007	Your login does not have permission to add users.
10012	Your login does not have permission to change the password.
10013	Unable to delete the operator. Its internal reference ID is invalid.
10014	Unable to delete the operator. Its internal reference ID could not be found.
10015	Your login does not have permission to delete the user.
10016	This online user can not be deleted.
10017	An operator cannot delete its own account.
10020	Invalid username, please try again.
10021	Invalid password.
10023	Invalid IP address or subnet mask.
10024	Invalid field in the attempted route entry configuration.

CODE	Description
10025	Invalid system configuration setting.
10028	There are ongoing conferences. Unable to synchronize time.
20001	Unable to add the address.
20002	Address cannot be found.
20003	Unable to add the group.
20004	Group cannot be found.
20005	Unable to add the profile.
20006	Profile cannot be found.
20007	Unable to delete the profile – it may be in use.
20008	Unable to add the meeting room.
20008	Unable to add the meeting room – the name or ID is already in use.
20008	Maximum meeting rooms exceed – meeting rooms must be deleted to add more.
20008	Unable to add the meeting room – the streaming address is already in use.
20009	Meeting room cannot be found.
20010	Unable to delete the meeting room – it may be in use.
20011	Unable to add the reservation.
20011	Maximum reservations already configured – reservations must be deleted to add more.
20011	Unable to add the reservation – the name or ID is already in use.
20011	Unable to add the reservation – insufficient resources at the requested time.
20011	Unable to add the reservation – insufficient CP resources.

CODE	Description
20011	Unable to add the reservation – the scheduled time is in the past.
20011	Unable to add the reservation – current licensing does not permit scheduling.
20011	Unable to add the reservation – the streaming address is already in use.
20012	Unable to modify the reservation.
20012	Unable to modify the reservation – the name or ID is already in use.
20012	Unable to modify the reservation – the requested resources are not available.
20012	Unable to modify the reservation – insufficient CP resources.
20012	Unable to modify the reservation – the scheduled time is in the past.
20012	Unable to modify the reservation – the conference is ongoing.
20012	Unable to modify the reservation – the reservation doesn't exist.
20012	Unable to modify the reservation – current licensing does not permit scheduling.
20012	Unable to modify the reservation – the streaming address is already in use.
20013	This reservation cannot be found.
20014	Unable to delete this reservation.
20015	Unable to add temporary conference.
20015	Maximum conferences already configured – conferences must be deleted to add more.
20015	Unable to add the conference – the name or ID is already in use.

CODE	Description
20015	Unable to add the conference – insufficient resources at the requested time.
20015	Unable to add the conference – insufficient CP resources.
20015	Unable to add the conference – the scheduled time is in the past.
20015	Unable to add the conference – current licensing does not permit this mode.
20015	Unable to add the conference – the streaming address is already in use.
20016	This temporary conference cannot be found.
20017	There was an error controlling the conference.
20018	This online endpoint cannot be found.
20019	This offline endpoint cannot be found.
20020	Duplicate entry – this participant name already exists.
20021	Unable to modify the participant – the participant name is already in use.
20022	Duplicate entry – this group name already exists.
20023	Unable to modify the group – the group name is already in use.
20024	Unable to modify offline participant – its name is in use in a conference.
20025	Unable to add conference participant to address book – an entry with this name already exists.
20027	Unable to modify the meeting room – the name or ID is already in use.
20027	Unable to modify the meeting room – the streaming address is already in use.
21001	Unable to invite the participant – requested participant is

CODE	Description
	already in another conference.
21002	Unable to invite the participant – requested participant name is already in use.
21003	Unable to invite the participant – needed resources are unavailable.
21004	Unable to invite the participant – incomplete information provided.
21005	Requested participant is already in another conference.
22001	Unable to connect – the participant is already connected.
22002	Unable to connect the participant – needed information is missing.
22003	Participant is already disconnected.
22004	Recording Links cannot be disconnected.
22005	Incomplete information – failed to add participant.
22006	Recording Link failed to record.
22007	Incomplete information – failed to record.
22008	Participant information is incomplete.
22009	Profile does not support encryption.
22010	Profile does not support encryption and HD video-switching.
22011	Duration cannot exceed 24 hours – failed to add conference.
22012	Your system does not support H.264 4CIF.
22013	Your system does not support H.264 720P.
22014	Encrypted profile can not be set to default profile. Disable encryption to make this a possible default profile.
22015	Default profile can not enable encryption.

CODE	Description
22016	The conference duration can not exceed 24 hours.
22017	Conference duration cannot be extended as requested - resources are not available.
22018	Your system does not support Cluster.
22019	Your system does not support Lost Packet Recovery.
22020	Unable to upload the personal skin. Please make sure the file being uploaded is in the correct format.

Specific of RMX API

Conference duration: conference Duration cannot exceed 24 hours.

Conference ID length: Conference ID length must be 4 digits.

Description of 4 streams setting

Video Switching Highest Common:

Mode	Default value of Stream1[Protocol, Format, Frame Rate]
Video Switch Mode	["AUTO", "AUTO", "AUTO"]

CP Mode

If the third party can not set the 4 streams parameters, here we will set default value for 4 streams according to the line rate.

Set the default value of People Video Definition according to line rate:

For RMX1000C:

if Line rate ≥ 1728 and RMX support 1080P, then set People Video Definition value to UP_TO_H264_1080P;

else if Line rate ≥ 1024 and Line rate ≤ 2560 , then set People Video Definition value to UP_TO_H264_720P

else if Line rate ≥ 256 , then set People Video Definition value to UP_TO_H264_4CIF/4SIF

else, set People Video Definition value to UP_TO_H264_CIF/SIF.

For RMX500C:

else if Line rate ≥ 832 , then set People Video Definition value to UP_TO_H264_720P

else if Line rate ≥ 256 , then set People Video Definition value to UP_TO_H264_4CIF/4SIF

else, set People Video Definition value to UP_TO_H264_CIF/SIF.

Set the default value of Protocol, Format and Frame Rate according to People Video Definition and Line Rate:

CP Mode: If Line rate is lower than 128, then set frame rate of stream 4 to 12.5; else if Line rate is lower than 256, then set frame rate of stream 2 to 12.5; Else set the Frame Rate to 25.

FIRST_VIDEO_DEFINITION	Default value of [STREAM_VIDEO_PROTOCOL, STREAM_VIDEO_FORMAT]			
	Stream1	Stream2	Stream3	Stream4
UP_TO_H264_CIF/SIF	["H.264", "CIF/SIF"]	["H.264", "CIF/SIF"]	["H.264", "CIF/SIF"]	["H.263", "CIF/SIF"]
UP_TO_H264_4CIF/4SIF	["H.264", "4CIF/4SIF_16:9"]	["H.264", "CIF/SIF"]	["H.264", "CIF/SIF"]	["H.263", "CIF/SIF"]
UP_TO_H264_720P	["H.264", "720P"]	["H.264", "4CIF/4SIF_16:9"]	["H.264", "CIF/SIF"]	["H.263", "CIF/SIF"]
UP_TO_H264_1080P	["H.264", "1080P"]	["H.264", "720P"]	["H.264", "CIF/SIF"]	["H.263", "CIF/SIF"]

Set default value of stream 2, stream 3 and stream 4 Line Rate according to stream1 Line Rate:

For RMX1000C, the line of stream1 can up to 4096; And the line rate of stream3 and stream4 can set to the following value;

For RMX500C, the line of stream1 can up to 1920; And the line rate of stream3 and stream4 only can set to "0";

Stream 1	Default value of STREAM_LINE_RATE		
	Stream 2	Stream 3	Stream 4
4096	1920	768	384
3072	1536	768	384
2560	1024	768	384
2048	1024	768	384
1920	1024	768	384
1536	768	512	384
1280	768	512	384

	Default value of STREAM_LINE_RATE		
1024	768	512	384
768	512	384	256
512	384	256	128
384	256	128	64
320	256	128	64
256	192	128	64
192	192	128	64
128	128	128	64
64	64	64	64

The detailed rules of the 4 streams parameters setting can refer to 'Description of 4 streams setting' in inner API document\ RMX 500C_100OC XML API Documentation.