



DEVELOPER GUIDE

API Version 3.6.0 | SEPTEMBER 2018 | 3725 77702 001K

# Polycom® RealPresence® Platform API



Copyright© 2018, Polycom, Inc. All rights reserved. No part of this document may be reproduced, translated into another language or format, or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Polycom, Inc.

6001 America Center Drive  
San Jose, CA 95002  
USA



Polycom®, the Polycom logo and the names and marks associated with Polycom products are trademarks and/or service marks of Polycom, Inc. and are registered and/or common law marks in the United States and various other countries. All other trademarks are property of their respective owners. No portion hereof may be reproduced or transmitted in any form or by any means, for any purpose other than the recipient's personal use, without the express written permission of Polycom.

**End User License Agreement** By installing, copying, or otherwise using this product, you acknowledge that you have read, understand and agree to be bound by the terms and conditions of the [End User License Agreement](#) for this product.

**Patent Information** The accompanying product may be protected by one or more U.S. and foreign patents and/or pending patent applications held by Polycom, Inc.

**Open Source Software Used in this Product** This product may contain open source software. You may receive the open source software from Polycom up to three (3) years after the distribution date of the applicable product or software at a charge not greater than the cost to Polycom of shipping or distributing the software to you. To receive software information, as well as the open source software code used in this product, contact Polycom by email at [OpenSourceVideo@polycom.com](mailto:OpenSourceVideo@polycom.com).

**Disclaimer** While Polycom uses reasonable efforts to include accurate and up-to-date information in this document, Polycom makes no warranties or representations as to its accuracy. Polycom assumes no liability or responsibility for any typographical or other errors or omissions in the content of this document.

**Limitation of Liability** Polycom and/or its respective suppliers make no representations about the suitability of the information contained in this document for any purpose. Information is provided "as is" without warranty of any kind and is subject to change without notice. The entire risk arising out of its use remains with the recipient. In no event shall Polycom and/or its respective suppliers be liable for any direct, consequential, incidental, special, punitive or other damages whatsoever (including without limitation, damages for loss of business profits, business interruption, or loss of business information), even if Polycom has been advised of the possibility of such damages.

**Customer Feedback** We are striving to improve our documentation quality and we appreciate your feedback. Email your opinions and comments to [DocumentationFeedback@polycom.com](mailto:DocumentationFeedback@polycom.com).



Visit the [Polycom Support Center](#) for End User License Agreements, software downloads, product documents, product licenses, troubleshooting tips, service requests, and more.

# Contents

- Contents ..... 3**
- Before You Begin ..... 8**
  - Scope of This Guide ..... 8
  - Audience, Purpose and Required Skills ..... 8
  - Get Help ..... 8
  - API Terminology Used ..... 8
  - Supported RealPresence Platform API Version ..... 8
  - Related Documentation ..... 9
    - RealPresence Resource Manager and DMA System Documentation* ..... 9
- RealPresence® Platform API History ..... 10**
  - Associated API and Product Versions ..... 10
  - RealPresence Platform API History ..... 11
    - RealPresence Platform API Version 3.6.0* ..... 11
      - plcm-ep-registration ..... 12
      - plcm-rest-login-details ..... 12
      - plcm-acl ..... 12
      - plcm-active-call ..... 12
      - plcm-backup ..... 12
      - plcm-port-range-settings ..... 13
      - plcm-power-management ..... 13
      - plcm-registration-policy-debug-request ..... 13
      - plcm-registration-policy-v2 ..... 13
      - plcm-call ..... 14
      - plcm-security-settings ..... 14
      - plcm-gatekeeper-config-v2 ..... 14
      - plcm-gatekeeper-config ..... 14
      - plcm-troubleshooting-utilities ..... 15
      - plcm-registration-summary ..... 15
    - RealPresence Platform API Version 3.5.0* ..... 15
      - Changes affecting multiple resources ..... 15
      - plcm-acl ..... 15
      - plcm-active-call ..... 16
      - plcm-alerting-config ..... 16
      - plcm-alert ..... 16
      - plcm-backup ..... 16
      - plcm-billing ..... 16
      - plcm-caa-tenant ..... 16
      - plcm-common ..... 17

plcm-conference-factory.....	17
plcm-conference-room .....	17
plcm-conference-settings .....	17
plcm-conference .....	17
plcm-conference-template .....	18
plcm-config .....	18
plcm-device.....	19
plcm-dial-plan .....	19
plcm-dma-supercluster.....	20
plcm-enterprise-directory-cache .....	21
plcm-enterprise-directory-reports.....	21
plcm-enterprise-directory.....	21
plcm-enterprise-group .....	21
plcm-eula.....	21
plcm-ext-sip-peer .....	22
plcm-hunt-group.....	22
plcm-license.....	22
plcm-ext-sip-peer .....	22
plcm-login-policy .....	22
plcm-login-sessions.....	23
plcm-mcu-pool-order .....	23
plcm-mcu .....	23
plcm-network .....	23
plcm-pkix .....	23
plcm-registration-policy .....	23
plcm-remote-backup .....	24
plcm-reports.....	24
plcm-rprm-integration .....	25
plcm-security-settings .....	25
plcm-signaling-config .....	25
plcm-snmp-config.....	26
plcm-software-upgrade.....	26
plcm-task .....	26
plcm-territory.....	26
plcm-troubleshooting-utilities .....	27
plcm-user-domain .....	27
plcm-user.....	27
plcm-user-domain .....	28
<i>RealPresence Platform API Version 3.4.0.....</i>	<i>28</i>
plcm-mcu .....	28
plcm-conference-room .....	28
<i>RealPresence Platform API Version 3.2.1.....</i>	<i>28</i>
plcm-conference .....	28
plcm-provision.....	28
plcm-network-device .....	29

plcm-enterprise-directory.....	29
<i>RealPresence Platform API Version 2.7.1</i> .....	29
<i>RealPresence Platform API Version 2.6.2</i> .....	30
<i>RealPresence Platform API Version 2.6.1</i> .....	31
<i>RealPresence Platform API Version 2.6.0</i> .....	31
<i>RealPresence Platform API Version 2.5.5</i> .....	32
<i>RealPresence Platform API Version 2.5.4</i> .....	32
<i>RealPresence Platform API Version 2.5.3</i> .....	32
<i>RealPresence Platform API Version 2.5.2</i> .....	33
<i>RealPresence Platform API Version 2.4.3</i> .....	35
<i>RealPresence Platform API Version 2.4.2</i> .....	35
<i>RealPresence Platform API Version 2.2.0</i> .....	36
<i>RealPresence Platform API Version 2.0.1</i> .....	36
<i>RealPresence Platform API Version 1.7.7</i> .....	36
<i>RealPresence Platform API Version 1.7.6</i> .....	37
<i>RealPresence Platform API Version 1.7.5</i> .....	37
<i>RealPresence Platform API Version 1.7.4</i> .....	37
<i>RealPresence Platform API Version 1.7.2</i> .....	37
<i>RealPresence Platform API Version 1.7.1</i> .....	37
<b>Getting Started .....</b>	<b>40</b>
Understanding Basic Concepts.....	40
<i>Resources and HTTP Methods</i> .....	40
Recommended HTTP GET Optimization (Conditional GET) .....	40
Recommended HTTP PUT Request Concurrency Control (Conditional PUT) .....	41
Updating resources .....	41
<i>Simple User Resource Example</i> .....	42
Get Method.....	42
POST Method .....	44
DELETE Method.....	45
<i>Resource Descriptions</i> .....	45
<i>Links and Navigation</i> .....	47
The <i>title</i> attribute .....	48
Example .....	48
The <i>type</i> Attribute.....	48
Example (using XML subtype).....	48
Example (using JSON subtype).....	48
The <i>rel</i> Attribute .....	48
Example .....	48
The <i>href</i> Attribute .....	48
Example .....	48
Link Example .....	49
<i>Custom Link Relationship Types</i> .....	50
RealPresence DMA System Custom Relationship Types:.....	51

Conference Link Relationships.....	51
Conference Room Link Relationships.....	51
MCU Link Relationships.....	51
MCU Pool Link Relationships .....	52
MCU Pool Order Link Relationships .....	52
User Link Relationships .....	52
RealPresence Resource Manager Custom Relationship Types: .....	52
Device Link Relationships .....	52
User Link Relationships .....	53
<i>Authentication and Authorization .....</i>	<i>53</i>
Auditing .....	53
Audit Logs.....	53
RealPresence DMA System.....	53
RealPresence Resource Manager System .....	54
<i>Error Handling .....</i>	<i>54</i>
<i>The Passback and Passthru Fields.....</i>	<i>56</i>
<i>XML and JSON Payload Formats.....</i>	<i>56</i>
<i>Software Versioning.....</i>	<i>59</i>
Transparent Versioning.....	59
Discrete Versioning.....	59
Best Practices to Prevent Versioning Issues.....	60
<i>Subscriptions and Notifications.....</i>	<i>61</i>
Authentication .....	61
Initiating a subscription.....	62
Subscription management.....	62
Error reporting.....	62
Notification format.....	62
Subscription to lists .....	62
High Availability.....	63
Subscription renewal.....	63
Learning from Sample Code .....	64
<b>API Services .....</b>	<b>65</b>
Alerting Capabilities.....	65
Site Topology Capabilities .....	66
Provisioning Capabilities.....	67
Billing Capabilities .....	67
Conference Control and Monitoring (CCM) Capabilities .....	68
Conference Scheduling Capabilities.....	69
Resource Reporting Capabilities.....	69
Area Capabilities .....	70
Group Capabilities.....	70

Device Capabilities .....	70
System Configuration and Status Capabilities.....	71

# Before You Begin

This preface describes the:

- Scope of this guide
- Audience, Purpose and Required Skills
- Related documentation

## Scope of This Guide

This guide shows you how to use the Polycom® RealPresence Platform API to develop applications that allow a client programmable access to the RealPresence Resource Manager and RealPresence Distributed Media Application™ (DMA™) systems.

- [Getting Started](#) introduces you to the technical concepts used in the API.
- [API Services](#) describes functional areas exposed by the API.

## Audience, Purpose and Required Skills

This guide is written for application developers. It assumes that you are a developer, and have a basic understanding of:

- How applications are developed in your environment.
- Functional understanding of the HTTP, JSON, and XML.
- Familiarity with Representational State Transfer (REST) architecture and principles is helpful, but not required.

## Get Help

For more information about installing, configuring, and administering Polycom products, see Documents and Downloads at [Polycom Support](#).

## API Terminology Used

REST API terminology can vary between organizations. This guide refers to each RealPresence Platform API element as a “resource collection”. Each entity contained by the resource collection is called a “resource” and the methods used to operate on the resources are called “methods”.

## Supported RealPresence Platform API Version

This guide addresses RealPresence Platform API version 3.6.0.



## Related Documentation

While planning, developing, deploying, or troubleshooting your application, you may need to refer other Polycom® documents including:

- RealPresence Platform Application Programming Interface (API) reference documentation.

This Developer Guide and the reference documentation are meant to be used together. The Developer Guide provides an overview, explains concepts and provides examples. The reference documentation provides precise specifications of the API.

### ***RealPresence Resource Manager and DMA System Documentation***

Since this API gives you programmable access to the RealPresence Resource Manager and RealPresence Distributed Media Application™ (DMA™) system features, you may wish to reference documents about those systems.

Documents from the RealPresence Resource Manager and RealPresence DMA™ system documentation sets provide additional information about administering those systems. They can be found at the Documents and Downloads link at: <https://documents.polycom.com/>

- RealPresence DMA™ 7000 system documentation:  
[https://documents.polycom.com/bundle/dma-ops-9-0/page/dma-ops-help/vol\\_intro/vol\\_intro.htm](https://documents.polycom.com/bundle/dma-ops-9-0/page/dma-ops-help/vol_intro/vol_intro.htm)
- RealPresence Resource Manager system documentation set:  
[https://documents.polycom.com/bundle/rprm-ops-10-3/page/rprm\\_ops/Ch1\\_Getting\\_Started/Ch1\\_Getting\\_Started.htm](https://documents.polycom.com/bundle/rprm-ops-10-3/page/rprm_ops/Ch1_Getting_Started/Ch1_Getting_Started.htm)

# RealPresence® Platform API History

This section describes the history of the RealPresence Platform API and changes between released versions. With this information, you can ensure that the applications you are developing are compatible with the versions of the products in your environment.

## Associated API and Product Versions

The following is a list of the product versions associated with each public release of the RealPresence Platform API.

Version	Associated Product Releases
3.6.0	RealPresence DMA System v10.0.0
3.5.4	RealPresence DMA System v9.0.1.2
3.5.3	RealPresence DMA System v9.0.1.1
3.5.2	RealPresence DMA System v9.0.1
3.5.1	RealPresence DMA System v9.0.0.3
3.5.0	RealPresence DMA System v9.0.0
3.4.0	RealPresence DMA System v6.4.0
3.2.2	RealPresence Resource Manager System v10.0
3.2.0 or 3.2.1	RealPresence Resource Manager System v9.0.0, v9.0.0R, or v9.0.1
2.7.1	RealPresence Resource Manager System v8.4.0
2.6.2	RealPresence DMA System v6.2.1
2.6.1	RealPresence Resource Manager System v8.3.1, v8.3.2
2.6.0	RealPresence DMA System v6.2.0
2.5.5	RealPresence DMA System v6.1.3

<b>Version</b>	<b>Associated Product Releases</b>
2.5.4	RealPresence DMA System v6.1.2
2.5.3	RealPresence DMA System v6.1.1
2.5.2	RealPresence DMA System v6.1.0
2.4.3	RealPresence Resource Manager System v8.3.0
2.4.2	RealPresence Resource Manager System v8.2.0, v8.2.1
2.4.0	RealPresence Access Director System v4.0.0, v4.0.1, v4.1
2.2.0	RealPresence Resource Manager System v8.1.0, v8.1.1
2.0.1	RealPresence Resource Manager System v8.0, v8.0.1
1.7.7	RealPresence DMA System v6.0.6
1.7.6	RealPresence DMA System v6.0.5
1.7.5	RealPresence DMA System v6.0.4
1.7.4	RealPresence DMA System v6.0.3
1.7.2	RealPresence DMA System v6.0.2.1
1.7.1	RealPresence DMA System v6.0.2
1.2.2	RealPresence DMA System v5.2.2.6

## RealPresence Platform API History

The following sections detail the changes between publicly released versions of the RealPresence Platform API.

### ***RealPresence Platform API Version 3.6.0***

The following are the changes between RealPresence Platform API version 3.6.0 and 3.5.0.

## **plcm-ep-registration**

New for this release, this resource collection allows you to control registration by endpoints.

## **plcm-rest-login-details**

New for this release, this resource collects all the login details for the current user.

## **plcm-acl**

- plcm-acl-policy-assignment:
  - A new policy assignment string has been added with H323 as the type of policy, SIP and webrtc.

## **plcm-active-call**

- Changes to plcm-active-call
  - New resource plcm-active-call-v3 has been added.
- New fields added to plcm-active-call-v3
  - The IP address
  - Port
  - Policy name
  - Dial Plan
  - Dial Rule

## **plcm-backup**

- Changes to plcm-backup
  - plcm-backup does not display file types which are in binary format.
- Changes to plcm-backup-resource:
  - Added new GET method to resource properties with PATH:  
RESTORE\_STATUS\_SUBRESOURCE\_PATH and variable defined as <backup-name>/restore.
- Changes to plcm-restore-status:
  - New status for the last restore operation has been added with enumeration status type as; UNKNOWN, SUCCESS, USER\_ACTION\_REQUIRED and FAILURE.
  - A new FILE NAME string has been added to restore the backup.
  - A unique Entity Tag (ETag) with a unique value generated from the server object instance has been added. This value is the same that MUST be applied to the HTTP (ETag) header for a single instance of this object.

## **plcm-port-range-settings**

The following parameters have been added to the plcm-port-range-call-settings resource:

- Call Settings
  - Method Name: GET
    - You can retrieve the calls settings regarding port ranges. This may be a real stored value, or it may be calculated from other system attributes.
  - Method Name: PUT
    - Updates the calls settings with respect to port ranges. i.e. an attempt is made so that port range sizes for various call related services (media/RTP, SIP, H323, etc.) are adjusted to accommodate the selected calls settings. Maximum call values less than 1 may result in a reset to defaults or may be ignored.
- A new max-call-count integer has been added to represent maximum call count where range for H323 signaling port must be at least 100 and range for media port must be at least 1000.
- System Ephemeral
  - Method Name: GET
    - Retrieves the system ephemeral port range settings.
  - Method Name: PUT
    - Sets the system ephemeral port range settings. A restart is required for settings to take full effect, but one is not performed because of this call.
- end-range:
- The start and end port range are integers from 0 to 65535.entity-tag:
  - A new string has been added to with character range from 1 to 64.

## **plcm-power-management**

- Added a new boolean to specify the various phases of DMA. These can be recognized as:
  - Shutdown to perform advance diagnostic procedures
  - Restarted to perform advance diagnostic procedures

## **plcm-registration-policy-debug-request**

- Changes to reg-whitelist-addresses:
  - Endpoint addresses that registrations are only allowed from or blank. This array can contain multiple values separated with commas.
- Changes to reg-whitelist-aliases:
  - Endpoint aliases that registrations are only allowed from or blank.

## **plcm-registration-policy-v2**

- Changes to inactive-registration-days-to-delete:

- Specify the endpoints with INACTIVE status whose registration has expired.
- Changes to registration-compliance-script:
  - Contains the registration policy script that the admin wants to apply.
- Changes to signaling-ports:
  - The signaling port assignments for this registration policy is in the format: [h323 | sip] :<port>:[private | public]

### **plcm-call**

- Changes to node-ip, policy-name, dial-plan and dial-rule:
  - The IP address of the source node is not populated if this call is returned as part of a list.

### **plcm-security-settings**

- Changes to plcm-security-settings-v2 and plcm-policy-settings-v3:
  - A new integer ssh-idle-timeout has been added to set the number of seconds before SSH connection times out. The value must be greater than 0.

### **plcm-gatekeeper-config-v2**

- dial-plan-public:
  - The name or UID of the dial plan to use for H.323 calls with endpoints in the public network. If not supplied the default authorized dial plan will be assigned.
- acl-id:
  - Changed from a mandatory field to a non-mandatory field.
- acl-id-public:
  - The UID for the H.323 calls with endpoints in private as well as in public network.
- h460-enabled:
  - A new boolean has been added to enable for H460 NAT traversal if TRUE.
- h460-registration-interval:
  - A new integer has been added to monitor registration expiration interval in seconds for registrations using H460. This keeps a RAS pinhole open in the endpoints firewall.
- registration-policy-id-public:
  - A new string has been added to identify the UID of registration policy for listening points in public network.

### **plcm-gatekeeper-config**

- dial-plan and dial-plan-public:
  - This field uses H.323 as the name of the UID in the private network. If not supplied the default authorized dial plan will be assigned.

## plcm-troubleshooting-utilities

- A new policy string has been added with method name GET and PING id.
- command-name:
  - A new string added to display the command invoked.
- command-output:
  - A new string has been added to display the output invoked.
- command-exit-code:
  - A new integer has been added to display the atom links provided in the body of the resource.

## plcm-registration-summary

- Following enumerations were added to plcm-registration-summary:
  - The type of device used for registration returns EXTERNAL\_SIP\_PEER
  - Authentication-status field returns SIP\_AUTHENTICATION\_NOT\_CHALLENGED

## ***RealPresence Platform API Version 3.5.0***

The following are the changes between RealPresence Platform API version 3.5.0 and 3.4.0.

### **Changes affecting multiple resources**

Several general changes that affect multiple resources were introduced in 3.5.0. Changes described here and any resources that implemented these changes are called out specifically in their own sections.

#### **1. *Paging***

A paging mechanism was added for GET calls that retrieve a list of resources. Two new query parameters were added to these GET methods: *limit* and *page*. Use *limit* to specify the maximum number of entries to be viewed in one page, use *page* to specify which page number to be viewed.

#### **2. *Default values and overriding defaults***

In the RealPresence DMA system API, we need a way to tell whether a value is a system default or has been overridden with a user-supplied value. The general design is to add a corresponding Boolean field to the API for each field where this distinction is needed. The new field is called *override-default-{corresponding-field}*; when set to true, the *corresponding-field* should be specified in the API and not pulled from the system default.

Default values are made available in the API so that the client does not need to do any computation to present the option that would be chosen if the *override-default* setting is false. The default values for each resource can be retrieved by a *GET* request, these calls return an object containing the current default values for all the necessary fields.

## plcm-acl

New for this release, this resource collection allows you to configure and view ACL policies.

## plcm-active-call

- Added paging support to the GET call to retrieve the list of active calls ([https://<RPP\\_IP>:8443/api/rest/active-calls](https://<RPP_IP>:8443/api/rest/active-calls)).
- Added new GET method to retrieve active call statistics ([https://<RPP\\_IP>:8443/api/rest/active-calls/statistics](https://<RPP_IP>:8443/api/rest/active-calls/statistics)).

## plcm-alerting-config

- Added new GET method to retrieve alerting configuration default values ([https://<RPP\\_IP>:8443/api/rest/alerting-config/defaults](https://<RPP_IP>:8443/api/rest/alerting-config/defaults)).

## plcm-alert

- Added paging mechanism to the call to retrieve the list of live alerts ([https://<RPP\\_IP>:8443/api/rest/alerts](https://<RPP_IP>:8443/api/rest/alerts)).
- Change to plcm-alert:
  - New field added:
    - cluster-name (string).

## plcm-backup

New for this release, this resource collection allows you to manage backup and restore. This includes getting a list of backups, create, delete, upload, download, and restore a backup.

The following resources were moved from plcm-remote-backup:

- plcm-local-backup.
- plcm-local-backup-list.

## plcm-billing

- The GET method to retrieve a list of CDRs [https://<RPP\\_IP>:8443/api/rest/billing](https://<RPP_IP>:8443/api/rest/billing) has been deprecated and replaced by [/billing/generate-cdrs](#) and [/billing/download-cdrs](#).
- Added POST [https://<RPP\\_IP>:8443/api/rest/billing/generate-cdrs](https://<RPP_IP>:8443/api/rest/billing/generate-cdrs) to initiate the process to generate CDRs.
- Added GET [https://<RPP\\_IP>:8443/api/rest/billing/status](https://<RPP_IP>:8443/api/rest/billing/status) to retrieve CDRs generation status.
- Added GET [https://<RPP\\_IP>:8443/api/rest/billing/download-cdrs](https://<RPP_IP>:8443/api/rest/billing/download-cdrs) to download CDR file.

## plcm-caa-tenant

- Added new GET method to retrieve default values for CAA tenant ([https://<RPP\\_IP>:8443/api/rest/caa-tenant/default](https://<RPP_IP>:8443/api/rest/caa-tenant/default)).
- The following new resources were added:
  - plcm-caa-tenant-v2.
  - plcm-caa-tenant-list-v2.
  - plcm-caa-tenant-defaults.



## plcm-common

- New for this release, this module defines resources shared by other resource collections. Examples of resources defined here: plcm-phys-address, plcm-mac-address, plcm-host, uuid, plcm-fqdn, etc.

## plcm-conference-factory

- New for this release, this resource collection allows you to configure and view SIP conference factories.

## plcm-conference-room

- Added paging support to the GET method to retrieve a list of conference rooms ([https://<RPP\\_IP>:8443/api/rest/conference-rooms](https://<RPP_IP>:8443/api/rest/conference-rooms)).
- Added new GET method to retrieve default values for conference room ([https://<RPP\\_IP>:8443/api/rest/conference-rooms/{conference-room-identifier}/{domain}/{username}/default](https://<RPP_IP>:8443/api/rest/conference-rooms/{conference-room-identifier}/{domain}/{username}/default)).
- The following new resources were added:
  - plcm-conference-room-v4.
  - plcm-conference-room-list-v4.
  - plcm-conference-room-defaults.
- New fields added to plcm-conference-room and plcm-conference-room-v2:
  - focus-uri (string).
  - conference-room-type (enumeration).
- Changes to plcm-conference-room-v3:
  - New field: conference-room-type (enumeration).
  - String length for the field “owner-first-name” increased to 256 characters.
  - String length for the field “owner-last-name” increased to 256 characters.

## plcm-conference-settings

- New fields added to plcm-conference-settings-v3:
  - min-generated-dynamic-conf-id (int64).
  - max-generated-dynamic-conf-id (int64).
  - min-ports-to-start (int32).
  - max-percent-to-start (decimal64).
  - skype-roster-cascade-indicator (string).

## plcm-conference

- Added paging support to the GET method to retrieve a list of conferences ([https://<RPP\\_IP>:8443/api/rest/conferences](https://<RPP_IP>:8443/api/rest/conferences)).
- Added new GET method to retrieve conference manager usage details ([https://<RPP\\_IP>:8443/api/rest/conferences/usage](https://<RPP_IP>:8443/api/rest/conferences/usage)).
- New resource added:
  - plcm-conference-manager-usage.

## plcm-conference-template

- Added new query parameter “type” to the PUT request to update a conference room template ([https://<RPP\\_IP>:8443/api/rest/conference-templates/{value}?type={name|identifier}](https://<RPP_IP>:8443/api/rest/conference-templates/{value}?type={name|identifier})).
- Added new query parameter “type” to the DELETE request to remove a conference room template ([https://<RPP\\_IP>:8443/api/rest/conference-templates/{value}?type={name|identifier}](https://<RPP_IP>:8443/api/rest/conference-templates/{value}?type={name|identifier})).
- Change in plcm-conference-template-v3 through v6:
  - New value added to enumeration conference-mode-experience: OPTIMIZED\_FOR\_LINE\_RATE.
- Changes in plcm-conference-template-v7:
  - New value added to enumeration conference-mode-experience: OPTIMIZED\_FOR\_LINE\_RATE.
  - New fields added:
    - ❖ cascading-for-svc (boolean).
    - ❖ panoramic-layout (boolean).
    - ❖ layout-indication-position (enumeration).
    - ❖ audio-participants-indication (boolean).
    - ❖ video-participants-indication (boolean).
    - ❖ layout-indication-audio-display-mode (enumeration).
    - ❖ indication-duration (int32).
    - ❖ network-quality-indication (boolean).
    - ❖ play-recording-message (boolean).
    - ❖ message-overlay-enable (boolean).
    - ❖ message-overlay-content (string).
    - ❖ message-overlay-font-size (int32).
    - ❖ message-overlay-color (enumeration).
    - ❖ message-overlay-vertical-position (int32).
    - ❖ message-overlay-transparency (int32).
    - ❖ message-overlay-display-repetition (int32).
    - ❖ message-overlay-display-speed (enumeration).
    - ❖ dial-rule-count (int32).
    - ❖ conference-room-count (int32).
    - ❖ enterprise-group-count (int32).

## plcm-config

- New fields added to plcm-call-server-v2:
  - options-ping-interval (int32).
  - options-ping-failure-codes (list).
  - proxy-failure-codes (list).
  - next-ext-proxy-timeout (int32).
  - blacklist-enabled (boolean).
  - grq-quarantine (boolean).
  - grq-ignore-vbp (boolean).

- rrq-quarantine (boolean).
- rrq-ignore-vbp (boolean).
- cleaner-task-interval-minute (int32).
- grq-duplicate-count (int32).
- grq-duplicate-interval-ms (int32).
- rrq-duplicate-count (int32).
- rrq-duplicate-interval-ms (int32).
- New resource added: plcm-logging-v2.
- New field added to plcm-logging:
  - advanced-diagnostics (boolean).
- New fields added to plcm-physical-limits-v2:
  - cluster-name (string).
  - cluster-identifier (string).

## plcm-device

- Changes to the GET method to retrieve a list of devices ([https://<RPP\\_IP>:8443/api/rest/devices](https://<RPP_IP>:8443/api/rest/devices)):
  - Added paging support.
  - Added new query parameter “exact-search” to specify whether the search should be limited to exact matches (vs. wildcard search).
- The following new resources were added:
  - plcm-device-defaults.
  - plcm-device-list-v4.
  - plcm-device-v4.
  - plcm-h323-alias-type-v2.
  - plcm-h323-identity-v2.
- New fields added to plcm-device-v3:
  - path-header (leaf-list).
  - sip-username (string).
  - authentication-name (string).
- Change to plcm-dial-rule-preliminary:
  - The field “script” is now optional.
- New field added to plcm-gateway-profile:
  - read-only (boolean).

## plcm-dial-plan

- Added POST method to create a new dial plan ([https://<RPP\\_IP>:8443/api/rest/dial-plans](https://<RPP_IP>:8443/api/rest/dial-plans)).
- Added GET method to retrieve default values for dial rule ([https://<RPP\\_IP>:8443/api/rest/dial-plans/dial-rules/default](https://<RPP_IP>:8443/api/rest/dial-plans/dial-rules/default)).
- Added PUT method to update a dial plan ([https://<RPP\\_IP>:8443/api/rest/dial-plans](https://<RPP_IP>:8443/api/rest/dial-plans)).
- Added DELETE method to delete a dial plan ([https://<RPP\\_IP>:8443/api/rest/dial-plans](https://<RPP_IP>:8443/api/rest/dial-plans)).
- The following new resources were added:

- plcm-dial-rule-defaults.
- plcm-dial-rule-list-v2.
- plcm-dial-rule-v2.
- New field added to plcm-dial-plan:
  - listening-points (leaf-list).
- Change to plcm-dial-rule:
  - New string value that can be set in the field “action”: RESOLVE\_SIP\_CONF\_FACTORY.

## plcm-dma-supercluster

- The POST method to join a RealPresence DMA system cluster was deprecated ([https://<RPP\\_IP>:8443/api/rest/dma-supercluster](https://<RPP_IP>:8443/api/rest/dma-supercluster) using request resource plcm-dma-supercluster-join), this POST method is replaced by:
  - POST to ([https://<RPP\\_IP>:8443/api/rest/dma-supercluster](https://<RPP_IP>:8443/api/rest/dma-supercluster) using request resource plcm-dma-send-invite-to-join-supercluster).
- The following new resources were added:
  - plcm-dma-send-invite-to-join-supercluster.
  - plcm-dma-supercluster-cluster-server.
- New fields added in dma-supercluster-cluster:
  - proxias-version (string).
  - application-version (string).
  - cluster-reachable (boolean).
  - host-name (string).
  - hardware-model (string).
  - serial-number (string).
  - up-time (string).
  - local-time-zone (string).
  - external-time-source (boolean).
  - active-management-host (string).
  - private-link-up (boolean).
  - private-link-duplex (string).
  - private-link-speed (int32).
  - management-link-up (boolean).
  - management-link-duplex (string).
  - management-link-speed (int32).
  - signaling-link-up (boolean).
  - signaling-link-duplex (string).
  - signaling-link-speed (int32).
  - split-network (boolean).
  - network-interface (list).
  - cpu-core-count (leaf).
  - cpu-utilization (int32).
  - total-memory (int32).

- free-memory (int32).
- total-swap (int32).
- buffer-memory (int32).
- cache-memory (int32).
- total-disk-space (int32).
- free-disk-space (int32).
- used-disk-space (int32).
- used-disk-percentage (int32).
- total-log-space (int32).
- used-log-space (int32).
- used-log-space-percentage (int32).
- next-log-purge-date (date-time).
- List of plcm-territory-v2.
- List of plcm-dma-supercluster-cluster-server.

### **plcm-enterprise-directory-cache**

- Added GET method to retrieve Enterprise Directory Cache defaults ([https://<RPP\\_IP>:8443/api/rest/config/enterprise-directory-cache/defaults](https://<RPP_IP>:8443/api/rest/config/enterprise-directory-cache/defaults)).
- New field added to plcm-enterprise-directory-cache-status:
  - duplicate-passcode-errors (int32).

### **plcm-enterprise-directory-reports**

New for this release, this resource collection allows users to retrieve a variety of error reports including Enterprise Directory conference room errors, passcode errors, orphaned groups and users reports, as well as Enterprise Directory integration report.

### **plcm-enterprise-directory**

- Added GET method to retrieve Enterprise Directory Integration defaults ([https://<RPP\\_IP>:8443/api/rest/config/enterprise-directory/defaults](https://<RPP_IP>:8443/api/rest/config/enterprise-directory/defaults)).
- New resource added:
  - plcm-enterprise-directory-config-v2.
- New field added to plcm-enterprise-directory-status:
  - connected (boolean).

### **plcm-enterprise-group**

New for this release, this resource collection allows users to retrieve groups from the Enterprise Directory server; then import, update, and delete those groups on the RPP server.

### **plcm-eula**

- New field added to plcm-eula-list:
  - entity-tag (string).

## plcm-ext-sip-peer

- Added GET method to retrieve the plcm-postliminary-script.xml ([https://<RPP\\_IP>:8443/api/rest/ext-sip-peer/postliminary-script](https://<RPP_IP>:8443/api/rest/ext-sip-peer/postliminary-script)).
- Added GET method to retrieve the external SIP peers ([https://<RPP\\_IP>:8443/api/rest/ext-sip-peer/status](https://<RPP_IP>:8443/api/rest/ext-sip-peer/status)).
- Added GET method to retrieve default values for external SIP peers ([https://<RPP\\_IP>:8443/api/rest/ext-sip-peer/defaults](https://<RPP_IP>:8443/api/rest/ext-sip-peer/defaults)).
- New resources added:
  - plcm-ext-sip-peer-defaults.
  - plcm-next-hop-address.
  - plcm-postliminary-script.
- Changes to plcm-ext-sip-peer, plcm-ext-sip-peer-v2, v3:
  - The field “password” in the “authentication” list is now optional.
  - New field added: challenge-peer (boolean).
- Changes to plcm-ext-sip-peer-v4:
  - The field “password” in the “authentication” list is now optional.
  - New fields added:
    - ❖ challenge-peer (boolean).
    - ❖ options-ping (boolean).
    - ❖ trusted-application-gruu-enabled (boolean).
    - ❖ lync-mcu-pool-order-id-override (boolean).
    - ❖ List of plcm-next-hop-address.

## plcm-hunt-group

- New field added to plcm-signaling-alias-type:
  - entity-tag (string).

## plcm-license

New for this release, this resource collection allows users to configure the license authority, get licenses update from the license server, apply CFS license, and retrieve the license status.

## plcm-ext-sip-peer

- Added POST method to initiate the process to take a snapshot of the logs ([https://<RPP\\_IP>:8443/api/rest/log-archives/snapshot-logs{server-name}](https://<RPP_IP>:8443/api/rest/log-archives/snapshot-logs{server-name})).
- New resources added:
  - plcm-log-archive-list-v2.
  - plcm-log-archive-metadata.
- New field added to plcm-log-archive:
  - List of plcm-log-archive-metadata.

## plcm-login-policy

- New field added to plcm-whitelist-config:

- snmp-port (int32).

### **plcm-login-sessions**

- New field added to plcm-login-session:
  - client-passthru (string).

### **plcm-mcu-pool-order**

- New fields added to plcm-mcu-pool-order-v2:
  - used-by-enterprise-groups (int32).
  - used-by-conference-rooms (int32).
  - used-by-sip-peer (boolean).
  - used-by-sip-conference-factories (int32).
  - used-by-on-premise-lync-dial-rule (boolean).

### **plcm-mcu**

- Added GET method to retrieve an MCU of type plcm-mcu-v7 ([https://<RPP\\_IP>:8443/api/rest/mcus/{value}/mcu-status](https://<RPP_IP>:8443/api/rest/mcus/{value}/mcu-status)).
- Added GET method to retrieve default values for MCUs ([https://<RPP\\_IP>:8443/api/rest/mcus/default](https://<RPP_IP>:8443/api/rest/mcus/default)).
- New resources added:
  - plcm-mcu-defaults.
  - plcm-mcu-list-v7.
  - plcm-mcu-v7.
- New fields added to plcm-mcu-v6:
  - supports-svc-cascade (boolean).
  - supports-in-conference-ivr (boolean).
  - cif-to-hd-ratio (decimal64).

### **plcm-network**

New for this release, this resource collection allows users to retrieve and modify the current network configuration.

### **plcm-pkix**

New for this release, this resource collection allows users to manage certificates, certificate signing requests (CSRs), and OCSP settings for the RPP system.

### **plcm-registration-policy**

- Added GET method to retrieve default values for REALPRESENCE DMA SYSTEM registration policy ([https://<RPP\\_IP>:8443/api/rest/registration-policy/default](https://<RPP_IP>:8443/api/rest/registration-policy/default)).
- New resource added:
  - plcm-registration-policy-defaults.

## plcm-remote-backup

- The following resources were moved to plcm-backup:
  - plcm-local-backup.
  - plcm-local-backup-list.

## plcm-reports

- Changes listed below:
  - Changes of query parameters from string to DateTime:
    - ❖ start-after
    - ❖ end-after
    - ❖ start-before
    - ❖ end-before
  - Paging support added.

Changes were applied to the following GET methods:

- Retrieve a list of alerts: [https://<RPP\\_IP>:8443/api/rest/reports/alerts](https://<RPP_IP>:8443/api/rest/reports/alerts).
  - Retrieve a list of registration summary: [https://<RPP\\_IP>:8443/api/rest/reports/registrations](https://<RPP_IP>:8443/api/rest/reports/registrations).
  - Retrieve a list of conference summary: [https://<RPP\\_IP>:8443/api/rest/reports/conferences](https://<RPP_IP>:8443/api/rest/reports/conferences).
  - Retrieve a list of call history: [https://<RPP\\_IP>:8443/api/rest/reports/calls](https://<RPP_IP>:8443/api/rest/reports/calls).
- The following GET methods also support paging:
    - Retrieve a list of orphaned groups: [https://<RPP\\_IP>:8443/api/rest/reports/orphaned-groups](https://<RPP_IP>:8443/api/rest/reports/orphaned-groups).
    - Retrieve a list of orphaned users: [https://<RPP\\_IP>:8443/api/rest/reports/orphaned-users](https://<RPP_IP>:8443/api/rest/reports/orphaned-users).
    - Retrieve a list of call events for a call: [https://<RPP\\_IP>:8443/api/rest/reports/calls/{call-identifier}/call-events](https://<RPP_IP>:8443/api/rest/reports/calls/{call-identifier}/call-events).
    - Retrieve a list of property changes for a call: [https://<RPP\\_IP>:8443/api/rest/reports/calls/{call-identifier}/property-changes](https://<RPP_IP>:8443/api/rest/reports/calls/{call-identifier}/property-changes).
    - Retrieve a list of subscription events for a call: [https://<RPP\\_IP>:8443/api/rest/reports/calls/{call-identifier}/subscription-events](https://<RPP_IP>:8443/api/rest/reports/calls/{call-identifier}/subscription-events).
  - Changes to the GET method to retrieve the export report of CDR data ([https://<RPP\\_IP>:8443/api/rest/reports/export-report](https://<RPP_IP>:8443/api/rest/reports/export-report)):
    - New field added:
      - ❖ export-type (string).
    - This method now supports paging.
  - New GET method to retrieve historical network usage data about the video network ([https://<RPP\\_IP>:8443/api/rest/reports/network-usage](https://<RPP_IP>:8443/api/rest/reports/network-usage)).
  - New GET method to retrieve export network usage data to download as CSV ([https://<RPP\\_IP>:8443/api/rest/reports/network-usage/export](https://<RPP_IP>:8443/api/rest/reports/network-usage/export)).
  - New GET method to retrieve a list of CDRs for the input call UUIDs ([https://<RPP\\_IP>:8443/api/rest/reports/calls/audit-info](https://<RPP_IP>:8443/api/rest/reports/calls/audit-info)).
  - New resources added:
    - plcm-network-usage-list.
    - plcm-network-usage.



- New field added to plcm-call:
  - fully-external (boolean).
- New fields added to plcm-conferences-statistics:
  - cluster-name (string).
  - cluster-identifier (string).
- New fields added to plcm-export-report:
  - start-after (date-time).
  - end-before (date-time).
- New fields added to plcm-throttle-point-reference:
  - type (string).
  - owner-site-name (string).
  - bandwidth-event-time (date-time).

## plcm-rprm-integration

New for this release, this resource collection allows users to retrieve the RealPresence Resource Manager Integration information and leave the integration.

## plcm-security-settings

- New resource added:
  - plcm-security-settings-v2.

## plcm-signaling-config

- The following new query parameters:
  - ClusterName (string).
  - ClusterIdentifier (string).
- were added to the following GET methods:
  - Retrieve the SIP configuration: [https://<RPP\\_IP>:8443/api/rest/signaling/sip](https://<RPP_IP>:8443/api/rest/signaling/sip).
  - Retrieve the gatekeeper configuration: [https://<RPP\\_IP>:8443/api/rest/signaling/gatekeeper](https://<RPP_IP>:8443/api/rest/signaling/gatekeeper).
  - Retrieve the WebRTC configuration: [https://<RPP\\_IP>:8443/api/rest/signaling/webrtc](https://<RPP_IP>:8443/api/rest/signaling/webrtc).
- New GET method to retrieve the default SIP configuration ([https://<RPP\\_IP>:8443/api/rest/signaling/sip/default](https://<RPP_IP>:8443/api/rest/signaling/sip/default)).
- New GET method to retrieve the default gatekeeper configuration ([https://<RPP\\_IP>:8443/api/rest/signaling/gatekeeper/default](https://<RPP_IP>:8443/api/rest/signaling/gatekeeper/default)).
- New GET method to retrieve the default WebRTC configuration ([https://<RPP\\_IP>:8443/api/rest/signaling/webrtc/default](https://<RPP_IP>:8443/api/rest/signaling/webrtc/default)).
- New GET method to retrieve the signaling status ([https://<RPP\\_IP>:8443/api/rest/signaling/status](https://<RPP_IP>:8443/api/rest/signaling/status)).
- New GET and PUT methods to retrieve and update ACL variables ([https://<RPP\\_IP>:8443/api/rest/signaling/acl/variables](https://<RPP_IP>:8443/api/rest/signaling/acl/variables)).
- New resources added:
  - plcm-acl-variable-list.
  - plcm-acl-variables.
  - plcm-acl-variable.

- plcm-signaling-status.
- plcm-sip-config-v2.
- New fields added to plcm-gatekeeper-config:
  - dial-plan (string).
  - internal-gk-active (boolean).
  - cluster-name (string).
  - cluster-identifier (string).
- New fields added to plcm-protocol-listening-point:
  - dial-plan (string).
  - enabled (boolean).
  - applySystemAcl (boolean).
- New fields added to plcm-sip-config:
  - cluster-name (string).
  - cluster-identifier (string).
- New fields added to plcm-webrtc-config:
  - dial-plan (string).
  - cluster-name (string).
  - cluster-identifier (string).

## plcm-snmp-config

- New GET method to retrieve a list of Polycom MIB files ([https://<RPP\\_IP>:8443/api/rest/snmp-config/mibs](https://<RPP_IP>:8443/api/rest/snmp-config/mibs)).
- New GET method to download a specific Polycom MIB ([https://<RPP\\_IP>:8443/api/rest/snmp-config/mibs/{mib-filename}](https://<RPP_IP>:8443/api/rest/snmp-config/mibs/{mib-filename})).
- New resource added:
  - plcm-mib-list.

## plcm-software-upgrade

New for this release, this resource collection allows users to retrieve details of current software version and history of installed software versions.

## plcm-task

- New field added to plcm-task:
  - task-type (enumeration).

## plcm-territory

- New resources added:
  - plcm-territory-list-v3.
  - plcm-territory-v3.
- New fields added to plcm-territory-v2:
  - calendaring-enabled (boolean).
  - enterprise-directory (boolean).

- call-server-sites (boolean)
- site-name (leaf-list).

## plcm-troubleshooting-utilities

- New GET method to do a traceroute on an IP address ([https://<RPP\\_IP>:8443/api/rest/troubleshooting-utilities/traceroute/{ip-address}](https://<RPP_IP>:8443/api/rest/troubleshooting-utilities/traceroute/{ip-address})).
- New GET method to start a DTO consistency check ([https://<RPP\\_IP>:8443/api/rest/troubleshooting-utilities/dto-consistency-check](https://<RPP_IP>:8443/api/rest/troubleshooting-utilities/dto-consistency-check)).
- New GET methods to get information of the last network packet capture and to retrieve capture data ([https://<RPP\\_IP>:8443/api/rest/troubleshooting-utilities/network-packet-capture](https://<RPP_IP>:8443/api/rest/troubleshooting-utilities/network-packet-capture)).
- New POST method to start a network packet capture ([https://<RPP\\_IP>:8443/api/rest/troubleshooting-utilities/network-packet-capture/start](https://<RPP_IP>:8443/api/rest/troubleshooting-utilities/network-packet-capture/start)).
- New POST method to stop a network packet capture ([https://<RPP\\_IP>:8443/api/rest/troubleshooting-utilities/network-packet-capture/stop](https://<RPP_IP>:8443/api/rest/troubleshooting-utilities/network-packet-capture/stop)).
- New POST method to reset the RPP system back to its factory default settings ([https://<RPP\\_IP>:8443/api/rest/troubleshooting-utilities/reset-to-defaults](https://<RPP_IP>:8443/api/rest/troubleshooting-utilities/reset-to-defaults)).
- New resources added:
  - plcm-network-packet-capture-request.
  - plcm-network-packet-capture.

## plcm-user-domain

New for this release, this resource collection allows users to get the domain list.

## plcm-user

- Changes to GET method to retrieve a list of users ([https://<RPP\\_IP>:8443/api/rest/users](https://<RPP_IP>:8443/api/rest/users)).
  - New query parameters:
    - ❖ user (string).
    - ❖ conference-room-id (string).
    - ❖ custom-conference-rooms-only (string).
  - Added paging support.
- New DELETE method to log out the current user ([https://<RPP\\_IP>:8443/api/rest/users/self/logout](https://<RPP_IP>:8443/api/rest/users/self/logout)).
- New GET method to retrieve the login history of current user ([https://<RPP\\_IP>:8443/api/rest/users/self/login-history](https://<RPP_IP>:8443/api/rest/users/self/login-history)).
- New GET method to retrieve the default values for user ([https://<RPP\\_IP>:8443/api/rest/users/default](https://<RPP_IP>:8443/api/rest/users/default)).
- New resources added:
  - plcm-user-defaults.
  - plcm-user-list-v4.
  - plcm-user-v4.
  - plcm-user-login-history.
- Changes to plcm-user-v3:
  - New fields added:
    - ❖ enterprise-chairperson-passcode (string).

- ❖ enterprise-conference-passcode (string)
- ❖ territory-name (string).
- ❖ conference-room (list).
- Field changes:
  - ❖ Max length for chairperson-passcode is 16.
  - ❖ Max length for conference-passcode is 16.

### **plcm-user-domain**

New for this release, this resource collection allows users to retrieve, update, and delete VPN configurations.

## ***RealPresence Platform API Version 3.4.0***

The following are the changes between RealPresence Platform API version 3.4.0 and 3.2.1.

### **plcm-conference-template**

- The enum value PARTICIPANTS\_PRIORITY has been added to all versions of the plcm-conference-template resource's *telepresence-layout-mode* field.
- Added new fields in the plcm-conference-template-v7 resource to support Microsoft RDP content sharing and panoramic layout:
  - enable-ms-rdp-content (boolean).
  - panoramic-layout (boolean).

### **plcm-mcu**

- Added new field in the plcm-mcu-v6 resource to support Microsoft RDP content sharing:
  - supports-ms-rdp-content (boolean).

### **plcm-conference-room**

- Added new fields in the plcm-conference-room-v3 resource to support Polycom One Touch Dial App for Microsoft Office 365:
  - focus-uri (string).
  - lync-registered-domain (string).

## ***RealPresence Platform API Version 3.2.1***

The following are the changes between RealPresence Platform API version 2.7.1 and 3.2.1.

### **plcm-conference**

The resources plcm-conference-list-v4 and plcm-conference-v4 resources have been added. The plcm-conference-list-v4 can be queried by device identifier.

### **plcm-provision**

New for this release, this resource collection allows you to config admin-config-provisioning-profile and network-provisioning-profile and retrieve information for them.

## **plcm-network-device**

New for this release, the resource `plcm-dma-local-cluster` can be added and retrieved.

## **plcm-enterprise-directory**

New for this release, this resource collection allows you to config enterprise directory and retrieve information about it.

## **RealPresence Platform API Version 2.7.1**

The following are the changes between RealPresence Platform API version 2.6.2 and 2.7.1.

### **plcm-conference**

In the `plcm-participant-v2` resource, the `dtmf-suffix` field now accepts the \* (asterisk) character.

### **plcm-conference-room**

In the `plcm-dial-out-participant` and `plcm-dial-out-participant-v2` resources, the `dtmf-suffix` field now accepts the \* (asterisk) character.

### **plcm-log-archive**

New for this release, this resource collection allows you to initiate the process to roll logs and retrieve information about log archives.

### **plcm-presence-config**

New for this release, this resource collection allows you to configure Microsoft Lync presence publishing.

## **Improved Validation Error Messages**

As part of this release, validation error messages have become more user-friendly. If you omit a mandatory field from a PUT or POST payload, validation error messages now specify clearly which field is missing. Unrecognized fields that you include with a PUT or POST payload are now silently discarded, and do not produce an error message.

## **JSON Support for the API**

JSON (JavaScript Object Notation) is now available for use with the API as an alternative to XML. It can be used as a payload format anywhere XML is used.

To enable JSON, use the `+json` media type suffix instead of the `+xml` media type suffix in the 'Accept' and 'Content-Type' headers. For example, to specify XML for the `plcm-conference-room-v2` resource, use the following request format:

```
Accept: application/vnd.plcm.plcm-conference-room-v2+xml
```

To specify JSON for the same request, use the following format:

```
Accept: application/vnd.plcm.plcm-conference-room-v2+json
```

If you do not specify an 'Accept' header in your requests, XML is used as the default payload type.

Note that there is a change in the way the server handles requests. Previously, the media type negotiation in the 'Accept' header was not enforced and could be omitted. Beginning with this version of the RealPresence DMA system API, you must specify the media type and version in the 'Accept' header for GET requests. If the 'Accept' header is omitted entirely, the server will choose a media type and

version on behalf of the client, even if the media type and version it chooses is not compatible with the client. Polycom strongly recommends using the 'Accept' header in all GET requests.

## ***RealPresence Platform API Version 2.6.2***

The following are the changes between RealPresence Platform API version 2.6.1 and 2.6.2.

### **plcm-conference**

- The resources plcm-participant-v2, plcm-participant-list-v2, and plcm-participant-notification-v2 have been added.
- The plcm-dial-out-participant-v2 resource supports the following new fields:
  - audio-only (boolean).
  - dtmf-suffix (string).
  - Encrypted-media (boolean).
  - auto-disconnect (boolean).
- POSTs to the “/api/rest/conferences/{conference-identifier}/participants” object now accept plcm-dial-out-participant-v2 as well as plcm-participant.

### **plcm-conference-room**

The resources plcm-dial-out-participant-v2, plcm-conference-room-v3, and plcm-conference-room-list-v3 have been added.

### **plcm-mcu**

The plcm-mcu-v5 and plcm-mcu-list-v5 resources have been added. They allow you to specify the number of “overall” reserved ports per MCU in cascading scenarios.

## **RealPresence Platform API Version 2.6.1**

The following are the changes between RealPresence Platform API version 2.6.0 and 2.6.1.

- The GET method of `https://<RPP IP>:8443/api/rest/conferences` can retrieve the ongoing and scheduled conferences according to the user's permission. If the user has the Conference Operator permission, the user can see the ongoing and scheduled conferences created by any user. If the user has the Conference Operator (Restricted) permission, the user can only see the ongoing conferences that user has created.
- The GET method of `https://<RPP IP>:8443/api/rest/reservations` is changed to allow for ISDN participants to be included in the retrieved conference since API v2.6.1.

## **RealPresence Platform API Version 2.6.0**

The following are the changes between RealPresence Platform API version 2.5.5 and 2.6.0.

### **plcm-alert**

The `plcm-alert-subscription` resource has been added to this resource collection.

### **plcm-call-server**

The `plcm-call-server-v2` resource has been added, supporting the **Bit rate to bandwidth conversion factor** Call Server setting.

### **plcm-conference**

The `plcm-conference-v2`, `plcm-conference-list-v2`, and `plcm-conference-notification-v2` resources have been added, allowing you to receive information on conference focus URIs associated with Lync conferences.

### **plcm-conference-template**

The `plcm-conference-template-v6` and `plcm-conference-template-list-v6` resources have been added. To support Polycom MCU version 8.5 features, these resources contain the following new fields:

- `site-name-color`.
- `site-name-display-position`.
- `site-name-display-mode`.
- `site-name-font-size`.
- `site-name-horizontal-position`.
- `site-name-text-color`.
- `site-name-transparency`.
- `site-name-vertical-position`.

### **plcm-ext-sip-peer**

- The `plcm-ext-sip-peer-v3` and `plcm-ext-sip-peer-list-v3` resources have been added, containing the additional "RequestUriFormat" value `REQ_MICROSOFT_WITHOUT_CSS`.
- The `plcm-ext-sip-peer-v4` and `plcm-ext-sip-peer-list-v4` resources have been added, allowing you to specify a SIP peer GRUU value for Microsoft Lync integrations.

### **plcm-site**

- The `plcm-data-rate` resource has been added, allowing you to specify the data rate of the bandwidth data in the object.

- The `plcm-site-list-v4` and `plcm-subnet-list-v3` resources have been added, supporting the `plcm-data-rate` resource.

### **plcm-site-link**

The `plcm-site-v4`, `plcm-site-link-v3`, and `plcm-site-link-list-v3` resources have been added, supporting the `plcm-data-rate` resource.

## ***RealPresence Platform API Version 2.5.5***

### **plcm-mcu**

The `plcm-mcu-v5` and `plcm-mcu-list-v5` resources have been added. They allow you to specify the number of “overall” reserved ports per MCU in cascading scenarios.

## ***RealPresence Platform API Version 2.5.4***

The following are the changes between RealPresence Platform API version 2.5.3 and 2.5.4.

- If you attempt to change the site topology while the RealPresence DMA system is integrated with the RealPresence Resource Manager system, the RealPresence DMA system returns the following status code:

409: Site topology change is not allowed when DMA is integrated with RPRM.

This change affects the following resource collections:

- `plcm-network-cloud`.
- `plcm-site`.
- `plcm-site-link`.
- `plcm-site-to-site-exclusion`.
- `plcm-territory`.

## ***RealPresence Platform API Version 2.5.3***

The following are the changes between RealPresence Platform API version 2.5.2 and 2.5.3.

### **plcm-user**

- The `plcm-user-v3` and `plcm-user-list-v3` resources have been added.
- The `plcm-user-v3` resource adds the new fields *chairperson-passcode* and *conference-passcode*.
- The `plcm-user-list-v3` resource returns a list of `plcm-user-v3` resources.



## **RealPresence Platform API Version 2.5.2**

The following are the changes between RealPresence Platform API version 2.4.1 and 2.5.2.

### **plcm-alert**

New for this release, this resource collection provides access to system alert information.

### **plcm-alerting-config**

New for this release, this resource collection allows system alert configuration.

### **plcm-conference**

- The following new resources have been added:
  - suspend-video-all-except.
  - resume-video-all.
  - mute-all-audio-except-chair.
  - set-join-mute-audio-except-chair.
  - unmute-all-audio-except-chair.
  - set-join-unmute-audio-except-chair.
  - mute-all-video-except-chair.
  - set-join-suspend-video-except-chair.
  - unmute-all-video-except-chair.
  - set-join-resume-video-except-chair.
- The documentation for the `set-join-new-calls-muted` resource has been updated for the **plcm-conference** resource collection.

### **plcm-conference-room**

- The `plcm-conference-room-v2` and `plcm-conference-room-list-v2` resources have been added.
- The 'max-participants' field is now validated in this resource collection.

### **plcm-conference-settings**

- Fields in the `plcm-conference-settings` and `plcm-conference-settings-v2` resources are now validated.
- The documentation for
- Media type has been updated for this resource collection.
- The documentation for status code 403 has been updated for this resource collection.
- New status codes have been documented for this resource collection.

### **plcm-conference-template**

- The `plcm-conference-template-v5` resource has been added.
- New status codes have been added to the documentation for this resource collection.

### **plcm-config**

New resources for call server, history retention, logging, and physical limits configuration have been added.

## **plcm-dma-supercluster**

- New status codes have been added to the documentation for this resource collection.
- The new parameters 'value' and 'type' have been added to the `/value` resource.
- The “password” field in the `plcm-dma-supercluster-join-request` resource is now mandatory.
- The new, optional field “detailed-error-code” has been added to the `plcm-error` resource.

## **plcm-ext-sip-peer**

- Field descriptions have been updated in the `plcm-ext-sip-peer` resource.
- The “password” field is now mandatory.
- New status codes have been added to the documentation for this resource collection.

## **plcm-mcu**

- The optional field “supports-SVC” has been added to the `plcm-mcu-v2` and `plcm-mcu-v3` resources.
- The `plcm-mcu-v4` and `plcm-mcu-list-v4` resources have been added.

## **plcm-mcu-pool**

The `plcm-mcu-pool-v2` and `plcm-mcu-pool-list-v2` resources have been added.

## **plcm-mcu-pool-order**

The `plcm-mcu-pool-order-v2` and `plcm-mcu-pool-order-list-v2` resources have been added.

## **plcm-network-cloud**

The documentation for media type has been updated for this resource collection.

## **plcm-presence-config**

New for this release, this resource collection allows you to configure presence settings.

## **plcm-signaling-config**

New status codes have been added to the documentation for this resource collection.

## **plcm-site**

The `plcm-site-v3` and `plcm-site-list-v3` resources have been added; be sure to use this version of the resources. The `plcm-site-v2` and `plcm-site-list-v2` resources have been deprecated.

## **plcm-site-topology**

The documentation for media type has been updated for this resource collection.

## **plcm-snmp-config**

New for this release, this resource collection allows you to configure SNMP settings.

## **plcm-territory**

The `plcm-territory-v2` and `plcm-territory-list-v2` resources have been added.

## **plcm-time**

New for this release, this resource collection allows you to configure time settings.

## **plcm-user**

- The `plcm-user-v2` and `plcm-user-list-v2` resources have been added.
- New status codes have been added to the documentation for this resource collection.

### **RealPresence Platform API Version 2.4.3**

The following are the changes between RealPresence Platform API version 2.2.0 and 2.4.2.

#### **plcm-device**

The following three parameters have been added to the `plcm-device` resource:

- Username – Return only devices that match the specified username.
- Domain – Return only devices that are part of the specified network domain.
- DeviceModel – Return only devices that match the specified device model.

Using these parameters in a GET request, you can filter the returned device list. For example:

<https://localhost:8443/api/rest/devices?username={username}&domain={domain}&device-model={deviceModel}>

### **RealPresence Platform API Version 2.4.2**

The following are the changes between RealPresence Platform API version 2.2.0 and 2.4.2.

#### **plcm-time**

New for this release, this resource collection allows you to retrieve and change the system's time configuration.

#### **plcm-reservation**

- The enum values `H323_E164`, `H323_ANNEX_O`, and `H323_ID` have been added to the `plcm-reserved-participant` resource's `connection-type-enum` field.
- The `dial-number` field is now strictly validated in the `plcm-reserved-participant` resource. The `dial-number` field must match `connection-type-enum` and `device-id` (if defined).

## ***RealPresence Platform API Version 2.2.0***

The following are the changes between RealPresence Platform API version 2.0.1 and 2.2.0.

### **plcm-snmp-config**

New for this release, this resource collection allows you to retrieve and change the system's SNMP configuration.

### **plcm-log-config**

New for this release, this resource collection allows you to retrieve and change the current logging configuration.

## ***RealPresence Platform API Version 2.0.1***

The following are the changes between RealPresence Platform API version 1.7.7 and 2.0.1.

### **plcm-device**

- The export-inventory resource has been added, allowing you to retrieve information about a device.
- The plcm-device-v2 resource has been added. The *device-type*, *manage-mode*, *serial-number*, *site*, and *plcm-device-status* fields have been added to the plcm-device-v2 and plcm-device-v3 resources.

### **plcm-dial-string-reservation**

New for this release, this resource collection allows you to create, update, retrieve and delete a dial string reservation.

### **plcm-remote-alert-profile**

New for this release, this resource collection allows you to list remote alert profiles and retrieve information about individual remote alert profiles.

### **plcm-reservation**

- The plcm-reservation resource now supports retrieving reservations based on recurrence identifiers.
- The plcm-sched-recurrence resource has been added to this resource collection.
- The fields *user-type-enum*, *mode-enum*, *bit-rate-enum*, and *device-id* have been added to the plcm-reserved-participant resource.

### **plcm-user**

- A link to plcm-dial-string-reservation information has been added to the GET method response of <https://localhost:8443/api/rest/users/{domain}/{username}>.
- The maximum length of the username, first-name, and last-name fields is now 256 characters.
- The *uuid* field has been added to this resource.

## ***RealPresence Platform API Version 1.7.7***

The following are the changes between RealPresence Platform API version 1.7.6 and 1.7.7.

### **plcm-billing**

The following status code has been added to this resource collection:

503: The maximum number of concurrent CDR queries has been reached. Please wait for an existing query to complete.

### **plcm-site**

The following has been added to this resource collection:

409: Site name is limited to 52 characters when Embedded DNS is enabled.

## ***RealPresence Platform API Version 1.7.6***

The following are the changes between RealPresence Platform API version 1.7.5 and 1.7.6.

### **plcm-notification**

The `plcm-subscription-v2` resource has been added. This resource includes a separate list of parameters previously embedded in the `plcm-subscription` resource.

## ***RealPresence Platform API Version 1.7.5***

The following are the changes between RealPresence Platform API version 1.7.4 and 1.7.5.

### **plcm-conference**

The field `audio-mute-by-mcu` has been added to the `plcm-participant` resource.

## ***RealPresence Platform API Version 1.7.4***

The following are the changes between RealPresence Platform API version 1.7.2 and 1.7.4.

### **plcm-conference-room**

In the `plcm-conference-room` resource, the `conference-room-identifier` field now accepts the pattern “[0-9a-zA-Z~!@%\*\\_-\+={}|\\|\\V?. ]\*”. Previously, the field only accepted the pattern “[0-9a-fA-F!~\$]\*”.

## ***RealPresence Platform API Version 1.7.2***

The following are the changes between RealPresence Platform API version 1.7.1 and 1.7.2.

### **plcm-conference-settings**

The width of the fields `min-room-id` and `max-room-id` has been changed from `int32` to `int64` in the `plcm-conference-settings` and `plcm-conference-settings-v2` resources.

### **plcm-conference-template**

The `plcm-conference-template-v4` resource and associated create, read, update, and delete operations have been added.

## ***RealPresence Platform API Version 1.7.1***

The following are the changes between RealPresence Platform API version 1.2.2, shipped with RealPresence system version 5.2.2.6, and RealPresence Platform API version 1.7.1.

### **plcm-conference**

- The `set-chairperson` resource has been added, allowing you to set a participant as a chairperson.

- The new fields *forward-dtmf-source*, *audio-mute-by-endpoint*, and *video-mute-by-endpoint* have been added to the `plcm-participant` resource.
- The field *cascade-link-identifier* is now mandatory for the `plcm-cascade-link` resource.

### **plcm-conference-room**

- In the `plcm-conference-room` resource, the allowed `conference-room-identifier` pattern has been changed from "[0-9a-fA-F!~\$]+" to "[0-9a-fA-F!~\$]\*".
- An error message for an invalid start or end time when creating or updating a conference room has been added.

### **plcm-conference-settings**

New for this release, this resource collection allows you to manipulate default conference settings system-wide.

### **plcm-conference-template**

- The new resource `plcm-conference-template-v3` has been added. This version of the resource adds the fields *audio-only-conference*, *conference-mode-experience*, *as-sip-content*, *conference-chairperson-termination*, and the enum `PREFER_TIP` in the field *tip-compatibility*.
- Support for creating and updating `plcm-conference-template-v3` as well as deleting any version of `plcm-conference-template` has been added.
- The field *conference-chairperson-termination* has been added to the `plcm-conference-template` and `plcm-conference-template-v2` resources.

### **plcm-dma-supercluster**

New for this release, this resource collection allows you to control supercluster operations.

### **plcm-ext-sip-peer**

New for this release, this resource collection allows you to control external SIP peer operations.

### **plcm-mcu**

The `plcm-mcu-v3` resource has been added. This resource adds the *alert-when-mcu-unregisters* field.

### **plcm-network-cloud**

New for this release, this resource collection allows you to control network cloud operations.

### **plcm-signaling-config**

New for this release, this resource collection allows you to control system signaling preferences.

### **plcm-site-link**

New for this release, this resource collection allows you to control site link configuration.

### **plcm-site**

New for this release, this resource collection allows you to configure site preferences.

### **plcm-site-topology**

Support has been added for the new **plcm-site**, **plcm-site-link**, and **plcm-site-to-site-exclusion** resource collections.

### **plcm-site-to-site-exclusion**

New for this release, this resource collection allows you to configure site-to-site exclusions within the site topology.

### **plcm-territory**

Support has been added for creating, updating, and deleting territories.

### **plcm-user**

- The length of the fields *first-name*, *last-name*, and *username* have been changed from 128 to 256 characters.
- The pattern validation for the *username* field has been changed from "[^\\|\\\\|/|:|;|=|+|\*|\\?|@|<|>|"]\*" to "[^:]\*".

# Getting Started

## Understanding Basic Concepts

The Polycom® RealPresence Platform API uses an XML or JSON encoding over HTTPS transport. The API adheres to a Representational State Transfer (REST) architecture. This document assumes that the reader has a basic understanding of XML, JSON, and HTTPS, and is unfamiliar with REST.

### ***Resources and HTTP Methods***

Each data object accessible through the API is modeled as a resource. Some example resources are ***users***, ***conference rooms***, ***conferences***, and ***conference participants***. Each resource is uniquely identified by a URL and is referenced by means of the URL path.

You access the API and manipulate resources by using standard REST conventions. Each API method is invoked using one of the following HTTP methods:

- GET - performs a read operation.  
This operation returns a representation of a resource. GET is guaranteed to be a “safe” operation. It will never modify the state of the system. GET typically takes no input except the resource URL, and typically returns HTTP headers and an XML or JSON document as output.
- POST - performs a create operation.  
This operation creates a new resource. POST is also used for operations that modify the state of the system that are more complex than simply updating a resource state. POST typically takes an XML or JSON document as input and returns only HTTP headers as output.
- PUT - performs an update operation.  
This operation replaces the data for a resource with a new set of data. PUT typically takes an XML or JSON document as input and returns only HTTP headers as output.
- DELETE - performs a delete operation.  
This operation removes a resource from the system. DELETE typically takes no input except the resource URL and returns only HTTP headers as output.

---

#### **Notes**

- All four HTTP methods are not always available for each resource. For example, some resources only support HTTP GET.
- In the case of errors, any of the HTTP methods can return an XML or JSON **plcm-error** document.

### **Recommended HTTP GET Optimization (Conditional GET)**

The use of an HTTP conditional GET ensures that representations are only returned to the API client if the resource has changed. This reduces unnecessary network bandwidth consumption by eliminating the XML or JSON entity body in the response.



As a result, the API client retrieves the Entity Tag (ETag) value from the initial HTTP GET. The API client can now specify this ETag in subsequent GET request's body or ***If-None-Match*** header. The server compares the request ETag value with the server's current resource state.

If you adopt this approach, the following rules apply:

- The <EntityTag> field in the request body is always checked first. If the <EntityTag> field is not present, the Etag in the header is checked.
- If the ETag does not exist in the body or in the ***If-None-Match*** header, always return the full representation with the HTTP status code of: 200 OK.
- If the ETag value exists in the body or in the ***If-None-Match*** header but does not match the ETag of the server resource, then the server assumes that the resource has changed. As a result, the server returns the full representation with the HTTP status code of: 200 OK.
- If the ETag value exists in the body or in the ***If-None-Match*** header, and the values match the ETag of the server resource, then no representation is supplied in the HTTP response entity-body. The server returns with the HTTP status code of: 304 Not Modified.

## Recommended HTTP PUT Request Concurrency Control (Conditional PUT)

The use of an HTTP conditional PUT ensures that resources are correctly modified by preventing lost updates.

As a result, the API client should be required to specify the previously retrieved representation's Entity Tag (ETag) value. This value will be added in the subsequent PUT request's ***If-Match*** header. Alternatively, the ETag can be defined in the entity-body. Either way, one or both ETag value parameters must be defined. The server compares the request ETag value with the server's current resource state.

If you adopt this approach, then the following rules apply:

- The <EntityTag> field in the request body is always checked first. If the <EntityTag> field is not present, the Etag in the header is checked.
- If the ETag does not exist in the body or in the ***If-Match*** header, the server will respond with an error representation (PlcmError) that contains a descriptive message stating that the ETag is required. Also, the response must contain the HTTP status code: 428 Precondition Required.
- If the ETag value exists in the body and/or in the ***If-Match*** header but does not match the ETag of the server resource, the server will respond with an error representation (PlcmError) that contains a descriptive message stating that the resource has changed on the server. Also, the response will contain the HTTP status code: 412 Precondition Failed.
- If the ETag value exists in the body and/or in the ***If-Match*** header, and the values match the ETag of the server resource, then the update can proceed. The HTTP response will contain an ETag header with the updated value, a resource link in the Location header, and an HTTP status code: 204 No Content.

## Updating resources

When you update a resource using the PUT method, it will completely replace the server's representation with the new version passed. For this reason, you should update a resource in the following way:

- GET the resource to be updated and its Etag.
- Change values in the returned representation.
- PUT the representation with the previously obtained Etag.

## ***Simple User Resource Example***

Consider the default 'LOCAL\admin' user as an example.

### **Get Method**

You can perform an HTTP GET request for this resource by typing the URL into a web browser.

The URL of a user is: `https://<server>:<port>/api/rest/users/<domain>/<username>`

On the RealPresence DMA system “carrot” the URL for the local admin user resource is:  
`https://carrot:8443/api/rest/users/LOCAL/admin`

On the RealPresence Resource Manager system “RPRM-01”, the URL for the local user admin resource is: `https://RPRM-01:8443/api/rest/users/LOCAL/admin`

The browser will report a certificate trust issue. This is expected since the server is configured by default to offer an unsigned certificate. Instruct the browser to ignore the issue.

The Authentication Required screen will be displayed. Authentication is required to access the resource. Enter the **username** and **password** of an administrator user. These are the same credentials used to login from the GUI.

The browser will ask you to identify what application to use to display the user resource:

Select **Browse** and choose to open the resource in a browser or text editor.

The browser displays the user resource.

## Response Body:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <ns4:plcm-user xmlns="http://www.w3.org/2005/Atom" xmlns:ns2="urn:com:polycom:api:rest:plcm-
user-role" xmlns:ns3="urn:com:polycom:api:rest:plcm-user-role-list"
xmlns:ns4="urn:com:polycom:api:rest:plcm-user">
  <link title="Self Relationship" type="application/vnd.plcm.plcm-user+xml" rel="self"
href="https://server:port/api/rest/users/LOCAL/admin"/>
  <link title="Owned Conference Rooms" type="application/vnd.plcm.plcm-conference-room+xml"
rel="urn:com:polycom:api:rest:link-relations:owned-conference-rooms"
href="https://server:port/api/rest/conference-rooms?username=admin&domain=LOCAL"/>
  <ns3:plcm-user-role-list>
    <ns2:plcm-user-role>
      <ns2:name>Administrator</ns2:name>
    </ns2:plcm-user-role>
    <ns2:plcm-user-role>
      <ns2:name>Auditor</ns2:name>
    </ns2:plcm-user-role>
    <ns2:plcm-user-role>
      <ns2:name>Provisioner</ns2:name>
    </ns2:plcm-user-role>
  </ns3:plcm-user-role-list>
  <ns4:username>admin</ns4:username>
  <ns4:first-name>Default</ns4:first-name>
  <ns4:last-name>System Administrator User</ns4:last-name>
  <ns4:domain>LOCAL</ns4:domain>
  <ns4:disabled>>false</ns4:disabled>
  <ns4:locked>>false</ns4:locked>
  <ns4:entity-tag>260dc1ffa9ac8d1bafa2d2ba41115bd7</ns4:entity-tag>
</ns4:plcm-user>
```

---

## Response Header:

```
Status Code: 200 OK
Cache-Control: no-cache
Content-Length: 1292
Content-Type: application/vnd.plcm.plcm-user+xml
Date: Fri, 18 May 2012 15:47:09 GMT
Etag: "260dc1ffa9ac8d1bafa2d2ba41115bd7"
Expires: Wed, 31 Dec 2015 17:00:00 MST
Pragma: No-cache
Server: false
X-Powered-By: Proximo
```

---

The URL <https://carrot:8443/api/rest/users/LOCAL/admin> represents a **plcm-user** resource with a **username** of **admin**, a **first-name** of **Default**, and a **last-name** of **System Administrator User**.

You can perform the same HTTP GET request using the RestClient plugin for Firefox instead of a web browser. The RestClient plugin allows you to specify the HTTP method, PUT/POST body, and header information that you cannot conveniently get from the browser.

You can also perform the same HTTP GET request from a Linux command line instead of a web browser using the curl utility:

```
curl --silent --insecure --user admin:admin
https://carrot:8443/api/rest/users/LOCAL/admin | xmllint --format -
```

The curl utility is a Linux command line program that will perform an HTTP GET.

- The --insecure option to curl tells it to skip certificate validation, similar to what you did in the browser.
- The --user option tells curl the **username** and **password** to use when authenticating with the server.

xmllint is a Linux program that will format XML for easier reading.

## POST Method

Similarly, you can use curl to create a user by sending an HTTP POST request with user data to the server.

The echo command writes the user XML to standard output:

- The --data @- option tells curl to read the data output by the echo command.
- The --header option tells curl what Content-Type HTTP header to use. The Content-Type header is explained in the “Software Versioning Considerations” section.
- The --request POST option tells curl to perform an HTTP POST operation instead of an HTTP GET operation.
- The --include option tells curl to output the HTTP headers returned by the server.

Request Body:

```
echo '<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<plcm-user
  xmlns="urn:com:polycom:api:rest:plcm-user"
  xmlns:ns2="urn:com:polycom:api:rest:plcm-user-role"
  xmlns:ns3="urn:com:polycom:api:rest:plcm-user-role-list">
  <ns3:plcm-user-role-list>
    <ns2:plcm-user-role>
      <ns2:name>Administrator</ns2:name>
    </ns2:plcm-user-role>
    <ns2:plcm-user-role>
      <ns2:name>Auditor</ns2:name>
    </ns2:plcm-user-role>
  </ns3:plcm-user-role-list>
  <username>example</username>
  <first-name>Example</first-name>
  <last-name>User</last-name>
  <domain>LOCAL</domain>
  <password>password</password>
  <disabled>false</disabled>
  <locked>false</locked>
</plcm-user>
' | curl --data @- --user admin:admin --header "Content-Type: application/vnd.plcm.plcm-
user+xml" \--request POST --insecure --include https://carrot:8443/api/rest/users
```

---

## Response Header:

```
Status Code: 201 Created
Content-Length: 0
Date: Fri, 18 May 2012 16:00:59 GMT
Etag: "7f2e3a433f13efc0dc91fcae435d4d03"
Location: https://server:port/api/rest/users/LOCAL/example
Server: false
X-Powered-By: Proximo
```

---

The response header returned by the server indicates that the URL for the newly created user resource is: <https://carrot:8443/api/rest/users/LOCAL/example>

---

### Note

The following example is for the RealPresence DMA system and will not work on the RealPresence® Resource Manager system.

## DELETE Method

The user can also be deleted with an HTTP DELETE request.

URL: <https://<server>:<port>/api/rest/users/LOCAL/user1>

Response Header:

In these examples, we used a web browser and curl as HTTP clients because they are easily available and their operation can be described in a self-contained way.

---

### Note

The system only supports HTTPS connections with BASIC authentication. The client must support this, or it will not work. The system does not support unencrypted HTTP.

The API is not platform specific, and any other HTTP client could have been used instead. Readers are encouraged to use the HTTP client best suited to their development environment. Nevertheless, it is frequently useful to have curl available as it is a very useful troubleshooting tool. Using curl and a text editor, a developer may invoke any API operation and see the result with minimal overhead.

## Resource Descriptions

The API is organized into collections of resources accessed by HTTP methods and is formally described by a set of Web Application Description Language (WADL) and XML Schema (XSD) documents.

The RealPresence Platform API reference documentation provides descriptions of the available resources and methods. On a RealPresence DMA system, you can access the RealPresence Platform API reference documentation by browsing to the following URL:

<https://<ip address or hostname of system>:8443/api/rest/documentation/>

---

**Note**

See <http://www.w3.org/Submission/wadl/> for the WADL specification.

See <http://www.w3.org/XML/Schema> for information about XSD.

The WADL resource documents provide the following information:

- The resources the API exposes (each WADL describes one resource).
- A list of referencing XSDs.
- Paths to available HTTP methods.
- The path to a resource.
- The HTTP methods that are available for each resource, their parameters and possible response codes.

The XSD documents provide the following information:

- The schema of the XML document that defines the attributes of each resource.

The WADL and XSD documents that describe the API are available in three different forms:

- The API development kit contains the complete set of these documents.
- The documents are available from each server at runtime, via special URLs.
- The API reference documentation is composed of the WADL and XSD documents reformatted to make them more readable.

---

**Note**

The following example is for the RealPresence DMA system and will not work on the RealPresence® Resource Manager system.

As an example, we will request the WADL and XSD documents from the server for the user resource using the curl command:

```
curl --silent --insecure --user admin:admin  
https://carrot:8443/api/rest/users?_wadl | xmllint --format -
```

## Response Body:

```
<application xmlns="http://wadl.dev.java.net/2009/02" xmlns:plcm-
error="urn:com:polycom:api:rest:plcm-error" xmlns:plcm-user="urn:com:polycom:api:rest:plcm-
user" xmlns:plcm-user-list="urn:com:polycom:api:rest:plcm-user-list"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://wadl.dev.java.net/2009/02
http://www.w3.org/Submission/wadl/wadl.xsd">
  <doc title="User" />
  <grammars>
    <include href="https://server:port/api/rest/users/xml.xsd" />
    <include href="https://server:port/api/rest/users/atom.xsd" />
    <include href="https://server:port/api/rest/users/plcm-error.xsd" />
    <include href="https://server:port/api/rest/users/plcm-user.xsd" />
    <include href="https://server:port/api/rest/users/plcm-user-list.xsd" />
    <include href="https://server:port/api/rest/users/plcm-user-role.xsd" />
    <include href="https://server:port/api/rest/users/plcm-user-role-list.xsd" />
  </grammars>
  <resources base="https://server:port/api/rest/users/">
    <resource id="com.polycom.api.rest.PlcmUserResource" path="/users">
      <method id="search" name="GET">
.....
      </resource>
    </resources>
  </application>
```

---

Once you have the WADL you can then request the user XSD document from the server with the command:

```
curl --silent --insecure --user admin:admin https://carrot:8443/api/rest/users/plcm-user.xsd | xmllint --
format -
```

The WADL can also be retrieved using RestClient. RestClient allows you to use a GUI to view the XSD instead of having to issue additional commands.

Readers are encouraged to familiarize themselves with the format of the WADL and XSD files and the RealPresence Platform API reference documentation. The most specific and detailed information about the API can be found in the reference documentation.

In theory, both WADL and XSD are specific and formal enough that they can be used by automated tools to generate client code. In practice, there are many good tools for generating XML parsers from XSD files, but there are not yet good tools for generating client code from WADL.

## ***Links and Navigation***

A RESTful API requires the server to provide a mechanism for an API client to navigate from one resource to another through a relationship of links. The XML representation document for a resource will contain a set of one or more links to related resources.

These links are supplied in the entity body in the form of an Atom link. When a resource is created, a location header is supplied that contains the URL to the created resource. The links contain various attributes that can be set with values that are either standard or customized values.

The general format is the following:

```
<link title="<some title>" type="<media type>" rel="<relation type>"
href="<resource URL>"
```

### The *title* attribute

This attribute will contain a short, but descriptive title for this link.

#### **Example**

```
title="Owned Conference Rooms"
```

### The *type* Attribute

The *type* attribute will contain the media type value that corresponds to the target link. The media type will comply with the following format:

- Application  
Indicates that the resource is associated with an application
- vnd.plcm  
The "vnd" section indicates that the media type is "vendor" specific and the "plcm" section identifies this as a Polycom vendor.
- Resource name  
The name of the resource, such as "plcm-user"
- Subtype  
Identifies the resource representation format such as XML, JSON, text, etc.

#### **Example (using XML subtype)**

```
type="application/vnd.plcm.plcm-conference-room+xml"
```

#### **Example (using JSON subtype)**

```
type="application/vnd.plcm.plcm-conference-room+json"
```

### The *rel* Attribute

The *relation* type attribute provides context for the API client that will operate on the link. All API resources use custom relation types instead of the standard *related* value.

For example, a **User** resource can be an owner to one or more conference rooms.

#### **Example**

```
rel="urn:com:polycom:api:rest:link-relations:conference-room-owner"
```

### The *href* Attribute

This attribute will contain the URL where the resource resides.

#### **Example**

```
href="https://<hostname>:8443/api/rest/conference-rooms?username=admin&domain=LOCAL"
```



---

## Note

Information on Atom links can be found here: <http://tools.ietf.org/html/rfc4287>.

## Link Example

For example, the **user** resource contains a self-link and a link to the list of **conference rooms** that belong to the user:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <ns4:plcm-user xmlns="http://www.w3.org/2005/Atom" xmlns:ns2="urn:com:polycom:api:rest:plcm-
user-role" xmlns:ns3="urn:com:polycom:api:rest:plcm-user-role-list"
xmlns:ns4="urn:com:polycom:api:rest:plcm-user">
  <link title="Self Relationship" type="application/vnd.plcm.plcm-user+xml" rel="self"
href="https://server:port/api/rest/users/LOCAL/admin"/>
  <link title="Owned Conference Rooms" type="application/vnd.plcm.plcm-conference-room+xml"
rel="urn:com:polycom:api:rest:link-relations:owned-conference-rooms"
href="https://server:port/api/rest/conference-rooms?username=admin&domain=LOCAL"/>
  <ns3:plcm-user-role-list>
    <ns2:plcm-user-role>
      <ns2:name>Administrator</ns2:name>
    </ns2:plcm-user-role>
    <ns2:plcm-user-role>
      <ns2:name>Auditor</ns2:name>
    </ns2:plcm-user-role>
    <ns2:plcm-user-role>
      <ns2:name>Provisioner</ns2:name>
    </ns2:plcm-user-role>
  </ns3:plcm-user-role-list>
  <ns4:username>admin</ns4:username>
  <ns4:first-name>Default</ns4:first-name>
  <ns4:last-name>System Administrator User</ns4:last-name>
  <ns4:domain>LOCAL</ns4:domain>
  <ns4:disabled>>false</ns4:disabled>
  <ns4:locked>>false</ns4:locked>
  <ns4:entity-tag>260dc1ffa9ac8d1bafa2d2ba41115bd7</ns4:entity-tag>
</ns4:plcm-user>
```

---

The self-link is the following line:

```
<link title="Self Relationship" type="application/vnd.plcm.plcm-user+xml"
rel="self" href="https://<server>:<port>/api/rest/users/LOCAL/admin"/>
```

Another example is the **conference** resource, which will contain links to:

- The **conference room** hosting the conference.
- The **MCU** hosting the conference.
- The **user** that owns the conference.
- The list of **participants** for the conference.

When a resource contains a related link to cross reference another resource, the link may resolve to a different server. This technique is used for a variety of virtualization purposes.

The following are some example scenarios:

- When the scheduler on a RealPresence Resource Manager system starts a scheduled conference on a RealPresence DMA 7000 system, the link to the conference that it returns is a resource on the RealPresence DMA 7000 system, not the RealPresence Resource Manager system.

Subsequent conference control and monitoring the API calls must be directed to the RealPresence DMA 7000 system server.

- When the API is used to start a conference on a superclustered RealPresence DMA 7000 system, the conference is started on the server that owns the conference room, not the server where the API request was made.

The link to the **conference** resource returned points to the server used by the API client. For example, in a dual cluster supercluster (cluster A and cluster B), a conference is in progress on cluster B. If the API client queries cluster A for the conference information, the links will point to cluster A even though it is actually running on cluster B. Subsequent conference control and monitoring can be done from any server in a supercluster.

---

**Note**

Whenever possible, use the links returned by the API rather than building your own.

## ***Custom Link Relationship Types***

The API defines custom relationship types for all atom links to help facilitate programmatic parsing of links.

The following tables are examples of RealPresence DMA system custom relationship types. Not all relationship types are included here.

## RealPresence DMA System Custom Relationship Types:

### Conference Link Relationships

Name	Title	Description
urn:com:polycom:api:rest:link-relations: <b>host-conference-room</b>	Hosting Conference Room	This is the specific <b>Conference Room</b> resource that is hosting this <b>Conference</b> resource.
urn:com:polycom:api:rest:link-relations: <b>conference-room-owner</b>	Conference Room Owner	This <b>Conference</b> resource is hosted by the <b>Conference Room</b> resource stated above. As a result, the <b>Conference Room</b> is owned by the specific <b>User</b> resource.
urn:com:polycom:api:rest:link-relations: <b>conference-participants</b>	Conference Participants	These are all the <b>User</b> resources that are participating in the <b>Conference</b> .

### Conference Room Link Relationships

Name	Title	Description
urn:com:polycom:api:rest:link-relations: <b>conference-room-owner</b>	Conference Room Owner	This <b>Conference Room</b> resource is owned by the specific <b>User</b> resource.
urn:com:polycom:api:rest:link-relations: <b>applied-conference-template</b>	Applied Conference Template	This <b>Conference Room</b> resource will apply this <b>Conference Template</b> resource when it hosts a conference.
urn:com:polycom:api:rest:link-relations: <b>containing-mcu-pool-order</b>	Containing MCU Pool Order	This <b>Conference Room</b> resource will use this <b>MCU Pool Order</b> resource to select MCUs when this room actively hosts a conference.
urn:com:polycom:api:rest:link-relations: <b>start-conference-request</b>	Start Conference Request	This <b>Conference</b> resource can start a conference that will be hosted by this <b>Conference Room</b> resource.

### MCU Link Relationships

Name	Title	Description
urn:com:polycom:api:rest:link-relations: <b>containing-mcu-pool</b>	Containing MCU Pool	An <b>MCU</b> resource is contained in zero or more <b>MCU Pool</b> resources.

### ***MCU Pool Link Relationships***

<b>Name</b>	<b>Title</b>	<b>Description</b>
urn:com:polycom:api:rest:link-relations: <b>containing-mcu-pool-order</b>	Containing MCU Pool Order	This <b>MCU Pool</b> resource is contained in zero or more <b>MCU Pool Order</b> resources.
urn:com:polycom:api:rest:link-relations: <b>contained-mcu</b>	Contained MCU	This is an <b>MCU</b> resource contained in the <b>MCU Pool</b> resource.

### ***MCU Pool Order Link Relationships***

<b>Name</b>	<b>Title</b>	<b>Description</b>
urn:com:polycom:api:rest:link-relations: <b>contained-mcu-pool</b>	Contained MCU Pool	This <b>MCU Pool Order</b> resource contains zero or more <b>MCU Pool</b> resources.

### ***User Link Relationships***

<b>Name</b>	<b>Title</b>	<b>Description</b>
urn:com:polycom:api:rest:link-relations: <b>owned-conference-rooms</b>	Owned Conference Rooms	This <b>User</b> resource owns zero or more <b>Conference Room</b> resources.

## **RealPresence Resource Manager Custom Relationship Types:**

### ***Device Link Relationships***

<b>Name</b>	<b>Title</b>	<b>Description</b>
urn:com:polycom:api:rest:link-relations: <b>device-owner</b>	Device Owner	This <b>Device</b> resource is owned by a <b>User</b> resource.
urn:com:polycom:api:rest:link-relations: <b>belongs-to-area</b>	Belongs to Area	This <b>Device</b> resource belongs to an <b>Area</b> resource.

## User Link Relationships

Name	Title	Description
urn:com:polycom:api:rest:link-relations: <b>owned-devices</b>	Owned Devices	This <b>User</b> resource owns zero or more <b>Endpoint</b> resources.
urn:com:polycom:api:rest:link-relations: <b>member-of-group</b>	Member of Group	This <b>User</b> resource is a member of a <b>Group</b> resource.
urn:com:polycom:api:rest:link-relations: <b>belongs-to-area</b>	Belongs to Area	This <b>User</b> resource belongs to an <b>Area</b> resource.

## Authentication and Authorization

Authentication is checked against the same set of **usernames** and **passwords** that allow GUI login access.

For a local user account, a domain does not need to be specified. If a domain is not specified, the system will assume **Local**. For an enterprise user account, the domain is required and **username** must be in the format of **domain\username**.

Each product in the RealPresence Platform uses a distinctive authentication realm for the API. Although it's not necessary to use this information, knowing this may be helpful to your client systems when interacting with multiple components of the platform. The RealPresence DMA system uses 'DMA 7000', whereas the RealPresence Resource Manager system uses 'XMA API'.

Authorization is performed by checking the roles associated with the user account. The roles required for each HTTP method of a given resource is documented in its WADL.

API access is only allowed for authenticated users who are authorized to use the particular API method.

## Auditing

All API access is audited.

The RealPresence DMA system can be configured to manage logging and history retention such as Call History, Conference History, and Conference Detail Records (CDRs) from the **Reports** menu. All API usage is tracked, and the history will be retained in the system log files for future reference if necessary.

## Audit Logs

### RealPresence DMA System

All auditing events are logged in server.log. The archived logs can be downloaded at **Maintenance > System Log Files**.

On the RealPresence DMA 7000 system, the audit logs can be obtained by downloading the archived logs, unzipping the log archive, and opening the server.log file with a text editor.

The following is an example of an audit entry in the server.log file:

```
2011-11-29 20:13:52,122|SECURITY|http-0.0.0.0-8443-7:|command_audit_auth|  
auth debug UserID=[admin] SessionID=[0ec9c161-a8c2-4c67-bc07-b9217cb3c965]
```

```
ClientIP=[111.222.111.222] UserAction=[getUser] Result=[SUCCESS]
Parameters=[domain=LOCAL,userId=admin]
```

## **RealPresence Resource Manager System**

All auditing events are logged in Jserver.log. The archived logs can be downloaded at **Reports > System Log Files**.

On the RealPresence Resource Manger system, the audit logs can be obtained by downloading the logs from the **System Log Files** and **Audit Log Files** pages, unzipping the archive, and searching the JServer.log file.

The log entries will look like:

- JServer.log:  
17/Nov/2011:17:13:50:796 -0700|INFO |ajp-127.0.0.1-8009-  
1|CMALoginModule| (\*\*\*\*\*)API method GET attempted on /api/rest/groups

## **Error Handling**

The API reports errors using HTTP status codes and sometimes by returning an XML document of type **plcm-error**.

For example: Attempting to get a conference that does not exist returns the following:

- a 404 'Not found' status code.
- a **plcm-error** document.

Response Body:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<plcm-error xmlns="urn:com:polycom:api:rest:plcm-error">
  <status-code>404</status-code>
  <description>Conference nonexistent does not exist.</description>
</plcm-error>
```

---

Response Header:

```
Status Code: 404 Not Found
Cache-Control: no-cache
Content-Length: 216
Content-Type: application/vnd.plcm.plcm-error+xml
Date: Fri, 18 May 2012 17:30:03 GMT
Expires: Wed, 31 Dec 2015 17:00:00 MST
Pragma: No-cache
Server: false
X-Powered-By: Proximo
```

---

The API servers that encounter an error include a **plcm-error** document in the response whenever possible. However, some requests may fail too early in their processing for a **plcm-error** to be generated.

The possible errors that each API method may encounter, and the status codes associated with them, may be found in the RealPresence Platform API reference documentation and the WADL documents.

In addition, the following table describes some common status codes returned by the API.

Status Code	Description
200 OK	The request has succeeded.
201 Created	The request has been fulfilled and a new resource has been created.
204 No Content	The server has fulfilled the request but does not need to return an entity-body.
304 Not Modified	The resource has not been modified on the server.
400 Bad Request	The request document is not well formed or is invalid according to its schema.
401 Unauthorized	No login credentials are supplied, or the supplied credentials are invalid.
403 Forbidden	The user's roles do not grant sufficient privileges to invoke the method. 403 may also be returned when the API license is not enabled.
404 Not Found	The URL does not resolve to a valid resource.
406 Not Acceptable	The response is not acceptable according to the accept headers sent in the request.
409 Conflict	The request is valid, but the server is unable to fulfill it due to a problem with the current state of the resource. The user may be able to resolve the conflict and resubmit the request.
412 Precondition Failed	The request contains an E-Tag header that does not match the current E-Tag of the resource.
415 Unsupported Media Type	The client has sent a POST to the server with a content type that the server cannot consume.
428 Precondition Required	The request does not contain an E-tag in the message body or in the header.
500 Internal Server Error	An error occurred on the server that was not related to any problem with the request. Typically, this status code will only be returned because of a server defect or misconfiguration.

## ***The Passback and Passthru Fields***

Many objects, such as users, conference notifications, virtual meeting rooms (VMRs), and templates, have a passback field. This field can hold any arbitrary data you wish to store in it. When you query the object, the system includes the passback value in the response. You can use this field to associate data that is meaningful to you with a certain object for the life of that object. This may be useful, for example, to associate user-related values in a corporate Active Directory database with RealPresence DMA system VMRs, to more easily determine which VMRs a corporate user owns.

Similar to the passback field, the passthru field can be used to associate arbitrary data with certain objects. The passthru field can be used for the same purpose as the passback field. However, data in the passthru field is also included in its own column in CDR output for that object.

See the RealPresence Platform API reference documentation for specific field details.

## ***XML and JSON Payload Formats***

You can send and receive data using the RealPresence Platform API in XML or JSON format.

To specify XML or JSON for requests, specify the `+xml` or `+json` media type suffix in the 'Accept' and 'Content-Type' headers. For example, to specify XML for the `plcm-conference-room-v2` resource, use the following request format:

```
Accept: application/vnd.plcm.plcm-conference-room-v2+xml
```

To specify JSON for the same request, use the following format:

```
Accept: application/vnd.plcm.plcm-conference-room-v2+json
```

If you do not specify an 'Accept' header in your requests, XML is used as the default payload type.



The following is an example of the XML representation of a user object:

```
<ns4:plcm-user xmlns="http://www.w3.org/2005/Atom"
xmlns:ns2="urn:com:polycom:api:rest:plcm-user-role"
xmlns:ns3="urn:com:polycom:api:rest:plcm-user-role-list"
xmlns:ns4="urn:com:polycom:api:rest:plcm-user"><link
href="https://10.47.19.213:8443/api/rest/users/LOCAL/admin" rel="self"
type="application/vnd.plcm.plcm-user" title="Self Relationship" /><link
href="https://10.47.19.213:8443/api/rest/conference-
rooms?domain=LOCAL&username=admin" rel="urn:com:polycom:api:rest:link-
relations:owned-conference-rooms" type="application/vnd.plcm.plcm-conference-
room" title="Owned Conference Rooms" /><ns3:plcm-user-role-list><ns2:plcm-
user-role><ns2:name>Administrator</ns2:name></ns2:plcm-user-role><ns2:plcm-
user-role><ns2:name>Auditor</ns2:name></ns2:plcm-user-role><ns2:plcm-user-
role><ns2:name>Provisioner</ns2:name></ns2:plcm-user-role></ns3:plcm-user-
role-list><ns4:username>admin</ns4:username><ns4:first-name>System</ns4:first-
name><ns4:last-name>Admin</ns4:last-
name><ns4:domain>LOCAL</ns4:domain><ns4:disabled>>false</ns4:disabled><ns4:lock-
ed>>false</ns4:locked><ns4:entity-
tag>341a633e18848ee915780495f002f32c</ns4:entity-tag></ns4:plcm-user>
```

---

The following is the same user object using JSON representation:

```
{
  "atomLinkList": [
    {
      "href": "https://10.47.19.213:8443/api/rest/users/LOCAL/admin",
      "rel": "self",
      "type": "application/vnd.plcm.plcm-user",
      "title": "Self Relationship"
    },
    {
      "href": "https://10.47.19.213:8443/api/rest/conference-rooms?domain=LOCAL&username=admin",
      "rel": "urn:com:polycom:api:rest:link-relations:owned-conference-rooms",
      "type": "application/vnd.plcm.plcm-conference-room",
      "title": "Owned Conference Rooms"
    }
  ],
  "plcmUserRoleList": {
    "plcmUserRoleList": [
      {
        "name": "Administrator"
      },
      {
        "name": "Auditor"
      },
      {
        "name": "Provisioner"
      }
    ]
  },
  "username": "admin",
  "firstName": "System",
  "lastName": "Admin",
  "domain": "LOCAL",
  "disabled": false,
  "locked": false,
  "entityTag": "341a633e18848ee915780495f002f32c"
}
```

---

XML and JSON use different naming conventions for resource fields. When using XML, resource fields use the “dash” naming convention. When using JSON, they use the camel case naming convention. An example of a field name in XML format is the *max-bandwidth* field; this field is named *maxBandwidth* when using JSON. Refer to the RealPresence Platform API reference documentation for field names in XML and JSON formats.

## Software Versioning

The API has a robust set of features for handling issues of software versioning.

The API is forward compatible for existing methods and schemas. Clients using a new version of the API can connect to servers using an old version of the API if they don't require the new functionality provided by the new version of the API.

The API is backward compatible. Clients using an old version of the API can connect to servers using a new version of the API.

The API is intended to support the following use case:

A customer with multiple servers has written a client application to interact with these servers using the API.

- The client should be able to upgrade one or all the servers without needing to modify their client application.
- The client should be able to upgrade some of their servers and modify their client application to take advantage of the new features in those upgraded servers, without upgrading all their servers.

There are two kinds of API versioning:

- Transparent versioning maintains forward and backward compatibility.
- Discrete versioning does not maintain forward or backward compatibility.

### Transparent Versioning

Typically, transparent versioning is used when a new optional attribute is added to a resource in a new version of the API.

In this case any of the following scenarios may occur:

- When a client using an old version of the API sends a request to a server using the new version of the API, the client does not include the new attribute in the request. The attribute is optional, and the server can cope with its absence.
- When a client using an old version of the API receives a response from a server using the new version of the API, the response contains the new attribute. The client copes with the attribute that it does not understand by ignoring it.
- When a client using the new version of the API sends a request to a server using an old version of the API, the client includes the new attribute in the request. The server copes with the attribute that it does not understand by ignoring it.
- When a client using the new version of the API receives a response from a server using an old version of the API, the response does not contain the new attribute. The attribute is optional, and the client can cope with its absence.

### Discrete Versioning

Discrete Versioning is used when new attributes are added that cannot simply be ignored as optional, or when a resource has structural changes that cannot simply be represented as new attributes.

The API uses the **Content-Type** and **Accept** HTTP headers to cope with incompatible differences. Each XML schema (including all versions of the schema with only compatible differences) is designated by a content type.

For example, the content type for the `plcm-user` resource when using XML is `application/vnd.plcm.plcm-user+xml`.

The content type is specified in the XSD document specifying the schema, and every WADL file indicates the content types supported for each method of the resource. In the case that a new version of the API must introduce an incompatible difference, it will be introduced as a new schema with a new content type.

The API client may use HTTP headers to specify the content type that it sends in a request, and/or the content type that it accepts in a response. When the HTTP headers sent by the client include a **Content-Type** header or an **Accept** header, the server will respect the header, and use the schema version specified by the header.

When your API client sends an XML or JSON document conforming to a certain version of the API, use the **Content-Type** header to specify the version of data you send. When your API client expects an XML or JSON document in response, use the **Accept** header to specify the version of data the request accepts.

The following is an example of an HTTP GET request that accepts only the original version of **plcm-user**, and continues to use it for new versions of the API, even after incompatible differences have been introduced:

```
curl --silent --insecure --user admin:admin --header 'Accept:
application/vnd.plcm.plcm-user+xml'
https://carrot:8443/api/rest/users/LOCAL/admin | xmllint --format -
```

The following is an example of an HTTP GET request that accepts only the 'v2' version of **plcm-user**:

```
curl --silent --insecure --user admin:admin --header 'Accept:
application/vnd.plcm.plcm-user-v2+xml'
https://carrot:8443/api/rest/users/LOCAL/admin | xmllint --format -
```

Polycom is committed to maintaining both forward and backward compatibility of the API. Where possible, the API will support both the old and new content types when incompatible differences occur.

API clients written to use an old version of the API may continue to do so, by specifying the old version in the **Accept** and **Content-Type** headers.

API clients written to use the new version of the API may do so by specifying the new version in the **Accept** and **Content-Type** headers.

In some cases, it may simply be impossible to support the old content type. In these cases, the release notes for the new version of the API will include a list of the content types that are no longer supported.

## Best Practices to Prevent Versioning Issues

Developers using the API should use the following practices, to minimize software versioning related issues and protect against unintentional agent upgrades:

- If you use XML format for API interaction, use an XML parser that respects extension points in the schema, and ignores attributes that it does not understand at these points.
- Include the '**Content-Type**' HTTP header in every request that sends an XML or JSON document.
- Include the '**Accept**' HTTP header in every request that expects an XML or JSON document in response, specifying the version of the resource you are addressing.
- Failure to include an '**Accept**' header may result in an incompatible version being sent in response.

To determine whether a version is compatible, refer to the reference documentation for the API methods, which specifies what representations are possible as returned values.

- As part of the upgrade process for Polycom products, check the release notes for content types that are no longer supported. If there are any, make sure that they are not used in the client application.

## ***Subscriptions and Notifications***

The subscription or notification mechanism allows API clients to register for and receive asynchronous HTTPS notifications about state changes.

---

### **Note**

The subscriptions and notifications mechanisms are not available for the RealPresence Resource Manager system.

When discussing this mechanism:

- The **consumer** is the HTTPS server that the API users set up to receive notifications.
- The **producer** is the Polycom product that publishes the notifications.

### **To receive notifications**

- 1 Prepare a HTTPS server for the producer to publish to.

The producer will not publish notifications to plaintext HTTP connections.

Your web server must implement a specific interface, which is defined in the following WADLs that Polycom provides:

- **plcm-conference-observer.wadl** for consuming conference notifications.
  - **plcm-participant-observer.wadl** for consuming participant notifications.
- 2 POST a subscription to the URL of the resource to be observed.

The producer will post a notification to this web server every time the resource changes.

Notification responses will be treated as follows:

    - If there is no response, the producer will wait 10 seconds before discarding the notification and moving on.
    - If there is a response, the server will cache up to 500 response notifications on a first-in, first-out basis.

The sample code files that are provided with this API contain examples of conference and participant observers.

### **Authentication**

The **username** and **password** in this section refer to the credentials you use to send notifications to the client. They are not the same credentials used for inbound API requests.

When creating a subscription, the user may provide a **username** and **password** in the request body. If provided, the producer will send these credentials over HTTPS BASIC authentication whenever it posts to the consumer. In addition, the producer is configured to offer its X.509 certificate when establishing an HTTPS connection. This allows MTLS, if desired by the consumer.

## Initiating a subscription

You can use the method **plcm-subscription** to subscribe to conferences, participants, or participant lists.

The API Client subscribes to an entity by POSTing a **plcm-subscription** to the resource that it wishes to observe. The producer then returns a link to a resource that can be used for managing the subscription.

You can use the **plcm-conference-list-subscription** method in a similar way to subscribe to conference lists.

## Subscription management

The API Client can cancel the subscription by using a DELETE request on the subscription resource.

The API Client can determine the status of the subscription by using a GET request on the subscription resource.

## Error reporting

If the producer is unable to publish a notification to the consumer, it will record the error in the **plcm-error** attribute of the **subscription**. This enables the consumer to troubleshoot the issue by reading the **subscription** resource.

## Notification format

Each type of observable resource has a different notification format, and a different WADL that the consumer must implement. This enables the consumer to use standard tools to implement a web server with strong typing.

Typically, each notification contains a **plcm-notification** structure, which describes the notification event, and a content structure, containing a resource or a list of resources.

---

### Note

Subscribing to asynchronous notifications does not guarantee a notification event order.

## Subscription to lists

When subscribing to a list, the list notification will not send full lists. The first notification will be several ADD notifications. For example, for a conference list, one ADD notification will be sent for each of the currently ongoing conferences. Subsequent notifications will be ADD, DELETE, and UPDATE notifications.

Each notification will have a sequence number beginning at 1 and incrementing by 1 for each notification. You can use this sequencing to determine if a notification has been missed.

The consumer can request an update of the subscription. In this case, the sequence number will reset to 1 and they will receive an ADD notification. In the case of conference lists, the consumer will receive one ADD notification for each ongoing conference.

---

**Note**

The producer may send an UPDATE notification for a list at any time. Consumers should be prepared to receive an unsolicited UPDATE notification for a list resource. The client can choose not to receive UPDATE notifications via the **plcm-subscription** representation.

## High Availability

The subscription or notification mechanism is intended to support a High Availability superclustered environment. Each subscription will be the responsibility of the cluster that received the subscription request. Clusters in a supercluster only know about subscriptions that they are managing and not those of other clusters.

Your subscription is serviced by the cluster you posted it to. If that cluster becomes unavailable, the subscription is lost.

## Subscription renewal

When POSTing a subscription, the consumer may specify a time at which the subscription expires. This enables the producer to automatically clean up subscriptions to consumers that are no longer active. In this case, it is the responsibility of the consumer to periodically extend the subscription if it is still active.

---

**Note**

The samples package contains two sample notification consumers: 'plcm-conference-observer' and 'plcm-participant-observer'.

## Learning from Sample Code

Another key learning tool is the set of sample code files that are provided with this API. Sample code and scripts that call the APIs are downloadable from any DMA and include samples in Python, Java, and Perl.



# API Services

The RealPresence Platform API provides programmatic access to the Polycom RealPresence Resource Manager and RealPresence DMA 7000 systems. This enables Polycom customers to customize their conferencing experience and integrate their Polycom infrastructure products with their existing IT infrastructure.

The API includes support for configuration, MCU management, provisioning, billing, resource reporting, conference control and monitoring.

## Alerting Capabilities

The alerting capabilities of the API allow application developers to manage the alerts and alerting settings of a system.

The alerting capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-alert	Allows you to view information on all currently active system alerts, subscribe to certain alerts, and get information about specific alerts.	No	Yes
plcm-alert-observer	Allows you to receive a notification about a new alert or a change to an existing alert.	No	Yes
plcm-alerting-config	Allows you to view or change the alerting configuration, as well as restore the alerting configuration to default values.	No	Yes
plcm-remote-alert-profile	Allows you to view the remote alert profiles configured on the system as well as retrieve information for a specific remote alert profile.	Yes	No

## Site Topology Capabilities

Application developers can utilize the site topology capabilities of the API to build a network model that mirrors the physical conferencing environment. The site topology capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-network-cloud	Allows you to list, create, update, and delete network clouds, as well as view information for a specific network cloud.	No	Yes
plcm-site	Allows you to list, create, update, and delete sites, as well as view information for a specific site.	No	Yes
plcm-site-link	Allows you to list, create, update, and delete site links, as well as view information for a specific site link.	No	Yes
plcm-site-to-site-exclusion	Allows you to list, create, and delete site-to-site exclusions, as well as view information for a specific site-to-site exclusion.	No	Yes
plcm-site-topology	Allows you to list, retrieve, create, change, and delete sites, site links, site-to-site exclusions, and network clouds.	No	Yes
plcm-territory	Allows you to create, update, delete, or get information about territories on the system. A territory contains one or more sites for which a RealPresence DMA system cluster is responsible.	No	Yes

## Provisioning Capabilities

The provisioning capabilities of the API allow application developers to manage users, their roles, and the resources assigned to each user. The provisioning API includes:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-user	Allows you to create, edit, delete, or get information on users of conferencing services and administrators of Polycom devices.	Yes	Yes
plcm-user-role	Allows you to view the current user roles on the system as well as retrieve information about a specific user role.	Yes	No
plcm-conference-room	Allows you to create, edit, delete, or get information on virtual meeting rooms for a user.	No	Yes
plcm-conference-template	Allows you to create, edit, delete, or get information on conference templates. Conference templates are standalone conference configurations that can be shared between multiple conference rooms.	No	Yes

## Billing Capabilities

The billing capabilities of the API allow application developers to obtain historical call and conference data, allowing billing based on usage. The billing capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-billing	Allows you to retrieve call and conference CDR information for billing purposes.	Yes	Yes

# Conference Control and Monitoring (CCM) Capabilities

The conference control and monitoring (CCM) capabilities of the API allow application developers to observe and manipulate ongoing conferences.

**Note: Unsupported API Operations with Cisco Codian MCUs**

The following API operations are not supported when using a Cisco Codian MCU:

- Set Conference Display Text:  
<https://<hostname>:8443/api/rest/conferences/{conference-identifier}/set-display-text/{display-text}>
  
- Start/Stop Conference Recording:  
<https://<hostname>:8443/api/rest/conferences/{conference-identifier}/start-recording>  
<https://<hostname>:8443/api/rest/conferences/{conference-identifier}/stop-recording>

The conference control and monitoring capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-conference	Allows you to create, edit, delete, or get information on ongoing conferences.	Yes	Yes
plcm-conference-observer	Allows you to receive notifications about changes to conferences or conference lists.	No	Yes
plcm-participant <sup>1</sup>	Allows you to manage and get information on conference participants.	No	Yes
plcm-participant-observer	Allows you to get notifications about changes to a participant or list of participants.	No	Yes

<sup>1</sup>There is no plcm-participant WADL file. Participant related methods are described in the plcm-conference.wadl because participants cannot exist without a conference.

## Conference Scheduling Capabilities

The conference scheduling capabilities of the API allow application developers to schedule future conferences on the RealPresence Resource Manager system. The conference scheduling capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-dial-string-reservation	Allows you to view, change, and delete dial string reservations.	Yes	No
plcm-reservation	Allows you to create, update, read, delete, and search for future conference reservations on the RealPresence DMA system.	Yes	No
plcm-participant-availability	Allows you to get information about a participant's conference schedule within a start-time and end-time.	Yes	No

## Resource Reporting Capabilities

The resource reporting capabilities of the API allow application developers to determine what conferencing resources are available for scheduling. The resource reporting capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-mcu	Allows you to create, update, delete, or get information about MCUs available to the RealPresence DMA System.	No	Yes
plcm-mcu-pool	Allows you to create, update, delete, or get information about the pools of deployed MCUs that are available to host conferences.	No	Yes
plcm-mcu-pool-order	Allows you to create, update, delete, or get information about prioritized lists of MCU pools that are available to host conferences.	No	Yes

## Area Capabilities

The area capabilities of the API allow application developers to read the list of area (or tenant) objects from the system.

The area capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-area	Allows you to get information about the system's area (or tenant) objects.	Yes	No

## Group Capabilities

The group capabilities of the API allow application developers to read the list of local groups from the system. The group capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-group	Allows you to read the list of local groups from the system.	Yes	No

## Device Capabilities

The device capabilities of the API provide an interface to retrieve either a single device or all devices.

The device capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-device	Allows you to retrieve or update a device's information.	Yes	No
plcm-ext-sip-peer	Allows you to add, configure, and remove external SIP peers.	No	Yes

## System Configuration and Status Capabilities

The system configuration and status capabilities of the API provide an interface to view and manage certain system properties and retrieve system status.

The system configuration and status capabilities include:

Resource Collection	Description	RealPresence Resource Manager system	RealPresence DMA system
plcm-conference-settings	Allows you to view and change the system's conference settings properties (located on the <b>Admin &gt; Conference Manager &gt; Conference Settings</b> page in the RealPresence DMA system administrative GUI).	No	Yes
plcm-config	Allows you to view and change the call server, history retention, and logging configuration.	No	Yes
plcm-status	Allows you to perform a "shallow ping" of the system. This indicates if the system is responsive, and if the system's API infrastructure is parsing commands.	No	Yes
plcm-subscription	Allows you to subscribe to events happening on the Polycom device.	No	Yes
plcm-signaling-config	Allows you to view and change the SIP signaling and H.323 gatekeeper configuration.	No	Yes
plcm-snmp-config	Allows you to view and change the SNMP configuration of the system.	Yes	Yes
plcm-dma-supercluster	Allows you to view the current supercluster configuration of a cluster, as well as join and remove clusters to and from a supercluster.	No	Yes
plcm-presence-config	Allows you to configure Microsoft Lync presence publishing.		
plcm-log-archive	Allows you to initiate the process to roll logs and retrieve information about log archives.	Yes	No
plcm-log-config	Allows you to view and change the logging configuration on the system.	Yes	No

<b>Resource Collection</b>	<b>Description</b>	<b>RealPresence Resource Manager system</b>	<b>RealPresence DMA system</b>
plcm-time	Allows you to retrieve and configure system time and time zone information.	Yes	Yes