

MTAS CCMP Interface

INTERWORK DESCR

© Ericsson AB 2011-20123

Disclaimer Statements

No part of this document may be reproduced in any form without the written permission of the copyright owner.

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing.

Ericsson shall have no liability for any error or damages of any kind resulting from the use of this document.

Trademark List

All trademarks are the property of their respective owners.

Contents

1	Document History	4
2	Scope and Purpose	4
2.1	Interface Entities	4
2.2	Interface Role	5
2.3	Services	5
2.4	Encapsulation and Addressing	6
2.4.1	CCMP Interface	6
2.4.2	Addressing	6
3	Procedures	7
3.1	Overview	7
3.2	Lower Level Procedures	7
3.3	Create Scheduled Conference	7
3.4	Retrieve Scheduled Conference	8
3.5	Delete Scheduled Conference	8
3.6	Update Scheduled Conference	9
3.7	Create Scheduled Conference User	9
3.8	Delete Scheduled Conference User	10
3.9	Update Scheduled Conference User	10
4	Information Model	11
4.1	General	11
4.2	HTTP	11
4.2.1	HTTP request	11
4.2.2	HTTP Response	11
4.3	Create Scheduled Conference	12
4.4	Retrieve Scheduled Conference	14

4.5	Delete Scheduled Conference	15
4.6	Update Scheduled Conference	17
4.7	Create Scheduled Conference User	19
4.8	Delete Scheduled Conference User	21
4.9	Update Scheduled Conference User	23
4.10	CCMP reason phrases for 409 Conflict Response	25
5	Formal Syntax or Schema	28
5.1	Requests and Responses	28
5.2	Conference Information Message format	28
5.3	Create conference information	28
5.4	Retrieve conference information	31
5.5	Update conference information	34
5.6	Create user information	34
5.7	Delete user information	36
5.8	Update user information	36
6	Related Standards	37
7	Terminology	37
7.1	Abbreviations	37
7.2	Definitions	37
8	References	37
9	Annex: POST message, CCMP request/response examples	38
9.1	CCMP confRequest/confResponse, operation create	38
9.2	CCMP confResponse, operation create, error 421	40
9.3	CCMP confResponse, operation create, error 409	41
9.4	CCMP confRequest/confResponse, operation retrieve	42
9.5	CCMP confRequest/confResponse, operation delete	46
9.6	CCMP confRequest/confResponse, operation update	47
9.7	CCMP userRequest/userResponse, operation create	49
9.8	CCMP userRequest/userResponse, operation delete	50
9.9	CCMP userRequest/userResponse, operation update	52

1 Document History

Rev	Date	Sign	Comment
A	2013-04-17	egergft	Clone from 5/155 19-CRA 119 2108 rev B Added: <ul style="list-style-type: none"> - Additional Reason Phrases included in table 18 - IPv6 support
B	2013-11-25	eratlim	The check of the Multiparty Conference license has been removed during provisioning.
C	2014-05-19	eratlim	Updated: <ul style="list-style-type: none"> - Added in chapter 3.1 and 4.1 operation "updated". - In table 6 are the CCMP codes 400 and 404 removed - In table 9 is the CCMP code 511 added - In table 12 are the CCMP codes 400 added and 404 removed - In table 15 are the CCMP codes 400 added and 404 removed - In table 18 is the CCMP code 511 removed - In table 21 is the CCMP code 511 removed - In table 24 is the CCMP code 511 removed

2 Scope and Purpose

2.1 Interface Entities

Within the IMS architecture the interface that is offered by MTAS to handle scheduled conferences is the Centralized Conferencing Manipulation Protocol (CCMP).

The interface can be used by a Conference Owner (CO) via a Conference Administration Server to handle scheduled conferences.
A pre-requisite for the service is that the CO

- must be an IMS user
- must be provisioned with the scheduled conference service
- have the service set to active

This document describes how MTAS deploys this interface.

Any routers or proxies that may be installed between the Conference Owner/Conference Administration Server and MTAS are excluded from the scope of this document.



Figure 1 Conference Owner / Conference Administration Server and MTAS

2.2 Interface Role

This IWD describes the services MTAS supports for scheduled conference service.

2.3 Services

Table 1: Offered Services

Offered Service	Description
Create Scheduled Conference	Creates a scheduled conference for the Conference Owner
Retrieve Scheduled Conference	Retrieves a scheduled conference for the Conference Owner

Delete Scheduled Conference	Deletes a scheduled conference for the Conference Owner
Update Scheduled Conference	Updates a scheduled conference for the Conference Owner
Create Scheduled Conference User	Creates a scheduled conference user for the Conference Owner
Delete Scheduled Conference User	Deletes a scheduled conference user for the Conference Owner
Update Scheduled Conference User	Updates a scheduled conference user for the Conference Owner

Table 2: Used Services

Used Service	Description
--------------	-------------

–

2.4 Encapsulation and Addressing

2.4.1 CCMP Interface

The protocol used by MTAS is CCMP, as defined in RFC6503 ref. [1]. It is used to administrate (create, update, retrieve and delete) the scheduled conference per user. To define a scheduled conference, the conference information and the extension conference information data model are used, as defined in RFC4575 ref. [2] and RFC6501 ref. [3] respectively.

2.4.2 Addressing

The format of the messages used is defined by the combination of the CCMP and the underlying HTTP protocol.

The CCMP root for the MTAS CCMP is as follows:

```
POST /mtasccmp/<service-number>
```

Where:

- <service-number> is the TEL-URI of the service number of the scheduled conference service

and the header

Host: <hostname>:8096

Where:

- < hostname > is the IP address. If <hostname> is a numeric IPv6 address, the address must be enclosed in brackets (for example: [2000::4:66])

For more information see Annex 9.

3 Procedures

3.1 Overview

The CCMP request is sent in HTTP ref. [4] method POST. The CCMP conference request includes information of the operation (create, retrieve, update or delete) in the body of the POST message. A CCMP conference response that may include information of the operation is responded in the body of the HTTP message.

3.2 Lower Level Procedures

N/A

3.3 Create Scheduled Conference

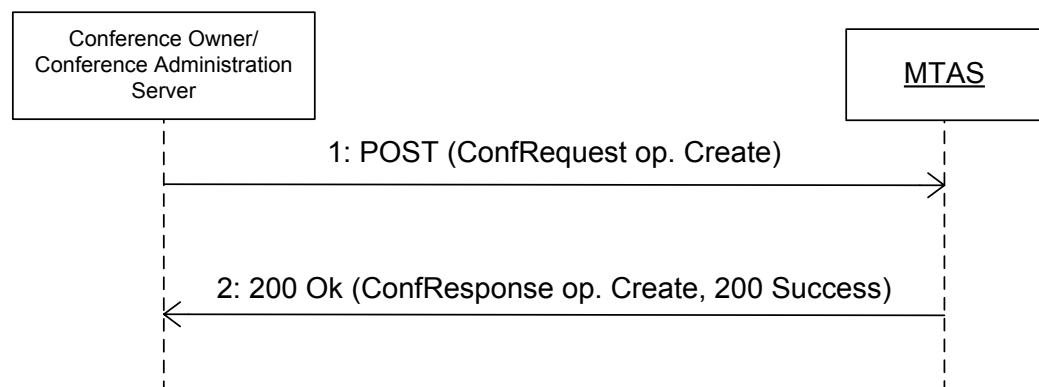


Figure 2 Create Scheduled Conference

1. MTAS receives a POST message with a body including a CCMP confRequest with operation set to create and conference information.

2. MTAS sends 200 OK including a CCMP confResponse with operation set to create, response code set to 200 and response string set to Success.

3.4 Retrieve Scheduled Conference

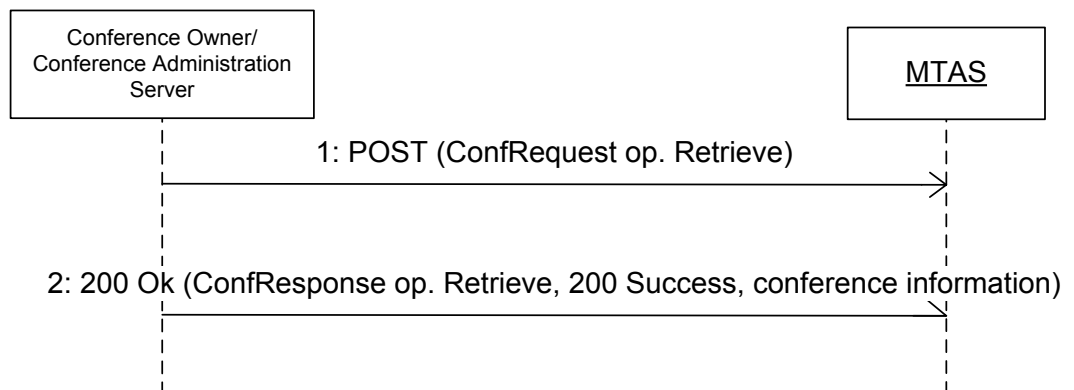


Figure 3 Retrieve Scheduled Conference

1. MTAS receives a POST message with a body including a CCMP confRequest with operation set to retrieve.
2. MTAS sends 200 OK including a CCMP confResponse with operation set to retrieve, response code set to 200, response string set to Success and conference information.

3.5 Delete Scheduled Conference

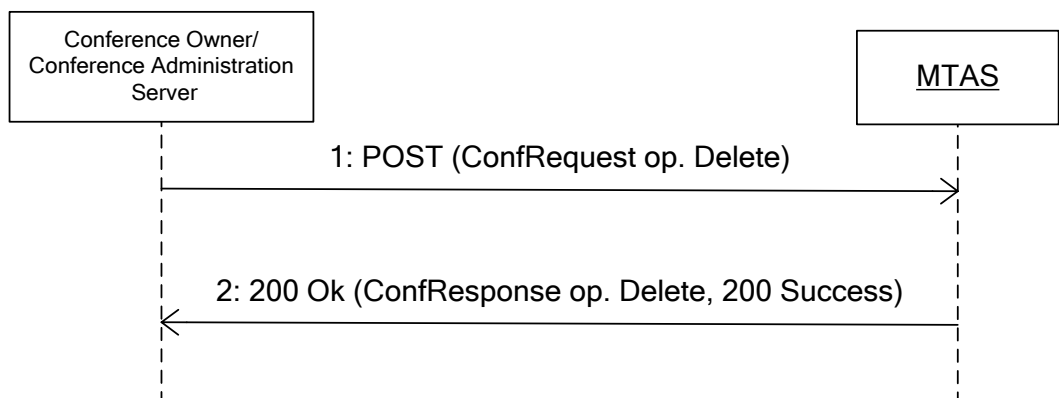


Figure 4 Delete Scheduled Conference

1. MTAS receives a POST message with a body including a CCMP confRequest with operation set to delete.
2. MTAS sends 200 OK including a CCMP confResponse with operation set to delete, response code set to 200, response string set to Success.

3.6 Update Scheduled Conference

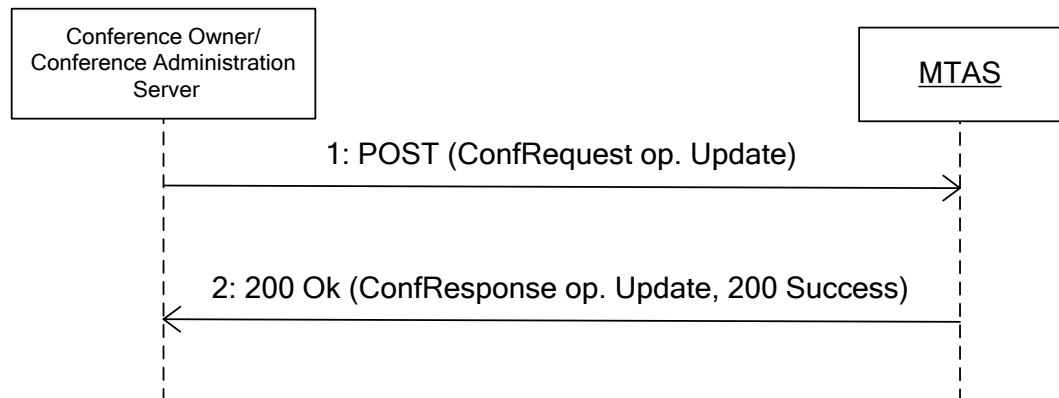


Figure 5 Update Scheduled Conference

1. MTAS receives a POST message with a body including a CCMP confRequest with operation set to update.
2. MTAS sends 200 OK including a CCMP confResponse with operation set to update, response code set to 200, response string set to Success.

3.7 Create Scheduled Conference User



Figure 6 Create Scheduled Conference User

1. MTAS receives a POST message with a body including a CCMP userRequest with operation set to create and conference information.
2. MTAS sends 200 OK including a CCMP userResponse with operation set to create, response code set to 200 and response string set to Success.

3.8 Delete Scheduled Conference User

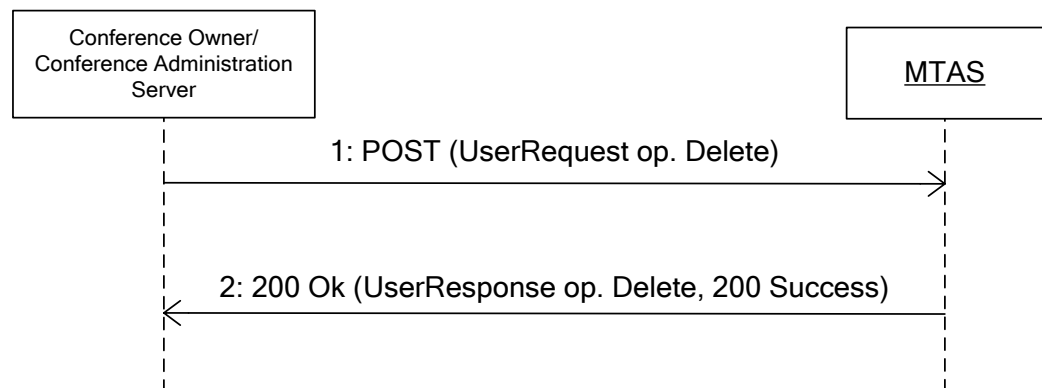


Figure 7 Delete Scheduled Conference

1. MTAS receives a POST message with a body including a CCMP userRequest with operation set to delete.
2. MTAS sends 200 OK including a CCMP userResponse with operation set to delete, response code set to 200, response string set to Success.

3.9 Update Scheduled Conference User



Figure 8 Update Scheduled Conference

1. MTAS receives a POST message with a body including a CCMP userRequest with operation set to update.
2. MTAS sends 200 OK including a CCMP userResponse with operation set to update, response code set to 200, response string set to Success.

4 Information Model

4.1 General

This section describes the parameters that are used by MTAS.

The CCMP Request and response are sent in a HTTP message.

XML examples of successful creating, retrieving, update and deleting and error messages when handling scheduled conferences are found in the Annex see section 9.

4.2 HTTP

4.2.1 HTTP request

CCMP uses the HTTP method POST for request.

Mandatory headers are Content-Type and Accept.

4.2.2 HTTP Response

MTAS uses the following HTTP response codes.

Table 3: HTTP Responses for Scheduled Conference

HTTP code and reason phrase	Reason
200 OK	It includes the CCMP response with response code and response string, see Table 6 (create), Table 9 (retrieve) and Table 12 (delete).
400 Bad Request	A POST method body includes non UTF-8 encoded content or not well-formed XML document.
404 Not Found	The URL is not found on the server.
405 Method Not allowed	For CCMP the REQUIRED method is POST.
406 Not Acceptable	The server is refusing to service the request because the entity of the request is in a format not supported by the requested resource for the requested POST method. The Content-Type and Accept header fields must have the value "application/ccmp+xml".
412 Precondition Failed	The precondition specified by the If-

	Match or If-None-Match header is not supported.
500 Internal Server Error	The server cannot complete the required service due to a system internal error.
503 Service Unavailable	<p>The server is currently unable to handle the request due to a temporary overload or maintenance of the server.</p> <p>The following circumstances are defined when this message is sent</p> <ul style="list-style-type: none"> • If the node refuses to accept requests because the administrative state of the conference service is set to "Locked". • If the node considers itself to be in overload. In this case the retry-after header will be populated.

4.3 Create Scheduled Conference

MTAS expects the following parameters in a CCMP conference request (confRequest) operation create.

Table 4: CCMP Request parameters for Create Scheduled Conference, in middle column the values mean mandatory (M) and optional (O)

Parameter		Comment
ccmpRequest	M	To handle a conference this must be set to "ccmp-conf-request-message-type".
confUserID	M	It contains the XCON-USERID of the user. The XCON-USERID is used to identify the IMS subscriber owning the conference.
operation	M	The request message type is specified to create.
confObjID	O	Not used. The clone mechanism is not supported.
confRequest	M	Specialized operation for scheduled conference. In create the conference information is expected to be included, see Table 26.

MTAS responds in a CCMP conference response (confResponse) operation create.

Table 5: CCMP Response parameters for Create Scheduled Conference, in middle column the values mean mandatory (M) and ignored (I)

Parameter		Comment
ccmpResponse	M	When handling a conference this is set to "ccmp-conf-request-message-type".
confUserID	M	The XCON-USERID received in the request is used.
operation	M	The request operation is specified and is set to create.
confObjID	M	The unique XCON-URI identity of the created conference.
version	M	Set to "1", since at creation the conference object is at its first version.
confResponse	I	Set to empty, because the conference is accepted as requested.

In the HTTP 200 OK the following CCMP responses can be included.

Table 6: CCMP Responses for Create Scheduled Conference

CCMP code and reason phrase	Reason
200 Success	The request was successfully processed.
403 Forbidden	Operation not allowed. The following circumstances are defined when this message is sent <ul style="list-style-type: none"> If the scheduled conference service is inactive for the user.
409 Conflict	There is a set of MTAS application constraints each of which is identified by a phrase. The possible phrase values are given in Table 25. The phrases start with a number and a colon, which can be used by a client to simplify the implementation of application constraints
421 Invalid confUserID	User missing at the server (this code is returned in the case of requests in which the "confUserID" of the sender is invalid).
500 Internal Server Error	The server cannot complete the required service due to a system internal error. This includes the cases where the Sh interface to the HSS does not respond successfully.
501 Not Implemented	Request type or operation envisaged in the protocol, but not implemented in the contacted server.

	Note: The confRequest request with operations create, retrieve, update and delete, and the userRequest request with operations create, update and delete are implemented.
511 Resources Not Available	This code is used when the CCMP server cannot execute a command because of resource issues e.g. it is not possible to create a new conference due to HSS storage limitation.

4.4 Retrieve Scheduled Conference

MTAS expects the following parameters in a CCMP conference request (confRequest) operation retrieve.

Table 7: CCMP Request parameters for Retrieve Scheduled Conference, in middle column the values mean mandatory (M) and optional (O)

Parameter		Comment
ccmpRequest	M	To handle a conference this must be set to "ccmp-conf-request-message-type".
confUserID	M	It contains the XCON-USERID of the user. The XCON-USERID is used to identify any conferencing user within the context of the conferencing system.
operation	M	The request operation is specified to retrieve.
confObjID	M	The unique XCON-URI identity of the conference.
confRequest	O	Not used.

MTAS responds in a CCMP conference response (confResponse) operation retrieve.

Table 8: CCMP Response parameters for Retrieve Scheduled Conference, in middle column the value means mandatory (M) and ignored (I)

Parameter		Comment
ccmpResponse	M	When handling a conference this is set to "ccmp-conf-request-message-type".
confUserID	M	The XCON-USERID received in the request is used.
operation	M	The request operation is specified and is set to retrieve.
confObjID	M	The XCON-URI received in the request is used.
version	I	Not sent.
confResponse	M	Specialized response message for

		scheduled conference, see Table 27.
--	--	-------------------------------------

In the HTTP response 200 OK the following CCMP responses can be included.

Table 9: CCMP Responses for Retrieve Scheduled Conference

CCMP code and reason phrase	Reason
200 Success	The request was successfully processed.
403 Forbidden	Operation not allowed. The following circumstances are defined when this message is sent <ul style="list-style-type: none"> If the scheduled conference service is inactive for the user.
404 Object Not Found	Target conference object is missing at the server. The "confObjID" parameter is not associated with a conference that the system is holding.
421 Invalid confUserID	User missing at the server (this code is returned in the case of requests in which the "confUserID" of the sender is invalid).
500 Internal Server Error	The server cannot complete the required service due to a system internal error. This includes the cases where the Sh interface to the HSS does not respond successfully.
501 Not Implemented	Request type or operation envisaged in the protocol, but not implemented in the contacted server. Note: The confRequest request with operations create, retrieve, update and delete, and the userRequest request with operations create, update and delete are implemented.
511 Resources Not Available	This code is used when the CCMP server cannot execute a command because of resource issues e.g. it is not possible to create a new conference due to HSS storage limitation.

4.5 Delete Scheduled Conference

MTAS expects the following parameters in a CCMP conference request (confRequest) operation delete.

Table 10: CCMP Request parameters for Delete Scheduled Conference, in middle column the values mean mandatory (M) and optional (O)

Parameter		Comment
ccmpRequest	M	To handle a conference this must be set to "ccmp-conf-request-message-type".
confUserID	M	It contains the XCON-USERID of the user. The XCON-USERID is used to identify any conferencing user within the context of the conferencing system.
operation	M	The request operation is specified to delete.
confObjID	M	The unique XCON-URI identity of the conference.
version	O	Not used, because the identified conference is deleted.
confRequest	O	Not used.

MTAS responds in a CCMP conference response (confResponse) operation delete.

Table 11: CCMP Response parameters for Delete Scheduled Conference, in middle column the values mean mandatory (M) and ignored (I)

Parameter		Comment
ccmpResponse	M	When handling a conference this is set to "ccmp-conf-request-message-type".
confUserID	M	The XCON-USERID received in the request is used.
operation	M	The request operation is specified and is set to delete.
confObjID	I	Not sent, because the conference is deleted.
version	I	Not sent.
confResponse	I	Set to empty.

In the HTTP 200 Ok the following CCMP responses can be included.

Table 12: CCMP Responses for Delete Scheduled Conference

CCMP code and reason phrase	Reason
200 Success	The request was successfully processed.
400 Bad Request	The request could not be understood by the server due to malformed syntax. The client SHOULD NOT repeat the request without modifications.
409 Conflict	There is a set of MTAS application constraints each of which is identified by a phrase. The possible phrase values are given in Table

	25. The phrases start with a number and a colon, which can be used by a client to simplify the implementation of application constraints
421 Invalid confUserID	User missing at the server (this code is returned in the case of requests in which the "confUserID" of the sender is invalid).
500 Internal Server Error	The server cannot complete the required service due to a system internal error. This includes the cases where the Sh interface to the HSS does not respond successfully. Note: Retry to send confRequest request and operation delete until a successful response is received.
501 Not Implemented	Request type or operation envisaged in the protocol, but not implemented in the contacted server. Note: The confRequest request with operations create, retrieve, update and delete, and the userRequest request with operations create, update and delete are implemented.
511 Resources Not Available	This code is used when the CCMP server cannot execute a command because of resource issues e.g. it is not possible to create a new conference due to HSS storage limitation.

4.6 Update Scheduled Conference

MTAS expects the following parameters in a CCMP conference request (confRequest) operation update.

Table 13: CCMP Request parameters for Update Conference, in middle column the values mean mandatory (M)

Parameter		Comment
ccmpRequest	M	To handle a conference this must be set to "ccmp-conf-request-message-type".
confUserID	M	It contains the XCON-USERID of the user. The XCON-USERID is used to identify the IMS subscriber owning the conference.
confObjID	M	The unique XCON-URI identity of the target conference.
operation	M	The request message type is specified to

		update.
confRequest	M	Specialized operation for scheduled conference. In update the conference information is expected to be included, see Table 28.

MTAS responds in a CCMP conference response (confResponse) operation update.

Table 14: CCMP Response parameters for Update Conference, in middle column the values mean mandatory (M), optional (O) and ignored (I).

Parameter		Comment
ccmpResponse	M	When handling a user this is set to "ccmp-conf-response-message-type".
confUserID	M	The XCON-USERID received in the request is used.
confObjID	M	The unique XCON-URI identity of the target conference.
operation	M	The request operation is specified and is set to update.
response-code	M	CCMP response code
response-string	O	CCMP response phrase
version	I	Not sent.
userResponse	I	Set to empty.

In the HTTP 200 Ok the following CCMP faults responses can be included.

Table 15: CCMP Responses for Update Scheduled Conference

CCMP code and reason phrase	Reason
200 Success	The request was successfully processed.
400 Bad Request	The request could not be understood by the server due to malformed syntax. The client SHOULD NOT repeat the request without modifications.
403 Forbidden	Operation not allowed. The following circumstances are defined when this message is sent <ul style="list-style-type: none"> If the scheduled conference service is inactive for the user. If no In Conference Control License is available
409 Conflict	There is a set of MTAS application constraints each of which is identified by a phrase. The possible phrase values are given in Table

	25. The phrases start with a number and a colon, which can be used by a client to simplify the implementation of application constraints
421 Invalid confUserID	User missing at the server (this code is returned in the case of requests in which the "confUserID" of the sender is invalid).
500 Internal Server Error	The server cannot complete the required service due to a system internal error. This includes the cases where the Sh interface to the HSS does not respond successfully.
501 Not Implemented	Request type or operation envisaged in the protocol, but not implemented in the contacted server. Note: The confRequest request with operations create, retrieve, update and delete, and the userRequest request with operations create, update and delete are implemented.
511 Resources Not Available	This code is used when the CCMP server cannot execute a command because of resource issues e.g. it is not possible to create a new conference due to HSS storage limitation.

4.7 Create Scheduled Conference User

MTAS expects the following parameters in a CCMP user request (userRequest) operation create.

Table 16: CCMP Request parameters for Create Scheduled Conference user, in middle column the values mean mandatory (M)

Parameter		Comment
ccmpRequest	M	To handle a conference this must be set to "ccmp-user-request-message-type".
confUserID	M	It contains the XCON-USERID of the user. The XCON-USERID is used to identify the IMS subscriber owning the conference.
confObjID	M	The unique XCON-URI identity of the target conference.
operation	M	The request message type is specified to create.

userRequest	M	Information on the user to be added to the conference, see Table 29.
-------------	---	--

MTAS responds in a CCMP user response (userResponse) operation create.

Table 17: CCMP Response parameters for Create Scheduled Conference User, in middle column the values mean mandatory (M), optional (O) and ignored (I)

Parameter		Comment
ccmpResponse	M	When handling a user this is set to "ccmp-user-response-message-type".
confUserID	M	The XCON-USERID received in the request is used.
confObjID	M	The unique XCON-URI identity of the target conference.
operation	M	The request operation is specified and is set to create.
response-code	M	CCMP response code
response-string	O	CCMP response phrase
version	I	Not sent.
userResponse	M	The response to the userRequest operation, see Table 30.

In the HTTP 200 OK the following CCMP responses can be included.

Table 18: CCMP Responses for Create Scheduled Conference User

CCMP code and reason phrase	Reason
200 Success	The request was successfully processed.
403 Forbidden	Operation not allowed. The following circumstances are defined when this message is sent <ul style="list-style-type: none"> • If the scheduled conference service is inactive for the user. • If no In Conference Control License is available
404 Object Not Found	Target conference object is missing at the server. The "confObjID" parameter is not associated with a conference that the system is holding.
409 Conflict	There is a set of MTAS application constraints each of which is identified by a phrase. The possible phrase values are given in Table 25. The phrases start with a number

	and a colon, which can be used by a client to simplify the implementation of application constraints
421 Invalid confUserID	User missing at the server (this code is returned in the case of requests in which the "confUserID" of the sender is invalid).
500 Internal Server Error	The server cannot complete the required service due to a system internal error. This includes the cases where the Sh interface to the HSS does not respond successfully.
501 Not Implemented	Request type or operation envisaged in the protocol, but not implemented in the contacted server. Note: The confRequest request with operations create, retrieve, update and delete, and the userRequest request with operations create, update and delete are implemented.

4.8 Delete Scheduled Conference User

MTAS expects the following parameters in a CCMP user request (userRequest) operation delete.

Table 19: CCMP Request parameters for Delete Scheduled Conference user, in middle column the values mean mandatory (M)

Parameter		Comment
ccmpRequest	M	To handle a conference this must be set to "ccmp-user-request-message-type".
confUserID	M	It contains the XCON-USERID of the user. The XCON-USERID is used to identify the IMS subscriber owning the conference.
confObjID	M	The unique XCON-URI identity of the target conference.
operation	M	The request message type is specified to delete.
userRequest	M	Information on the user to be added to the conference, see Table 31.

MTAS responds in a CCMP user response (userResponse) operation delete.

Table 20: CCMP Response parameters for Delete Scheduled Conference User, in middle column the values mean mandatory (M), optional (O) and ignored (I)

Parameter		Comment
-----------	--	---------

ccmpResponse	M	When handling a user this is set to "ccmp-user-response-message-type".
confUserID	M	The XCON-USERID received in the request is used.
confObjID	M	The unique XCON-URI identity of the target conference.
operation	M	The request operation is specified and is set to delete.
response-code	M	CCMP response code
response-string	O	CCMP response phrase
version	I	Not sent.
userResponse	I	Set to empty.

In the HTTP 200 OK the following CCMP responses can be included.

Table 21: CCMP Responses for Delete Scheduled Conference User

CCMP code and reason phrase	Reason
200 Success	The request was successfully processed.
403 Forbidden	Operation not allowed. The following circumstances are defined when this message is sent <ul style="list-style-type: none"> • If the scheduled conference service is inactive for the user. • If no In Conference Control License is available
404 Object Not Found	Target conference object is missing at the server. The "confObjID" parameter is not associated with a conference that the system is holding.
409 Conflict	There is a set of MTAS application constraints each of which is identified by a phrase. The possible phrase values are given in Table 25. The phrases start with a number and a colon, which can be used by a client to simplify the implementation of application constraints
420 User Not Found	Target user missing at the server (it is related to the XCON-USERID in the "entity" attribute of the "userInfo" parameter when it is included in userRequests)
421 Invalid confUserID	User missing at the server (this code is returned in the case of

	requests in which the "confUserID" of the sender is invalid).
500 Internal Server Error	The server cannot complete the required service due to a system internal error. This includes the cases where the Sh interface to the HSS does not respond successfully.
501 Not Implemented	Request type or operation envisaged in the protocol, but not implemented in the contacted server. Note: The confRequest request with operations create, retrieve, update and delete, and the userRequest request with operations create, update and delete are implemented.

4.9 Update Scheduled Conference User

MTAS expects the following parameters in a CCMP user request (userRequest) operation update.

Table 22: CCMP Request parameters for Update Scheduled Conference user, in middle column the values mean mandatory (M)

Parameter		Comment
ccmpRequest	M	To handle a conference this must be set to "ccmp-user-request-message-type".
confUserID	M	It contains the XCON-USERID of the user. The XCON-USERID is used to identify the IMS subscriber owning the conference.
confObjID	M	The unique XCON-URI identity of the target conference.
operation	M	The request message type is specified to update.
userRequest	M	Information on the user to be updated, see .

MTAS responds in a CCMP user response (userResponse) operation update.

Table 23: CCMP Response parameters for Update Scheduled Conference User, in middle column the values mean mandatory (M), optional (O) and ignored (I)

Parameter		Comment
ccmpResponse	M	When handling a user this is set to "ccmp-user-response-message-type".
confUserID	M	The XCON-USERID received in the request is used.
confObjID	M	The unique XCON-URI identity of the target conference.

operation	M	The request operation is specified and is set to update.
response-code	M	CCMP response code
response-string	O	CCMP response phrase
version	I	Not sent.
userResponse	I	Set to empty.

In the HTTP 200 OK the following CCMP responses can be included.

Table 24: CCMP Responses for Update Scheduled Conference User

CCMP code and reason phrase	Reason
200 Success	The request was successfully processed.
403 Forbidden	Operation not allowed. The following circumstances are defined when this message is sent <ul style="list-style-type: none"> • If the scheduled conference service is inactive for the user. • If no In Conference Control License is available
404 Object Not Found	Target conference object is missing at the server. The "confObjID" parameter is not associated with a conference that the system is holding.
409 Conflict	There is a set of MTAS application constraints each of which is identified by a phrase. The possible phrase values are given in Table 25. The phrases start with a number and a colon, which can be used by a client to simplify the implementation of application constraints
420 User Not Found	Target user missing at the server (it is related to the XCON-USERID in the "entity" attribute of the "userInfo" parameter when it is included in userRequests)
421 Invalid confUserID	User missing at the server (this code is returned in the case of requests in which the "confUserID" of the sender is invalid).
500 Internal Server Error	The server cannot complete the required service due to a system internal error. This includes the cases where the Sh interface to the

	HSS does not respond successfully.
501 Not Implemented	Request type or operation envisaged in the protocol, but not implemented in the contacted server. Note: The confRequest request with operations create, retrieve, update and delete, and the userRequest request with operations create, update and delete are implemented.

4.10 CCMP reason phrases for 409 Conflict Response

In the CCMP conference response 409 Conflict the following reason phrases exist.

The response-string always starts with "Conflict – " and followed by a phrase.

In the phrase the number before the colon, e.g. "3: ", can be used by the client implementation to simplify the implementation, instead of doing a comparison of the whole string.

The numbering starts with faults that occur in operations:

- Common: 3
- Conference Create, from 4-22
- Conference Delete, no special phrase
- Conference Retrieve, no special phrase
- Conference Update, 401
- User Create, 301, 302, 307-310
- User Delete, 305, 307
- User Update, 304-307

Table 25: CCMP phrases for 409 Conflict Response for Scheduled Conference

Phrase	Description
3: Failed, not valid XML	A POST method body includes XML document that fails schema validation
4: Failed, the element "conference-description" is missing in "conference-info"	A "conference-description" element is missing in "conference-info".
5: Failed, the element "subject" is missing in "conference-info/conference-description"	A "subject" element is missing in "conference-info/conference-description".

Phrase	Description
6: Failed, the element "conf-uris" is missing in "conference-info/conference-description"	A "conf-uris" element is missing in "conference-info/conference-description/".
7: Failed, the element "display-text" with value moderator is missing in "conference-info/conference-description/conf-uris/entry"	A "display-text" element with value moderator is missing in "conference-info/conference-description/conf-uris/entry".
8: Failed, the element "conference-password" for the moderator is missing in "conference-info/conference-description/conf-uris/entry"	A "conference-password" element for the "moderator" is missing in "conference-info/conference-description/conf-uris/entry" or sent empty.
9: Failed, the element "display-text" with value participation is missing in "conference-info/conference-description/conf-uris/entry"	A "display-text" element with value participation is missing in "conference-info/conference-description/conf-uris/entry".
10: Failed, the element "conference-password" for the participation is missing in "conference-info/conference-description/conf-uris/entry"	A "conference-password" element for the "participation" is missing in "conference-info/conference-description/conf-uris/entry" or sent empty.
11: Failed, the element "display-text" has an unsupported value in "conference-info/conference-description/conf-uris/entry"	A "display-text" element has an unsupported value in "conference-info/conference-description/conf-uris/entry". The supported values are moderator and participation.
12: Failed, the element "maximum-user-count" exceeds the limit in "conference-info/conference-description"	A "maximum-user-count" element exceeds the limit in "conference-info/conference-description". The limit is 32.
13: Failed, the element "available-media" is missing in "conference-info/conference-description"	An "available-media" element is missing in "conference-info/conference-description".
14: Failed, the element "display-text" is missing in "conference-info/conference-description/available-media/entry"	A "display-text" element is missing in "conference-info/conference-description/available-media/entry".
15: Failed, the element "type" includes unsupported type "conference-info/conference-description/available-media/entry"	A "type" element includes unsupported type in "conference-info/conference-description/available-media/entry". The supported types are audio and video.

Phrase	Description
16: Failed, the element "entry" is missing in "conference-info/conference-description/conference-time"	When the element "conference-time" exists. An "entry" element is missing in "conference-info/conference-description/conference-time".
17: Failed, the "mixing-start-offset" is missing in "conference-info/conference-description/conference-time/entry"	When the element "conference-time" exists. A "mixing-start-offset" element is missing in "conference-info/conference-description/conference-time/entry".
18: Failed, the element "mixing-start-offset" has a faulty value for the attribute in "conference-info/conference-description/conference-time/entry"	A "mixing-start-offset" element has a faulty value for the attribute in "conference-info/conference-description/conference-time/entry". Allowed value for the attribute required-participant is "moderator".
19: Failed, the element "host-info" is missing in "conference-info"	A "host-info" element is missing in "conference-info".
20: Failed, the element "uris" is missing in "conference-inf/host-info"	A "uris" element is missing in "conference-info/host-info".
21: Failed, exceeded the max number of scheduled conferences	The number of scheduled conferences is exceeded. The limit is 5.
22: Failed, password already in use	The password already in use
201: Failed, the conference is active	OBSOLETE A scheduled conference can not be deleted if it is active.
301: Failed, maximum number of participants is reached.	The maximum number of participants is reached.
302: Failed, capacity license limit is reached.	The limit set by the nodal capacity license for the number of simultaneous conference participants is reached.
304: Failed, no active conference	There is no active conference instance for the specified conference object.
305: Failed, not in stable state	The requested operation cannot be executed as the target user is not in stable state. If the client repeats the request, it MAY succeed.
306: Invalid media stream	The specified media stream is invalid.
307: User is waiting for moderator	The requested operation cannot be executed as the target user is waiting for the operator to join.

Phrase	Description
308: Failed, user is already in the conference	The requested operation cannot be executed as the user to be invited to the conference is already in the conference.
309: Failed, conference is in waiting state	The requested operation cannot be executed as the conference is in waiting state.
310: Failed, not valid uri	The requested operation cannot be executed as the URI in the request is not valid.
401: Failed, the element "join-handling" is missing from request	The "join-handling" element is mandatory in the conference update request.

5 Formal Syntax or Schema

5.1 Requests and Responses

The syntax for CCMP Requests and Responses as defined in ref. [1].

5.2 Conference Information Message format

The conference information and the extension conference information data model are used to define a scheduled conference, as defined in ref. [2] and ref. [3] respectively.

The scheduled conference XML elements supported in MTAS on the interface are described in the following sections. Indentation in the tables is used to indicate nesting in the corresponding XML structure.

5.3 Create conference information

The top level element is conference-info in a request to create a scheduled conference. The elements are mandatory if nothing is mentioned in the description. All other elements/attributes in the schemas are ignored, but remain in the document. In the description it is indicated from which standard the element is introduced, if it is not defined in [2] or [3].

Table 26: Conference Information in a Create Scheduled Conference

XML element			Description
<confInfo>			This is the root element for the conference information.
'entity'			<p>This attribute contains the conference URI that identifies the conference being described in the document.</p> <p>In Create operation the value must be set to "AUTO_GENERATE" or to "AUTO_GENERATE_X" (where X is an integer).</p> <p>The XCON-URI is set in the document before it is stored in HSS. It is included in the parameter "confUserID" in the response.</p> <p>In Retrieve the generated XCON-URI is sent in the response.</p> <p>Introduce in RFC</p>
	<conference-description>		This element describes the conference as a whole.
		<subject>	<p>This element contains the subject of the conference.</p> <p>Optional.</p>
		<conf-uris>	<p>This element contains a sequence of <entry> child elements – each containing the URI to be used in order to access the conference by different signaling means.</p> <p>Two <entry> are expected, one for the "moderator" and another for the "participant".</p>
		<entry>	<p>Every <entry> element contains the following child element.</p>
		<uri>	<p>The value of the URI is mandatory.</p> <p>The URIs are neither used for the "moderator" nor for the "participant". It should be set to "sip:notused".</p>
		<display-text>	<p>This element contains the display text.</p> <p>Two values are valid, "moderator" and "participant".</p>
		<conference-password>	<p>This element contains the password(s) of the conference.</p> <p>The "moderator" and the "participant" have different passwords. All participants have the same passwords.</p>
		<maximum-user-count>	<p>This value represents the overall number of users allowed to join the conference.</p> <p>Optional, if not set the default value 32 is used.</p>
		<available-media>	<p>This element contains a sequence of <entry> child elements available-media, each being indexed by the attribute 'label'.</p>

XML element				Description
			<entry> 'label'	<p>Each <entry> describes a single media stream available to the participants in the conference.</p> <p>This attribute is the media stream identifier assigned by the conferencing server: its value will be unique in the <conference-info> context. The value of this attribute will typically correspond to the Session Description Protocol (SDP).</p> <p>In Create the value must be set to "AUTO_GENERATE" or to "AUTO_GENERATE_X" (where X is an integer). The server generates an identifier. The identifier is set in the document before it is stored in HSS.</p> <p>In Retrieve the generated identifier is sent in the response.</p>
			<display-text>	<p>This element contains the display text for the media stream.</p>
			<type>	<p>This defines the media policy of the created conference. The media types allowed are "audio" , "video" and "application".</p>
			<conference-time>	<p>This element contains the information related to conference policy of a conference. It contains an <entry> element defining the Start-up condition for the conference session. The only policy condition currently supported is whether or not the moderator must join before conference media mixing starts.</p> <p>Optional, if the element is missing the conference starts when the first entity joins the conference.</p>
			<entry>	<p>Every <entry> element contains the following child element.</p>
			<mixing-start-offset> 'required-participant'	<p>This element specifies when the conference media mixing starts.</p> <p>It is not used, but it is mandatory.</p> <p>This attribute is mandatory if the <conference-time> element is provided. The only supported value is 'moderator'</p>
			<host-info>	<p>This element contains information about the entity hosting the conference.</p>
			<uris>	<p>This element contains a set of <entry> child elements, each containing the URI value.</p>
			<entry>	<p>Every <entry> element contains the following child element.</p>
			<uri>	<p>The value of the URI indicates the "owner of the conference".</p>

XML element		Description
	<users>	<p>This element contains information related to conference policy of a conference.</p> <p>The only policy condition currently supported is whether the conference is open or closed for dial-in participants, except the moderator.</p> <p>The <users> element is also a container of <user> child elements, each describing a single participant in the conference. The <user> child elements are not used at conference creation.</p>
	<join-handling>	<p>The <join-handling> element defines the actions used by the conference focus to control conference participation. This element defines the action that the focus is to take when processing a particular request to join a conference. Two values are valid: "block" and "allow".</p> <p>This element is optional. The missing <join-handling> element means "block".</p> <p>Note: value "block" only restricts of dial-in join of participants not using the moderator PIN.</p>

5.4 Retrieve conference information

The top level element is conference-info in a response to a retrieve of a scheduled conference. The information is the same as in create, when nothing is mentioned in the description see Table 26.

In the Table 27 the unique part of a retrieve message is displayed.

In the description it is indicated from which standard the element is introduced, if it is not defined in [2] or [3].

Table 27: Conference Information in a Retrieve Scheduled Conference

XML element		Description
	<confInfo>	<p>By including this element in the conference document, the server can inform the subscribers about the changes in the overall conference information. If the scheduled conference is active this information is collected dynamically.</p>
	'version'	<p>This attribute identifies a state instance in integer form and valid only during the life time of the conference traffic instance.</p> <p>It is set to 1 at start of the conference traffic instance and increased at any state changes.</p> <p>This attribute is not included when no conference traffic instance exists.</p>

XML element				Description
			<user-count>	The value defines the current number of users participating in the conference users at a certain moment.
			<active>	This Boolean element indicates whether the conference is currently active or not.
			<locked>	This Boolean element indicates whether the conference is currently locked or not.
			<users>	This element is a container of <user> child elements, each describing a single participant in the conference. Optional, if the scheduled conference is active the value is presented.
			<user>	This element each describes a single participant in the conference.
			'entity'	This attribute contains the XCON-USERID of the user in the conference and the value MUST be unique among all participants in the conference.
			<associated-aors>	This element contains additional (to the 'entity') URIs being associated with the <user>.
			<entry>	An associated URI entry.
			<uri>	SIP or TEL URI of the participant in the conference. For participants joining with dial-in method it contains the TEL URI received in the P-Asserted-Identity header in the SIP INVITE message. If no P-Asserted-Identity header with TEL URI is received, the SIP URI received in the P-Asserted-Identity header is used. For participants joining with dial-out method it contains the URI received in the order of participant addition.
			<roles>	A role provides the context for the set of conference operations that a participant can perform. This element can contain one of the following values: "moderator", "participant".
			<entry>	A role.
			<endpoint>	By including one or more <endpoint> elements under a parent <user> element, the server can provide the desired level of detail (including the state, media streams, and access information) about the user's devices and signaling sessions taking part in the conference. The server MUST generate this attribute for each <endpoint> element included under the parent <user>, such that its value is unique in the user context. The value received in the "contact" header in the sip INVITE or 200 OK message is used, if available. Otherwise, the same URI is used as in the <associated-aors>.
			'entity'	
			<status>	This element contains the status of the endpoint. The allowed values are "connected", "disconnected", "on-hold", "muted-via-focus" and "alerting".

XML element					Description
				<joining-method>	This element contains the method by which the endpoint joined the conference and can assume the following values: dialed-in, dialed-out.
				<disconnection-method>	This element contains the method by which the endpoint departed the conference and can assume the following values: departed, booted, failed, busy.
				<disconnection-info>	This element contains information about the endpoint's departure from the conference. Note: Only applicable to participants attempted to be joined with dial-out from Web.
				<reason>	This element contains the reason the endpoint departed the conference, in the format defined by RFC 3326.
				<media>	This element contains information about a single media stream and is included for each media stream being established between the focus and the <endpoint>.
				'id'	This attribute is the media stream identifier being generated by the server such that its value is unique in the endpoint context. This attribute is the key to refer to a particular media stream in the conference document.
				<display-text>	This element contains the display text for the media stream. The value of this element correspond the value of the "a=content" SDP attribute: "slides" or "main". If no "a=content" SDP attribute is available, the value of this element correspond to the <display-text> media attribute defined in <available-media><entity label>and is exchanged between a participant and a focus over the signalling connection between them.
				<type>	This defines the media policy of the created conference. The value of this element corresponds to the <type> media attribute defined in <available-media><entity label>and is exchanged between a participant and a focus over the signalling connection between them.
				<label>	The <label> element carries a unique identifier for this stream among all streams in the conference. The value of this element correspond to the SDP "label" media attribute defined in <available-media><entity label>and is exchanged between a participant and a focus over the signaling connection between them.
				<status>	The element <status> indicates the status in both directions of the media stream and has the values "sendrecv", "sendonly", "recvonly", or "inactive". Note that value specifies the direction from the participant's point of view.
				<focus-control-state>	The <focus-control-state> element indicates the media stream restriction set by the CCMP client. This element mirrors back the value of the <status> element set with a CCMP request last time. The value is "sendrecv" means not restriction. This element is defined in [6].

XML element				Description
			<allow-invite-users-dynamically>	The participant is allowed to trigger dial-out INVITE by sending REFER to the focus. Note: This element always has the value of “true” for moderators and “false” for participants.
			<allow-remove-users-dynamically>	The participant is allowed to kick-out participants by sending REFER with method BYE to the focus. Note: This element always has the value of “true” for moderators and “false” for participants.

5.5 Update conference information

The top level element is conference-info in a request to update a scheduled conference. The elements are optional if nothing is mentioned in the description. All other elements/attributes in the schemas are ignored, but remain in the document. In the description it is indicated from which standard the element is introduced, if it is not defined in [2] or [3].

Table 28: Conference Information in a Update Scheduled Conference

XML element				Description
			<confInfo>	This element describes a single participant in the conference to be updated.
			‘entity’	The unique XCON-URI identity of the target conference.
			<users>	This element contains information related to conference policy of a conference. The only policy condition currently supported is whether the conference is open or closed for dial-in participants, except the moderator.
			<join-handling>	The <join-handling> element defines the actions used by the conference focus to control conference participation. This element defines the action that the focus is to take when processing a particular request to join a conference. Two values are valid: “block” and “allow”. Note: value “block” only restricts of dial-in join of participants not using the moderator PIN.

5.6 Create user information

The top level element is user-info in a request to create a scheduled conference user. The elements are mandatory if nothing is mentioned in the description. All other elements/attributes in the schemas are ignored. In the description it is indicated from which standard the element is introduced, if it is not defined in [2] or [3].

Table 29: User Information in a Create Scheduled Conference User

XML element		Description
	<userInfo>	This element describes a single participant in the conference to be joined with dial-out.
	'entity'	This attribute shall contain the fake URI of xcon-userid:AUTO_GENERATE_1@example.com or xcon-userid:AUTO_GENERATE.
	<associated-aors>	This element contains additional (to the 'entity') URIs being associated with the <user>.
	<entry>	An associated URI entry. Only one entry can be specified.
	<uri>	SIP or TEL URI of the user to be invited to the conference.
	<roles>	A role provides the context for the set of conference operations that a participant can perform. This element can contain one of the following values: "moderator", "participant".
	<allow-invite-users-dynamically>	The participant is allowed to trigger dial-out INVITE by sending REFER to the focus. This element is optional. The value of this element is ignored.
	<allow-remove-users-dynamically>	The participant is allowed to kick-out participants by sending REFER with method BYE to the focus. This element is optional. The value of this element is ignored.

The top level element is user-info in a response to a request to create a scheduled conference user.

In the description it is indicated from which standard the element is introduced, if it is not defined in [2] or [3].

Table 30: User Information in a Create Scheduled Conference User Response

XML element		Description
	<userInfo>	This element describes a single participant in the conference to be joined with dial-out.
	'entity'	This attribute contains the XCON-USERID of the joining user.

5.7 Delete user information

The top level element is user-info in a request to delete a scheduled conference user. The elements are mandatory if nothing is mentioned in the description. In the description it is indicated from which standard the element is introduced, if it is not defined in [2] or [3].

Table 31: User Information in a Delete Scheduled Conference User

XML element		Description
	<userInfo>	This element describes a single participant in the conference to be kicked-out.
	'entity'	This attribute contains the XCON-USERID of the user in the conference.

5.8 Update user information

The top level element is user-info in a request to update a scheduled conference user. The elements are mandatory if nothing is mentioned in the description. In the description it is indicated from which standard the element is introduced, if it is not defined in [2] or [3].

Table 32: User Information in a Update Scheduled Conference User

XML element		Description
	<userInfo>	This element describes a single participant in the conference to be updated.
	'entity'	This attribute contains the XCON-USERID of the user in the conference.
	<roles>	A role provides the context for the set of conference operations that a participant can perform. This element can contain one of the following values: "moderator", "participant". This element is OPTIONAL.
	<endpoint>	Top level element of the endpoint.
	'entity'	This attribute contains the identity of the endpoint received in the conference information. This element is OPTIONAL.
	<media>	Top level element of one media stream.
	'id'	This attribute contains the stream identifier received in the conference information.

XML element					Description
				<status>	The element <status> indicates the status in both directions of the media stream and has the values "sendrecv", "sendonly", "recvonly", or "inactive". Note that value specifies the direction from the participant's point of view.

6 Related Standards

- Centralized Conferencing Manipulation Protocol (CCMP) ref. [1]
- A Session Initiation Protocol (SIP) Event Package for Conference State ref. [2]
- Conference Information Data Model for Centralized Conferencing (XCON) ref.[3]
- The Reason Header Field for the Session Initiation Protocol (SIP) ref. [5]

7 Terminology

7.1 Abbreviations

CCMP Centralized Conferencing Manipulation Protocol

7.2 Definitions

Conference Owner (CO) The IMS subscriber that owns and creates the conference (focus).

8 References

- [1] Centralized Conferencing Manipulation Protocol, RFC 6503
- [2] A Session Initiation Protocol (SIP) Event Package for Conference State, RFC 4575
- [3] Conference Information Data Model for Centralized Conferencing (XCON) RFC 6501

- [4] HTTP 1.1
<http://www.w3.org/Protocols/rfc2616/rfc2616.html>
- [5] The Reason Header Field for the Session Initiation Protocol (SIP)
RFC 3326
- [6] DS CCMP Interface XML Schemas
3/190 01-AVA 901 18

9 Annex: POST message, CCMP request/response examples

9.1 CCMP confRequest/confResponse, operation create

This is an example of creating a scheduled conference with a CCMP confRequest/Response.

The conference information is included in the request example below, but is explained in detail in section 5.3.

```
POST /mtasccmp/tel:+46107192000/HTTP/1.1
Content-Length: xxx
Content-Type: application/ccmp+xml
Host: <hostname>:8096
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.0.1 (java 1.5)
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpRequest
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info">
  <ccmpRequest
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-request-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <operation>create</operation>
    <ccmp:confRequest>
      <confInfo entity="xcon:AUTO_GENERATE_1">
        <info:conference-description>
          <info:subject>Phone Meeting</info:subject>
          <info:conf-uris>
            <info:entry>
              <info:uri>sip:user1@telco.com</info:uri>
              <info:display-text>moderator</info:display-text>
              <xcon:conference-password>7654321</xcon:conference-password>
            </info:entry>
            <info:entry>
              <info:uri>sip:notused</info:uri>
              <info:display-text>participation</info:display-text>
              <xcon:conference-password>7123456</xcon:conference-password>
            </info:entry>
          </info:conf-uris>
          <info:maximum-user-count>8</info:maximum-user-count>
          <info:available-media>
            <info:entry label="AUTO_GENERATE_2">
              <info:display-text>audio stream</info:display-text>
              <info:type>audio</info:type>
            </info:entry>
            <info:entry label="AUTO_GENERATE_3">
              <info:display-text>video stream</info:display-text>
              <info:type>video</info:type>
            </info:entry>
          </info:available-media>
          <xcon:conference-time>
            <xcon:entry>
              <xcon:mixing-start-offset required-participant="moderator">
                2010-01-27T14:29:00Z
              </xcon:mixing-start-offset>
            </xcon:entry>
          </xcon:conference-time>
        </info:conference-description>
        <info:host-info>
          <info:uris>
            <info:entry>
              <info:uri>sip:user1@telco.com</info:uri>
            </info:entry>
```

```

        </info:uris>
        </info:host-info>
        <info:users>
          <xcon:join-handling>allow</xcon:join-handling>
        </info:users>
      </confInfo>
    </ccmp:confRequest>
  </ccmpRequest>
</ccmp:ccmpRequest>

```

An example of a successful, 200 OK CCMP confResponse with operation create.

```

HTTP/1.1 200 OK
X-Powered-By: Servlet/2.5
Server: Sun GlassFish Communications Server 1.5
Content-Type: application/ccmp+xml;charset=ISO-8859-1
Content-Length: xxx
Date: Thu, 04 Feb 2010 14:47:56 GMT

```

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>create</operation>
    <response-code>200</response-code>
    <response-string>Success</response-string>
    <version>1</version>
  </ccmpResponse>
</ccmp:ccmpResponse>

```

9.2 CCMP confResponse, operation create, error 421

This is an example of a HTTP 200 Ok including a CCMP Response with error message 421 Invalid confUserID.

HTTP/1.1 200 OK
X-Powered-By: Servlet/2.5
Server: Sun GlassFish Communications Server 1.5
Content-Type: application/ccmp+xml;charset=ISO-8859-1
Content-Length: xxx
Date: Thu, 04 Feb 2010 14:47:56 GMT

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <operation>create</operation>
    <response-code>421</response-code>
    <response-string>Invalid confUserID</response-string>
    <ccmp:confResponse/>
  </ccmpResponse>
</ccmp:ccmpResponse>
```

9.3 CCMP confResponse, operation create, error 409

This is an example of a HTTP 200 Ok including a CCMP Response with error message 409 Conflict with application constraint.

HTTP/1.1 200 OK
X-Powered-By: Servlet/2.5
Server: Sun GlassFish Communications Server 1.5
Content-Type: application/ccmp+xml;charset=ISO-8859-1
Content-Length: xxx
Date: Thu, 04 Feb 2010 14:47:56 GMT

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <operation>create</operation>
    <response-code>409 </response-code>
    <response-string> Conflict - 16: Failed, the element "<display-text>"
with value moderator is missing in "<conference-info><conference-description><conf-
uris><entry>"
    </response-string>
  </ccmp:confResponse/>
</ccmp:ccmpResponse>
</ccmp:ccmpResponse>
```

9.4 CCMP confRequest/confResponse, operation retrieve

This is an example of retrieving a scheduled conference with a CCMP confRequest/Response.

The conference information is included in the response example below, but is explained in detailed in section 5.4.

POST /mtasccmp/tel:+46107192000/ HTTP/1.1
Content-Length: xxx
Content-Type: application/ccmp+xml
Host: <hostname>:8096
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.0.1 (java 1.5)

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpRequest
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info">
  <ccmpRequest xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-request-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>retrieve</operation>
  </ccmp:confRequest/>
</ccmp:ccmpRequest>
</ccmp:ccmpRequest>
```

An example of a successful, 200 OK CCMP confResponse with operation retrieve.

HTTP/1.1 200 OK
 X-Powered-By: Servlet/2.5
 Server: Sun GlassFish Communications Server 1.5
 Content-Type: application/ccmp+xml;charset=ISO-8859-1
 Content-Length: xxx
 Date: Thu, 04 Feb 2010 14:47:56 GMT

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:mmt-ccmp="http://schemas.ericsson.com/mmtel-ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>retrieve</operation>
    <response-code>200</response-code>
    <response-string>Success</response-string>
    <ccmp:confResponse>
      <confInfo entity="xcon:1_user1@telco.com" version="16">
        <info:conference-description>
          <info:subject>Phone Meeting</info:subject>
          <info:conf-uris>
            <info:entry>
              <info:uri>sip:user1@telco.com</info:uri>
              <info:display-text>moderator</info:display-text>
              <xcon:conference-password>7654321</xcon:conference-password>
            </info:entry>
            <info:entry>
              <info:uri>sip:notused</info:uri>
              <info:display-text>participation</info:display-text>
              <xcon:conference-password>7123456</xcon:conference-password>
            </info:entry>
          </info:conf-uris>
          <info:maximum-user-count>8</info:maximum-user-count>
          <info:available-media>
            <info:entry label="1">
              <info:display-text>audio stream</info:display-text>
              <info:type>audio</info:type>
            </info:entry>
            <info:entry label="2">
              <info:display-text>video stream</info:display-text>
              <info:type>video</info:type>
            </info:entry>
          </info:available-media>
          <xcon:conference-time>
            <xcon:entry>
              <xcon:mixing-start-offset required-participant="moderator">
                2010-01-27T14:29:00Z
              </xcon:mixing-start-offset>
            </xcon:entry>
          </xcon:conference-time>
        </info:conference-description>
        <info:host-info>
          <info:uris>

```

```

    <info:entry>
      <info:uri>sip:user1@telco.com</info:uri>
    </info:entry>
  </info:uris>
</info:host-info>
<info:conference-state>
  <info:user-count>2</info:user-count>
  <info:active>true</info:active>
  <info:locked>>false</info:locked>
</info:conference-state>
<info:users>
  <info:user entity="xcon-userid:12345_0">
    <info:associated-aors>
      <info:entry>
        <info:uri>sip:user1@telco.com</info:uri>
        <info:display-text>pintyoke</info:display-text>
      </info:entry>
    </info:associated-aors>
    <info:roles>
      <info:entry>moderator</info:entry>
    </info:roles>
    <info:endpoint entity="sip:user1@telco.com;transport=tcp">
      <info:status>connected</info:status>
      <info:joining-method>dialed-in</info:joining-method>
      <info:media id="0">
        <info:display-text>audio stream</info:display-text>
        <info:type>audio</info:type>
        <info:label>1</info:label>
        <info:status>sendrecv</info:status>
        <mmt-ccmp:focus-control-state>sendrecv</crs:focus-control-state>
      </info:media>
      <info:media id="1">
        <info:display-text>video stream</info:display-text>
        <info:type>video</info:type>
        <info:label>2</info:label>
        <info:status>sendrecv</info:status>
        <mmt-ccmp:focus-control-state>sendrecv</crs:focus-control-state>
      </info:media>
    </info:endpoint>
    <xcon:allow-invite-users-dynamically>true</xcon:allow-invite-users-
dynamically>
    <xcon:allow-remove-users-dynamically>true</xcon:allow-remove-users-
dynamically>
  </info:user>
  <info:user entity="xcon-userid:12345_1">
    <info:associated-aors>
      <info:entry>
        <info:uri>sip:user12@telco.com</info:uri>
      </info:entry>
    </info:associated-aors>
    <info:roles>
      <info:entry>participant</info:entry>
    </info:roles>
    <info:endpoint entity="sip:user12@telco.com;transport=tcp">
      <info:status>on-hold</info:status>
      <info:joining-method>dialed-out</info:joining-method>

```

```
<info:media id="0">
  <info:display-text>audio stream</info:display-text>
  <info:type>audio</info:type>
  <info:label>1</info:label>
  <info:status>inactive</info:status>
  <mmt-ccmp:focus-control-state>sendonly</crs:focus-control-state>
</info:media>
<info:media id="1">
  <info:display-text>video stream</info:display-text>
  <info:type>video</info:type>
  <info:label>2</info:label>
  <info:status>inactive</info:status>
  <mmt-ccmp:focus-control-state>sendonly</crs:focus-control-state>
</info:media>
<xcon:allow-invite-users-dynamically>>false</xcon:allow-invite-users-
dynamically>
<xcon:allow-remove-users-dynamically>>false</xcon:allow-remove-users-
dynamically>
  </info:endpoint>
</info:user>
<info:user entity="xcon-userid:12345_2">
  <info:associated-aors>
    <info:entry>
      <info:uri>sip:user16@telco.com</info:uri>
    </info:entry>
  </info:associated-aors>
  <info:roles>
    <info:entry>participant</info:entry>
  </info:roles>
  <info:endpoint entity="sip:user16@telco.com;transport=tcp">
    <info:status>disconnected</info:status>
    <info:joining-method>dialed-out</info:joining-method>
    <info:disconnection-method>failed</info:disconnection-method>
    <info:disconnection-info>
      <info:reason>SIP ;cause=603 ;text="Decline"</info:reason>
    </info:disconnection-info>
  </info:endpoint>
</info:user>
  <xcon:join-handling>allow</xcon:join-handling>
</info:users>
</confInfo>
</ccmp:confResponse>
</ccmpResponse>
</ccmp:ccmpResponse>
```

9.5 CCMP confRequest/confResponse, operation delete

This is an example of deleting a scheduled conference with a CCMP confRequest/Response.

```
POST /mtasccmp/tel:+46107192000/ HTTP/1.1
Content-Length: xxx
Content-Type: application/ccmp+xml
Host: <hostname>:8096
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.0.1 (java 1.5)

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpRequest
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info">
  <ccmpRequest
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-request-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>delete</operation>
    <ccmp:confRequest/>
  </ccmpRequest>
</ccmp:ccmpRequest>
```

An example of a successful, 200 OK CCMP confResponse with operation delete.

```
HTTP/1.1 200 OK
X-Powered-By: Servlet/2.5
Server: Sun GlassFish Communications Server 1.5
Content-Type: application/ccmp+xml; charset=ISO-8859-1
Content-Length: xxx
Date: Thu, 04 Feb 2010 14:47:56 GMT

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>delete</operation>
    <response-code>200</response-code>
    <response-string>Success</response-string>
    <ccmp:confResponse/>
  </ccmpResponse>
</ccmp:ccmpResponse>
```

9.6 CCMP confRequest/confResponse, operation update

This is an example of updating a scheduled conference with a CCMP confRequest/Response.

The conference information is included in the request example below, but is explained in detail in section 5.5.

```
POST /mtasccmp/tel:+46107192000/ HTTP/1.1
Content-Length: xxx
Content-Type: application/ccmp+xml
Host: <hostname>:8096
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.0.1 (java 1.5)

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpRequest
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info">
  <ccmpRequest
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>update</operation>
    <ccmp:confRequest>
      <confInfo entity="xcon:1_user1@telco.com">
        <info:users>
          <xcon:join-handling>block</xcon:join-handling>
        </info:users>
      </confInfo>
    </ccmp:confRequest>
  </ccmpRequest>
</ccmp:ccmpRequest>
```

An example of a successful, 200 OK CCMP confResponse with operation update.

```
HTTP/1.1 200 OK
X-Powered-By: Servlet/2.5
Server: Sun GlassFish Communications Server 1.5
Content-Type: application/ccmp+xml; charset=ISO-8859-1
Content-Length: xxx
Date: Thu, 04 Feb 2010 14:47:56 GMT

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-conf-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>update</operation>
    <response-code>200</response-code>
    <response-string>Success</response-string>
  </ccmp:confResponse>
</ccmp:ccmpResponse>
```

9.7 CCMP userRequest/userResponse, operation create

This is an example of creating a scheduled conference user with a CCMP userRequest/Response.

The user information is included in the request example below, but is explained in detail in section 5.6.

```
POST /mtasccmp/tel:+46107192000/ HTTP/1.1
Content-Length: xxx
Content-Type: application/ccmp+xml
Host: <hostname>:8096
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.0.1 (java 1.5)

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpRequest
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info">
  <ccmpRequest
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-user-request-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>create</operation>
    <ccmp:userRequest>
      <userInfo entity="xcon-userid:AUTO_GENERATE_1@example.com">
        <info:associated-aors>
          <info:entry>
            <info:uri>sip:user12@telco.com</info:uri>
          </info:entry>
        </info:associated-aors>
        <info:roles>
          <info:entry>participant</info:entry>
        </info:roles>
        <xcon:allow-invite-users-dynamically>true</xcon:allow-invite-
users-dynamically>
        <xcon:allow-remove-users-dynamically>true</xcon:allow-
remove-users-dynamically>
      </userInfo>
    </ccmp:userRequest>
  </ccmp:ccmpRequest>
```

An example of a successful, 200 OK CCMP userResponse with operation create.

```

HTTP/1.1 200 OK
X-Powered-By: Servlet/2.5
Server: Sun GlassFish Communications Server 1.5
Content-Type: application/ccmp+xml;charset=ISO-8859-1
Content-Length: xxx
Date: Thu, 04 Feb 2010 14:47:56 GMT

```

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-user-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>create</operation>
    <response-code>200</response-code>
    <response-string>Success</response-string>
    <ccmp:userResponse>
      <userInfo entity="xcon-userid:12345_0"/>
    </ccmp:userResponse>
  </ccmpResponse>
</ccmp:ccmpResponse>

```

9.8 CCMP userRequest/userResponse, operation delete

This is an example of deleting a scheduled conference user with a CCMP userRequest/Response.

The user information is included in the request example below, but is explained in detail in section 5.7.

```

POST /mtasccmp/tel:+46107192000/ HTTP/1.1
Content-Length: xxx
Content-Type: application/ccmp+xml
Host: <hostname>:8096
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.0.1 (java 1.5)

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpRequest
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info">
  <ccmpRequest
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-user-request-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>delete</operation>
    <ccmp:userRequest>
      <userInfo entity="xcon-userid:12345_0"/>
    </ccmp:userRequest>
  </ccmpRequest>
</ccmp:ccmpRequest>

```

An example of a successful, 200 OK CCMP userResponse with operation delete.

```

HTTP/1.1 200 OK
X-Powered-By: Servlet/2.5
Server: Sun GlassFish Communications Server 1.5
Content-Type: application/ccmp+xml; charset=ISO-8859-1
Content-Length: xxx
Date: Thu, 04 Feb 2010 14:47:56 GMT

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-user-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>delete</operation>
    <response-code>200</response-code>
    <response-string>Success</response-string>
    <ccmp:userResponse />
  </ccmpResponse>
</ccmp:ccmpResponse>

```

9.9 CCMP userRequest/userResponse, operation update

This is an example of updating a scheduled conference user with a CCMP userRequest/Response.

The user information is included in the request example below, but is explained in detail in section 5.8.

```
POST /mtasccmp/tel:+46107192000/ HTTP/1.1
```

```
Content-Length: xxx
```

```
Content-Type: application/ccmp+xml
```

```
Host: <hostname>:8096
```

```
Connection: Keep-Alive
```

```
User-Agent: Apache-HttpClient/4.0.1 (java 1.5)
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpRequest
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp"
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info">
  <ccmpRequest
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-user-request-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>update</operation>
    <ccmp:userRequest>
      <userInfo entity="xcon-userid:12345_0">
        <info:roles>
          <info:entry>moderator</info:entry>
        </info:roles>
        <info:endpoint entity="sip:user12@telco.com;transport=tcp">
          <info:media id="0">
            <info:status>recvonly</info:status>
          </info:media>
        </info:endpoint>
      </userInfo>
    </ccmp:userRequest>
  </ccmpRequest>
</ccmp:ccmpRequest>
```

An example of a successful, 200 OK CCMP userResponse with operation update.

HTTP/1.1 200 OK
 X-Powered-By: Servlet/2.5
 Server: Sun GlassFish Communications Server 1.5
 Content-Type: application/ccmp+xml;charset=ISO-8859-1
 Content-Length: xxx
 Date: Thu, 04 Feb 2010 14:47:56 GMT

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ccmp:ccmpResponse
  xmlns:xcon="urn:ietf:params:xml:ns:xcon-conference-info"
  xmlns:info="urn:ietf:params:xml:ns:conference-info"
  xmlns:ccmp="urn:ietf:params:xml:ns:xcon:ccmp">
  <ccmpResponse xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="ccmp:ccmp-user-response-message-type">
    <confUserID>xcon-userid:user1@telco.com</confUserID>
    <confObjID>xcon:1_user1@telco.com</confObjID>
    <operation>update</operation>
    <response-code>200</response-code>
    <response-string>Success</response-string>
    <ccmp:userResponse />
  </ccmpResponse>
</ccmp:ccmpResponse>
```