International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.609.9
(09/2020)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

OSI networking and system aspects – Networking

Managed P2P communications: Overlay content management protocol

Recommendation ITU-T X.609.9

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEMS INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.369
IP-based networks	X.370–X.379
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	**X.600–X.629**
Efficiency	X.630–X.639
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems management framework and architecture	X.700–X.709
Management communication service and protocol	X.710–X.719
Structure of management information	X.720–X.729
Management functions and ODMA functions	X.730–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, concurrency and recovery	X.850–X.859
Transaction processing	X.860–X.879
Remote operations	X.880–X.889
Generic applications of ASN.1	X.890–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	X.1000–X.1099
SECURE APPLICATIONS AND SERVICES (1)	X.1100–X.1199
CYBERSPACE SECURITY	X.1200–X.1299
SECURE APPLICATIONS AND SERVICES (2)	X.1300–X.1499
CYBERSECURITY INFORMATION EXCHANGE	X.1500–X.1599
CLOUD COMPUTING SECURITY	X.1600–X.1699
QUANTUM COMMUNICATION	X.1700–X.1729
DATA SECURITY	X.1750–X.1799
5G SECURITY	X.1800–X.1819

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.609.9

Managed P2P communications: Overlay content management protocol

Summary

Recommendation ITU-T X.609.9 specifies an overlay content management protocol (OCMP) that runs on an interface between an index server and a peer to carry meta-information of overlay content over managed peer-to-peer (MP2P) architecture defined in Recommendation ITU-T X.609. The meta-information includes attributes of a content to be distributed and mapping information with an overlay network. This Recommendation provides message formats and protocol operations.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T X.609.9	2020-09-29	11	11.1002/1000/14421

Keywords

Managed peer-to-peer, management, overlay content, protocol.

[*] To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Page

Recommendation ITU-T X.609.9

Managed P2P communications: Overlay content management protocol

1 Scope

This Recommendation specifies the overlay content management protocol (OCMP) to be used by peers and index server. The purpose of this protocol is to carry meta-information of an overlay content to be distributed over managed peer-to-peer (MP2P) network.

It describes following details:

– protocol overview;

– protocol operations including information flows;

– protocol messages and its parameters.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T X.609] Recommendation ITU-T X.609 (2015), *Managed P2P communications: Functional architecture.*

[ITU-T X.609.3] Recommendation ITU-T X.609.3 (2017), *Managed P2P communications: Multimedia streaming signalling requirements.*

[ITU-T X.609.6] Recommendation ITU-T X.609.6 (2018), *Managed P2P communications: Content distribution signalling requirements.*

[ITU-T X.609.10] Recommendation ITU-T X.609.10 (2020), *Managed P2P communications: Signalling requirements for data streaming.*

[IETF RFC 7159] IETF RFC 7159 (2014), *The JavaScript Object Notation (JSON) Data Interchange Format.*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 overlay network [b-ITU-T X.1162]: An overlay network is a virtual network that runs on top of another network. Like any other network, the overlay network comprises a set of nodes and links between them. Because the links are logical ones, they may correspond to many physical links of the underlying network.

3.1.2 peer [b-ITU-T X.1161]: Communication node on P2P network that functions simultaneously as both "client" and "server" to the other nodes on the network.

3.1.3 peer-to-peer (P2P) [b-ITU-T Y.2206]: A system is considered to be P2P if the nodes of the system share their resources in order to provide the service the system supports. The nodes in the system both provide services to other nodes and request services from other nodes.

NOTE – Peer is the node in a P2P system.

3.1.4 managed P2P [b-ISO/IEC TR 20002]: P2P with manageability features to manage the P2P-based service and P2P network by the P2P participants such as P2P service provider, ISP, and peer.

3.1.5 buffermap [ITU-T X.609]: A map showing downloading status of fragments comprising a shared content.

3.1.6 fragment [ITU-T X.609]: A piece of the shared content.

3.1.7 fragmentation [ITU-T X.609]: A process that divides the shared content into multiple fragments for sharing the content in a distributed manner.

3.1.8 source peer [ITU-T X.609.3]: A peer that streams the multimedia contents to the overlay network. The peer only provides content data to other peers and does not receive it. This peer generates fragments using the multimedia data received from the contents source.

3.1.9 client peer [ITU-T X.609.3]: A peer that sends fragments received from other peers to other peers and does not generate its own fragments.

3.2 Terms defined in this Recommendation

This Recommendation defines the following term:

3.2.1 overlay content: A content to be distributed through an overlay network.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

CBR	Constant Bit Rate
CMIM	Content Meta-Info Management
CVBR	Constrained Variable Bit Rate
FE	Functional Entity
FQDN	Fully Qualified Domain Name
HTTP	Hypertext Transfer Protocol
ID	Identifier
IDR	Instantaneous Decode Refresh
IXS	Index Server
KiB	Kibibyte
MIC	Meta Information Control
MP2P	Managed Peer-to-Peer
OMS	Overlay Management Server
OCMP	Overlay Content Management Protocol
VBR	Variable Bit Rate

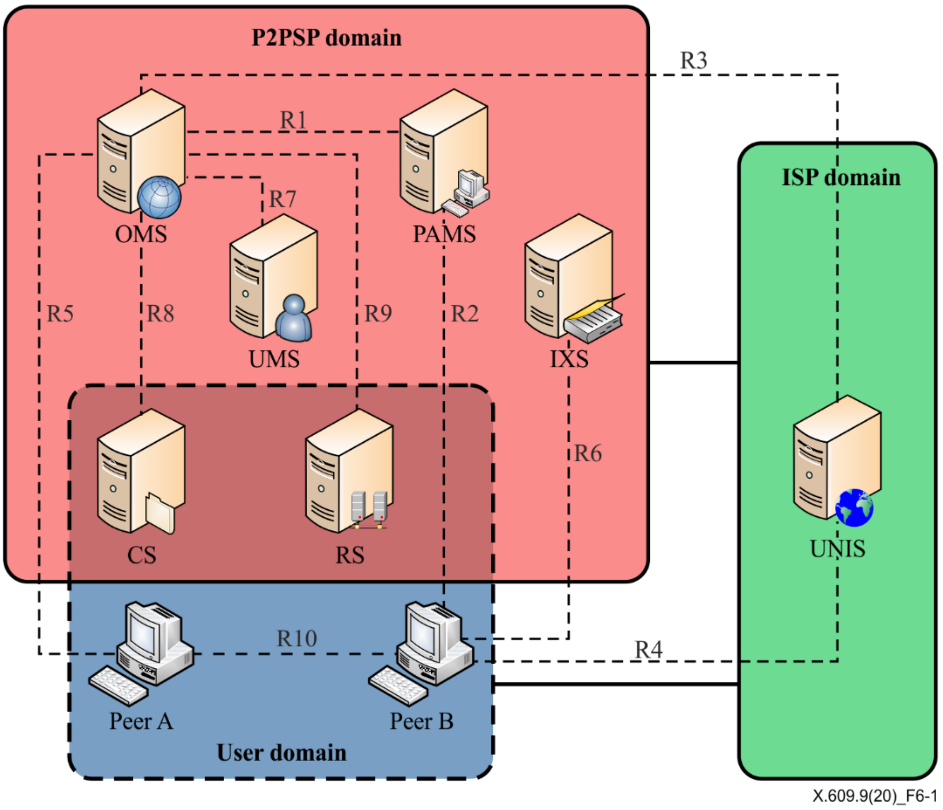
5 Conventions

See clause 8.1 for the grammar used in object representation.

6 Overview

When it comes to specify a content to be distributed over an overlay network that is specified in [ITU-T X.609], the content needs meta-information describing the content. The index server (IXS) maintains the meta-information related to the overlay content, and also maintains mapping information between the overlay content and an overlay network which is created in the overlay management server (OMS). This meta-information is used to specify the size of fragment, overlay network identifier, hash of fragment for assuring integrity, etc.

Figure 6-1 shows a framework and reference points of managed peer-to-peer (MP2P) that is defined in [ITU-T X.609]. This Recommendation specifies an overlay content management protocol for reference point R6.



UMS: user profile management server
OMS: overlay management server
PAMS: peer activity management server
RS: relay server
CS: cache server
IXS: index server
UNIS: underlying network information server

Figure 6-1 – Framework and reference points of MP2P [ITU-T X.609]

7 Protocol operations

This clause describes basic and extended operations with high-level information flows. The basic operations include creation, removal, update and query overlay content information. In the extended operations, this describes the protocol operations for different types of services.

7.1 Basic operations

This clause describes the basic procedural operations on managing an overlay content information management protocol.

7.1.1 Overlay content information registration

This clause provides procedures of registration of an overlay content. It is assumed that the source peer had already created an overlay network by interactions with the overlay management server

(OMS) to map the overlay content to the specific overlay network. Figure 7-1 shows procedures for overlay content registration.

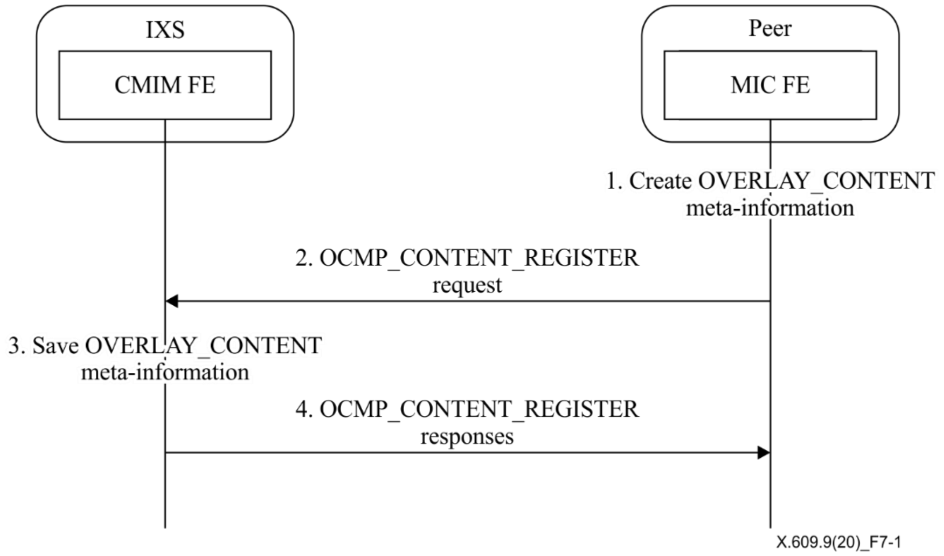

IXS
CMIM FE
Peer
MIC FE
1. Create OVERLAY_CONTENT meta-information
2. OCMP_CONTENT_REGISTER request
3. Save OVERLAY_CONTENT meta-information
4. OCMP_CONTENT_REGISTER responses
X.609.9(20)_F7-1

Figure 7-1 – Procedures for overlay content registration

1. Meta information control FE (MIC FE) of a source peer creates a meta-information and embeds it into OVERLAY_CONTENT element, which is specified in clause 8.1.1.
2. MIC FE of the source peer sends OCMP_CONTENT_REGISTER request, which is specified in clause 8.2.1.1, to IXS.
3. Content meta-info management FE (CMIM FE) of IXS stores the meta-information when validation is successful.
4. CMIM FE of IXS sends OCMP_CONTENT_REGISTER response, which is specified in clause 8.1.1.2, to the peer.

7.1.2 Overlay content information update

When it needs to modify an overlay content, the source peer sends a request with updated meta-information to IXS. On receiving this request, IXS increments the version of the meta-information. Figure 7-2 shows the procedures for the overlay content update.

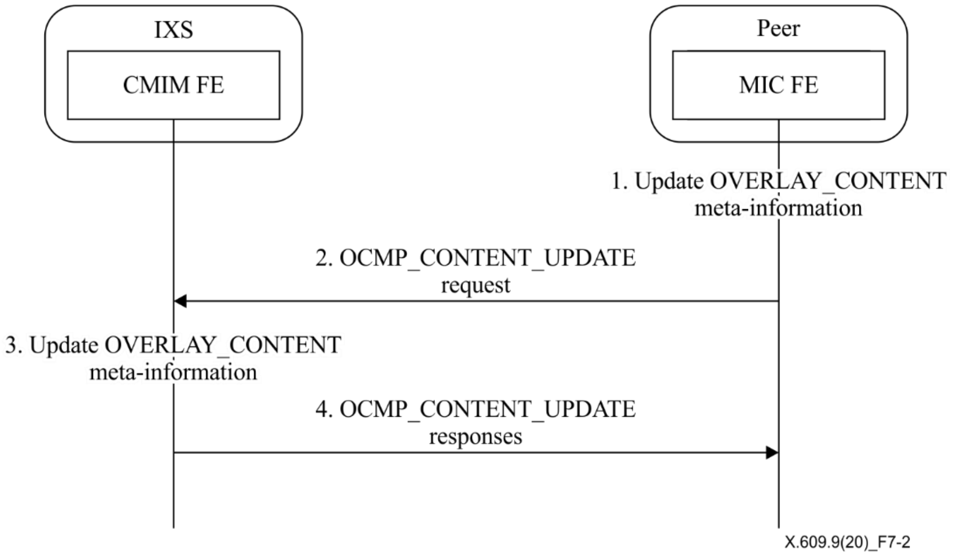

IXS
CMIM FE
Peer
MIC FE
1. Update OVERLAY_CONTENT meta-information
2. OCMP_CONTENT_UPDATE request
3. Update OVERLAY_CONTENT meta-information
4. OCMP_CONTENT_UPDATE responses
X.609.9(20)_F7-2

Figure 7-2 – Procedures for overlay content update

1. MIC FE of a source peer updates the overlay content and updates the FILES element of OVERLAY_CONTENT element, which is specified in clauses 8.1.1 and 8.1.2. If a HASHES element exists in the OVERLAY_CONTENT element, it is also to be updated accordingly. Finally, the MIC FE increments the version of the OVERLAY_CONTENT element.
2. MIC FE of the source peer sends OCMP_CONTENT_UPDATE request, which is specified in clause 8.2.2.1, to IXS.
3. CMIM FE of IXS saves the meta-information if validation check is successful.
4. CMIM FE of IXS sends OCMP_CONTENT_UPDATE response, which is specified in clause 8.2.2.2, to the peer.

7.1.3 Overlay content information removal

When a source peer wants to remove a specific overlay content that will be mapped with an overlay network, it sends removal request to IXS. Unlike the modification specified in clause 7.1.2, this removes the overlay content. Figure 7-3 shows the procedures for deregistration/removal of an overlay content from IXS.

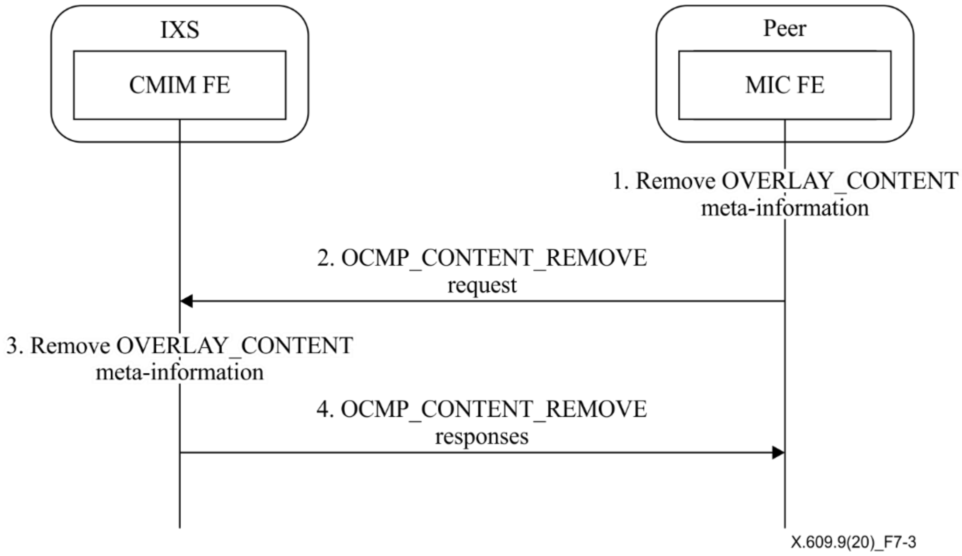

IXS
CMIM FE
Peer
MIC FE
1. Remove OVERLAY_CONTENT meta-information
2. OCMP_CONTENT_REMOVE request
3. Remove OVERLAY_CONTENT meta-information
4. OCMP_CONTENT_REMOVE responses
X.609.9(20)_F7-3

Figure 7-3 – Procedures for overlay content removal

1. MIC FE of a source peer removes OVERLAY_CONTENT element.

   NOTE – It is also possible to perform this procedure after receiving response for the removal request. This is implementation dependent.
2. MIC FE of the source peer sends OCMP_CONTENT_REMOVE request, which is specified in clause 8.2.3.1, to IXS.
3. CMIM FE of IXS removes the meta-information if validation check is successful.
4. CMIM FE of IXS sends OCMP_CONTENT_REMOVE response, which is specified in clause 8.2.3.2, to the peer.

7.1.4 Overlay content information query

This clause describes the procedures for querying overlay content by a client peer.

When a peer wants to receive a specific overlay content, it needs to find a meta-information that indicates the attribute of the content and the overlay network information. IXS provides necessary information to the requesting peer as shown in Figure 7-4.

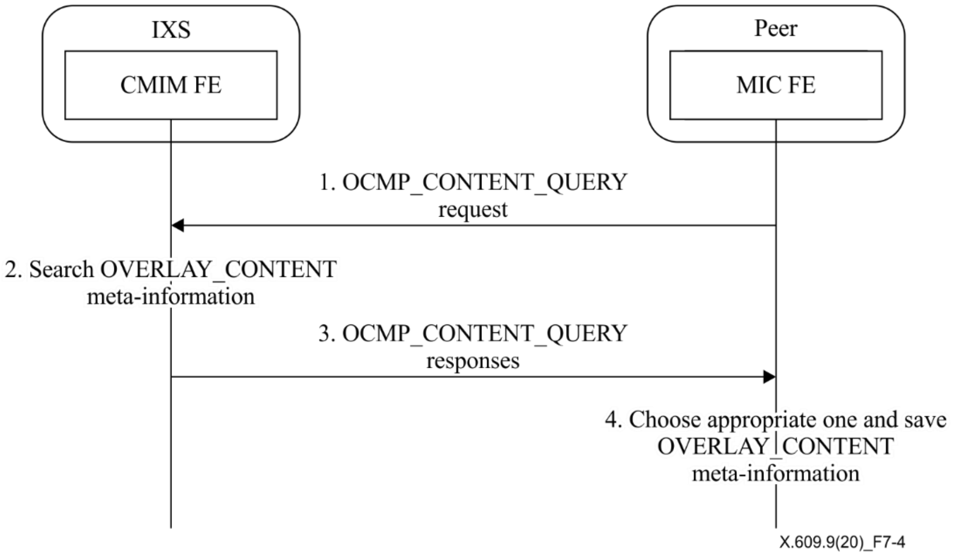

IXS
CMIM FE
Peer
MIC FE
1. OCMP_CONTENT_QUERY request
2. Search OVERLAY_CONTENT meta-information
3. OCMP_CONTENT_QUERY responses
4. Choose appropriate one and save OVERLAY_CONTENT meta-information
X.609.9(20)_F7-4

Figure 7-4 – Procedures for overlay content query

1. MIC FE of a client peer sends OCMP_CONTENT_QUERY request, which is specified in clause 8.2.4.1, to IXS to find the overlay content that it wants to receive. When it finds the specific content, it embeds an appropriate keyword for title, owner-id and content-id.
2. CMIM FE of IXS searches overlay contents that match the keyword. If there is no keyword in the request, it returns the list of all overlay networks of the IXS.
3. CMIM FE of IXS sends OCMP_CONTENT_ QUERY response, which is specified in clause 8.2.4.2, to the peer.
4. A user selects one overlay content of the list, and MIC FE of the client peer save the relevant OVERLAY_CONTENT meta-information.

7.2 Extended operations

Since the characteristics of each service are different, the usage of the overlay content management protocol will be also not be same. Clauses 7.21, 7.2.2 and 7.2.3 describe extended operations for content distribution service, multimedia streaming service and data streaming service, respectively.

7.2.1 Content distribution service

[ITU-T X.609.6] describes the requirements for reference point R6 (see Figure 6-1) for content distribution service, and this clause describes extended operations to support the content distribution service.

Table 7-1 shows in detail which of the parameters of OVERLAY_CONTENT is included for each overlay content management protocol (OCMP) message for the content distribution service.

Table 7-1 – Parameters of OVERLAY_CONTENT for content distribution service

Element		REGISTER		UPDATE		REMOVE		QUERY	
OVERLAY_CONTENT		req	rsp	req	rsp	req	rsp	req	rsp
	version	–	m	m	m	m	–	–	m
	title	m	–	o	–	–	–	o	m
	content-id	–	m	m	–	m	–	o	m
	owner-id	m	–	m	–	m	–	o	m
	overlay-id	m	–	–	–	–	–	–	c1
	oms-address	m	–	–	–	–	–	–	c1
	piece-size	m	–	–	–	–	–	–	c1
	content-type	m	–	–	–	–	–	–	c1
	piece-num	m	–	m	–	–	–	–	c1
	auth	m	–	m	–	m	–	–	c1
	files	m	–	m	–	–	–	–	c1
	desc	o	–	o	–	–	–	o	o

– c1: When the OVERLAY_CONTENT_QUERY request message has one of title, content-id and owner-id in the parameter, it shall include the parameters. If not, it is optional;

– m: mandatory;

– o: optional.

When it comes to register a new content, the owner of the content creates an index file containing the OVERLAY_CONTENT element and sends an OVERLAY_CONTENT_REGISTER request to IXS. When the IXS registers the index file, it issues a new content identifier for the overlay content and embeds the identifier in the *content-id* parameter of the response. On updating the index file, the owner sends an OVERLAY_CONTENT_UPDATE message containing the OVERLAY_CONTENT element as shown in Table 7-1. When the update procedure is successful, the IXS increments the *version* parameter before embedding the OVERLAY_CONTENT element in the response.

For the client peers that need the content, the peer sends an OVERLAY_CONTENT_QUERY message with parameters regarding *title*, *content-id* or *overlay-id*. If the message does not contain those parameters, the IXS responds with the list of the overlay contents including only the *title*, *content-id* and *overlay-id* for each index file. This is because it is preferable not to provide all the information of all overlay contents to avoid performance issues. On receiving the index file from IXS, the peer creates directory structures by using the *files* parameter.

Figure 7-5 shows a procedure for the content distribution service initiation phase. To register a content distribution session, a source peer sends an OCMP_CONTENT_REGISTER request to IXS. The detailed parameters of the message are listed in Table 7-1.

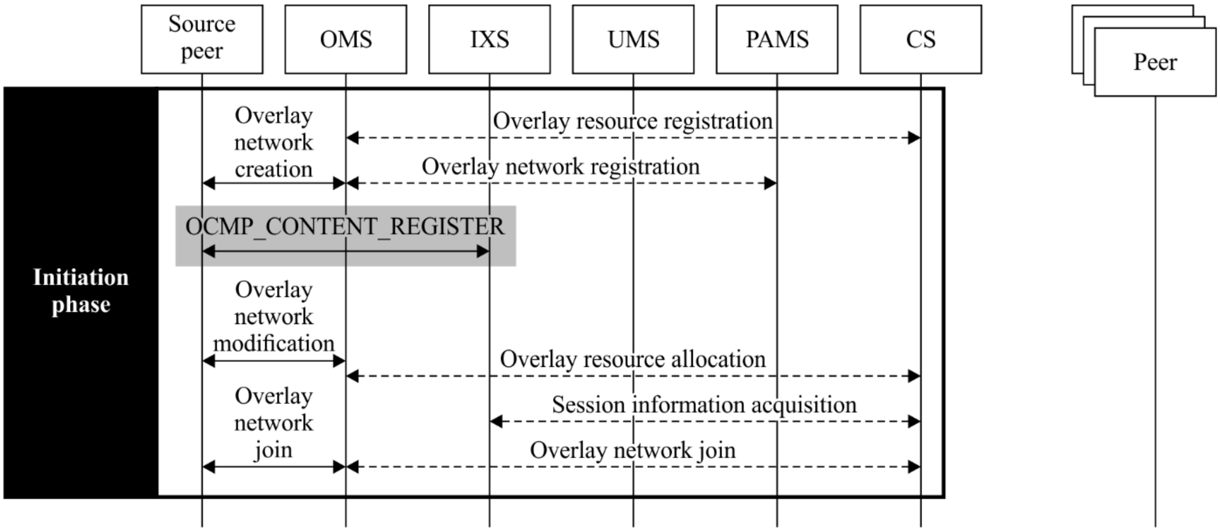


Figure 7-5 – Procedure for the content distribution service initiation phase

Figure 7-6 shows the procedure for the content distribution service distribution phase. To join a content distribution session, peers request IXS to send an index file which describes session information. The index file sent by the IXS shall be the latest version. The source peer sends IXS an OCMP_CONTENT_UPDATE request to modify the information of the registered session, when it needs. Peers can obtain the modified session information when it requests IXS to send the latest session information. In addition, peers can send IXS an OCMP_CONTENT_QUERY request to gather session information based on parameters such as title, content ID or owner ID.

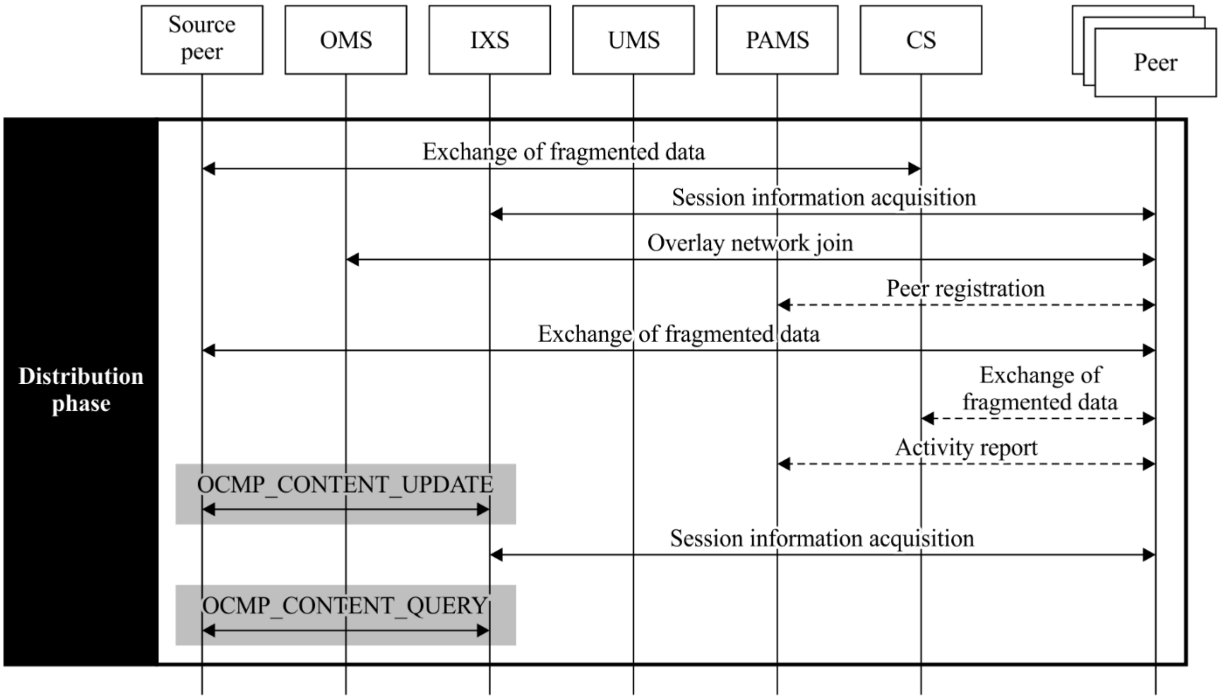
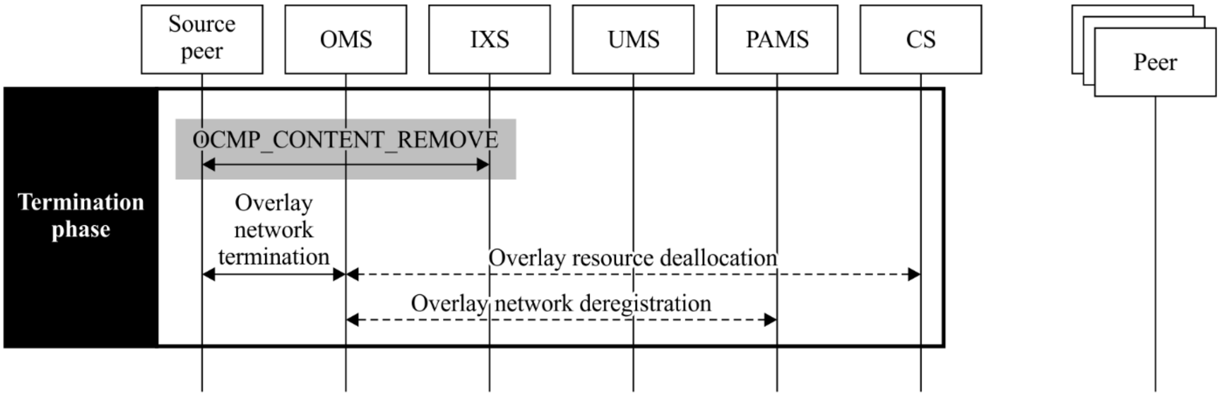


Figure 7-6 – Procedure for the content distribution service distribution phase

Figure 7-7 shows the procedure for the content distribution service termination phase. A source peer can request IXS to deregister a data streaming session by sending an OMCP_CONTENT_REMOVE request.


Source peer
OMS
IXS
UMS
PAMS
CS
Peer
Termination phase
OCMP_CONTENT_REMOVE
Overlay network termination
Overlay resource deallocation
Overlay network deregistration

X.609.9(20)_F7-7

Figure 7-7 – Procedure for the content distribution service termination phase

7.2.2 Multimedia streaming service

[ITU-T X.609.3] describes the requirements for reference point R6 (See Figure 6-1) for the multimedia streaming service, and this clause describes extended operations to support of the multimedia streaming service.

Table 7-2 shows in detail which of the parameters of OVERLAY_CONTENT is included for each OCMP message for the multimedia streaming service.

Table 7-2 – Parameters of OVERLAY_CONTENT for the multimedia streaming service

Element		REGISTER		UPDATE		REMOVE		QUERY	
OVERLAY_CONTENT		**req**	**rsp**	**req**	**rsp**	**req**	**rsp**	**req**	**rsp**
	version	–	m	m	m	m	–	–	m
	title	m	m	o	–	–	–	o	m
	content-id	–	m	m	–	m	–	o	m
	owner-id	m	–	m	–	m	–	o	m
	overlay-id	m	–	–	–	–	–	–	c1
	oms-address	m	–	–	–	–	–	–	c1
	piece-size	m	–	–	–	–	–	–	c1
	content-type	o	c2	–	–	–	–	–	c1
	piece-num	m	–	–	–	–	–	–	c1
	auth	m	–	m	–	m	–	–	c1
	files	–	–	–	–	–	–	–	–
	desc	o	–	o	–	–	–	o	o

- c1: When the OVERLAY_CONTENT_QUERY request message has one of title, content-id and owner-id in the parameter, it shall include the parameters. If not, it is optional;
- c2: When the OVERLAY_CONTENT_REGISTER request message has a 'content-type' parameter, the response includes this parameter. If not, it is optional;
- m: mandatory;
- o: optional.

When it comes to register a new multimedia streaming session, the owner of the session creates an index file containing the OVERLAY_CONTENT element and sends an OVERLAY_CONTENT_REGISTER request to the IXS. When the IXS registers the index file, it issues a new overlay content identifier for the content and embeds the identifier in the *content-id* parameter of the response. When registering a multimedia streaming session, the IXS specifies the content type within the 'content type' written in MIME format. However, it is assumed that the type of content is 'video/MP2T' if not specified.

On updating the index file, the owner sends an OVERLAY_CONTENT_UPDATE message containing the OVERLAY_CONTENT element as shown in Table 7-2. When the update procedure is successful, the IXS increments the *version* parameter before embedding the OVERLAY_CONTENT element in the response.

For the client peers that want to receive multimedia stream, the peer sends OVERLAY_ CONTENT_QUERY message with parameters regarding *title*, *content-id* or *overlay-id*. If the message does not contain those parameters, the IXS responds with the list of overlay contents including only the *title*, *content-id* and *overlay-id* for each index file. This is because it is preferable not to provide all the information of all overlay contents to avoid performance issues. On receiving the index file from the IXS, the peer creates directory structures by using the *files* parameter.

Figure 7-8 shows a procedure for the multimedia streaming service initiation phase. To register a multimedia streaming session, a source peer sends an OCMP_CONTENT_REGISTER request to the IXS. The detailed parameters of the message are listed in Table 7-2.

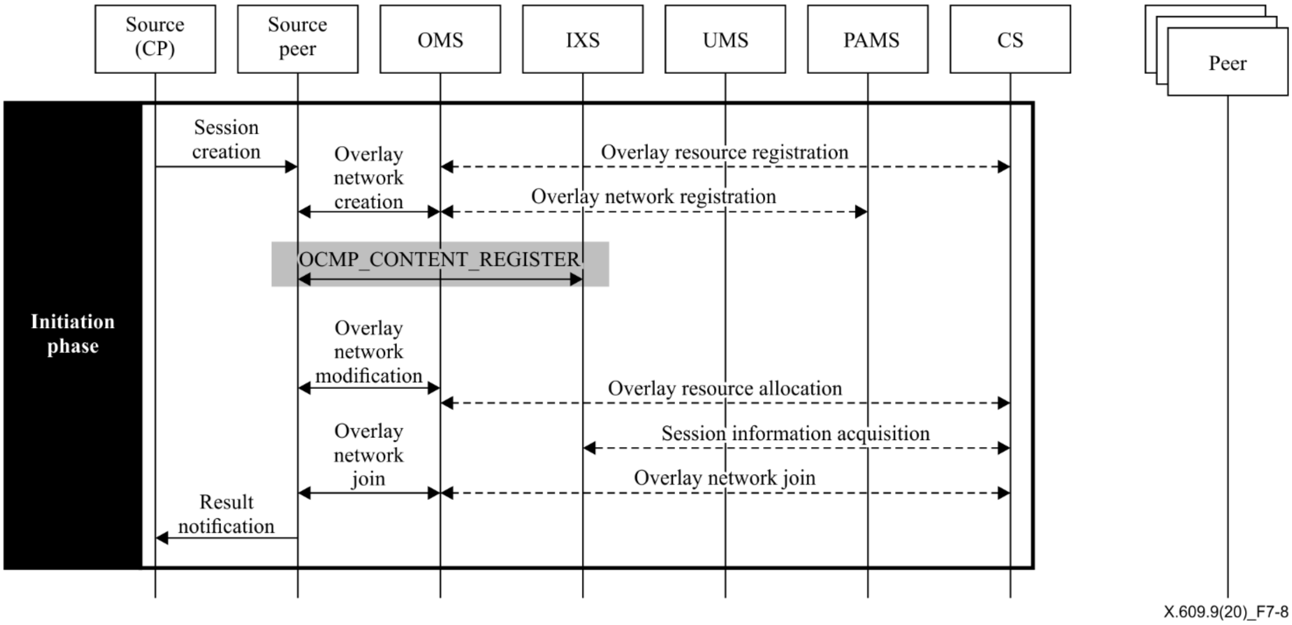


Figure 7-8 – Procedure for the multimedia streaming service initiation phase

Figure 7-9 shows the procedure for the multimedia streaming service distribution phase. To join a multimedia streaming session, peers request IXS to send an index file which describes session information. The index file sent by IXS shall be the latest version. The source peer sends IXS an OCMP_CONTENT_UPDATE request to modify the information of the registered session, when it needs. Peers can obtain the modified session information when it requests IXS to send the latest session information. In addition, peers can send IXS an OCMP_CONTENT_QUERY request to gather session information based on parameters such as title, content ID or owner ID.

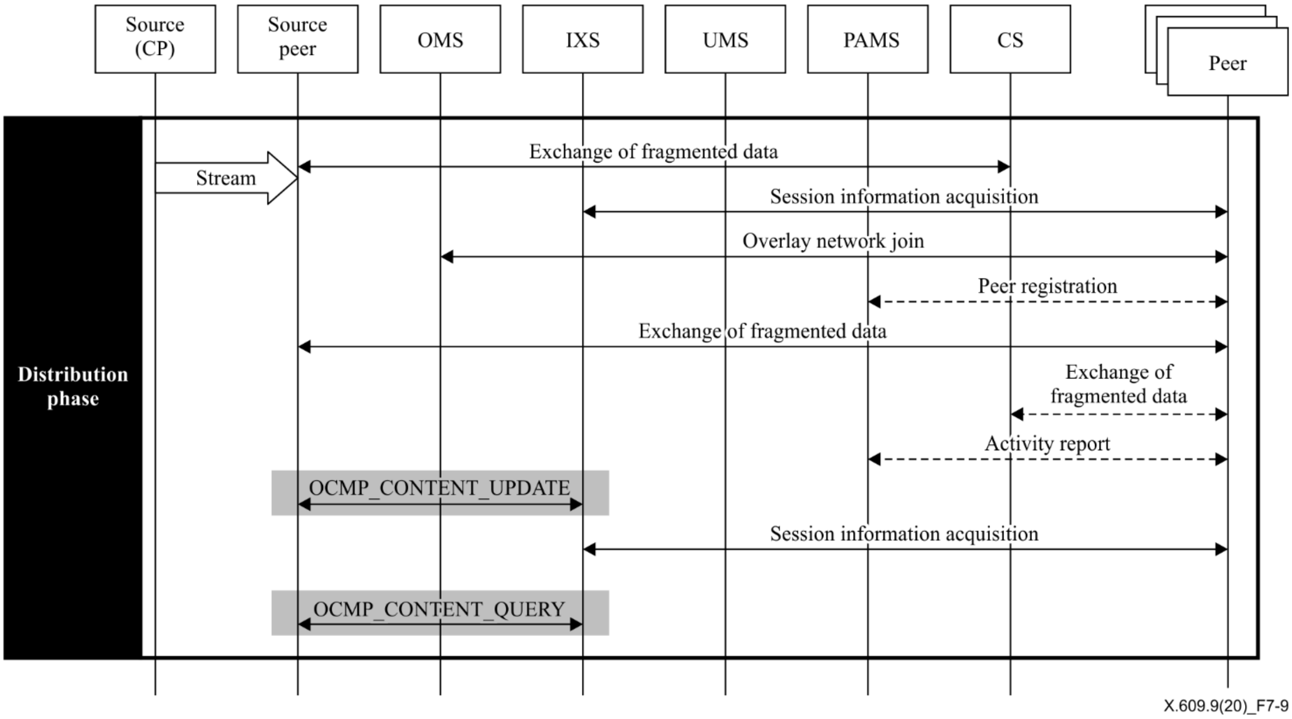


Figure 7-9 – Procedure for the multimedia streaming service distribution phase

Figure 7-10 shows the procedure for the multimedia streaming service termination phase. A source peer can request IXS to deregister a data streaming session by sending an OMCP_CONTENT_REMOVE request.

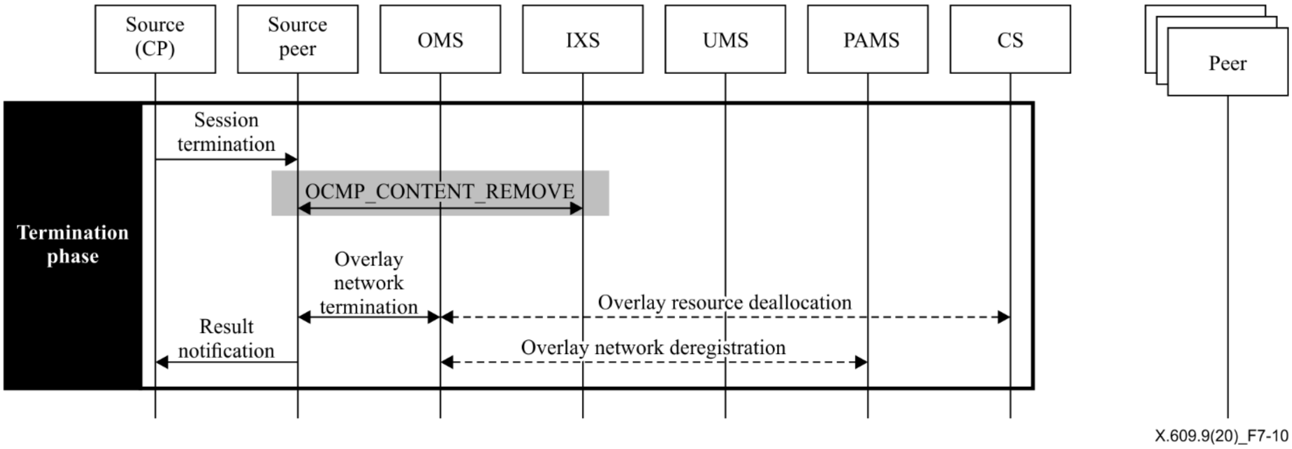


Figure 7-10 – Procedure for the multimedia streaming service termination phase

7.2.3 Data streaming service

[ITU-T X.609.10] describes the requirements for reference points R6 (see Figure 6-1) for the data streaming service, and this clause describes extended operations to support the data streaming service. Table 7-3 shows in detail which of the parameters of OVERLAY_CONTENT is included for each OCMP message for the data streaming service.

Table 7-3 – Parameters of OVERLAY_CONTENT for the data streaming service

Element		**REGISTER**		**UPDATE**		**REMOVE**		**QUERY**	
OVERLAY_CONTENT		**req**	**rsp**	**req**	**rsp**	**req**	**rsp**	**req**	**rsp**
	version	–	m	m	m	m	–	–	m
	title	m	m	o	–	–	–	o	m
	content-id	–	m	m	–	m	–	o	m
	owner-id	m	–	m	–	m	–	o	m
	overlay-id	m	–	–	–	–	–	–	c1
	oms-address	m	–	–	–	–	–	–	c1
	piece-size	m	–	–	–	–	–	–	c1
	content-type	–	–	–	–	–	–	–	–
	piece-num	m	–	–	–	–	–	–	c1
	auth	m	–	m	–	m	–	–	c1
	files	–	–	–	–	–	–	–	–
	desc	m	–	o	–	–	–	o	o

– c1: When an OVERLAY_CONTENT_QUERY request has one of title, content-id and owner-id, the marked attribute is required to be included. If not, it is optional;

– m: mandatory;

– o: optional.

For data streaming service over MP2P communications, a source peer interacts with IXS to register a data streaming session and with OMS to establish an overlay network corresponding to the registered session. Figure 7-11 shows a procedure for the data streaming service initiation phase. To register a data streaming session, a source peer sends an OCMP_CONTENT_REGISTER request to IXS.

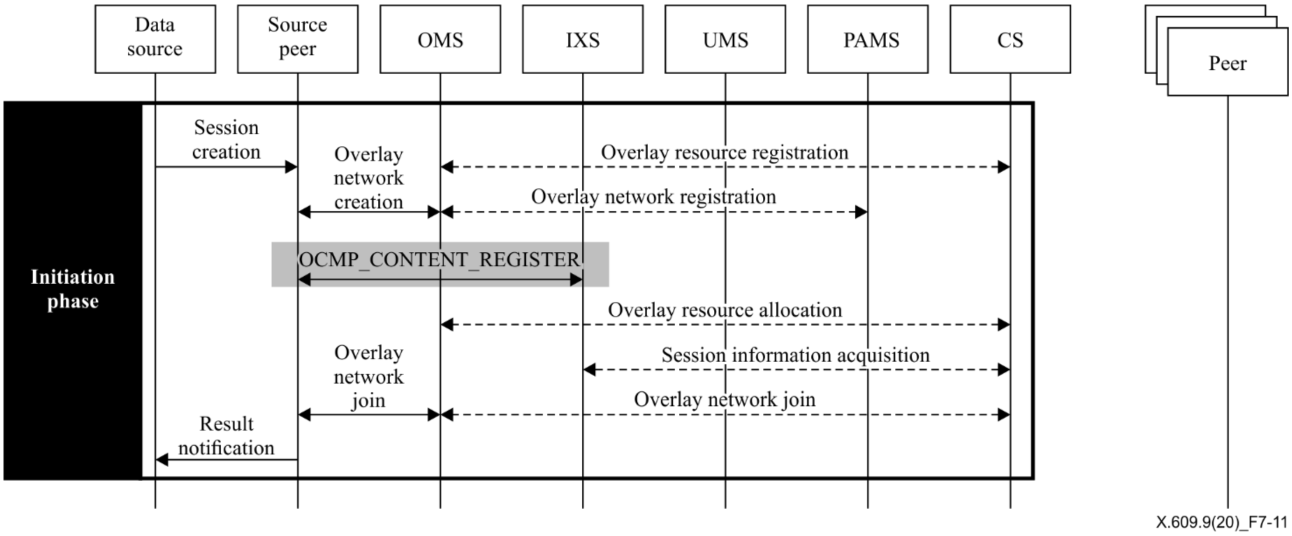


Figure 7-11 – Procedure for the data streaming service initiation phase

Figure 7-12 shows the procedure for the data streaming service distribution phase. To join a data streaming session, peers request IXS to send an index file which describes session information. The index file sent by IXS shall be the latest version. The source peer sends IXS an OCMP_CONTENT_UPDATE request to modify the information of the registered session, when it needs. Peers can obtain the modified session information when it requests IXS to send the latest session information. In addition, peers can send IXS an OCMP_CONTENT_QUERY request to gather session information based on parameters such as title, content ID or owner ID.

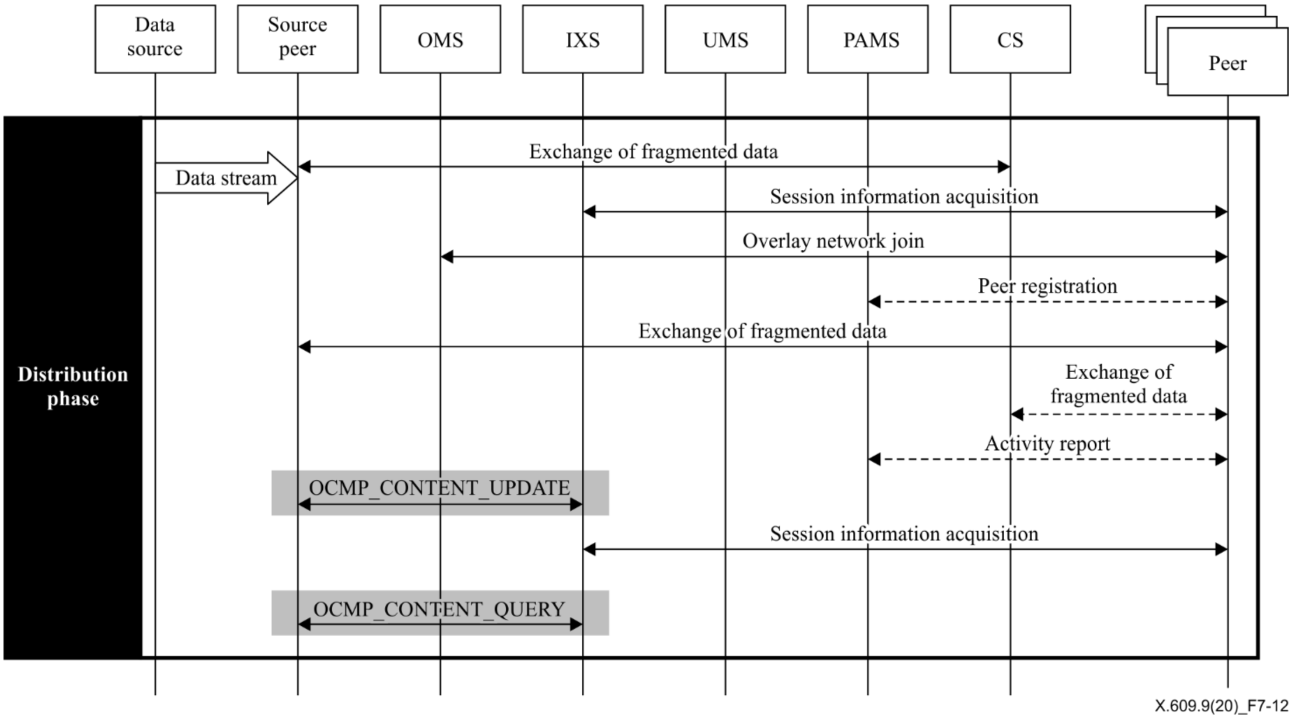


Figure 7-12 – Procedure for the data streaming service distribution phase

Figure 7-13 shows the procedure for the data streaming service termination phase. A source peer can request IXS to deregister a data streaming session by sending an OMCP_CONTENT_REMOVE request.

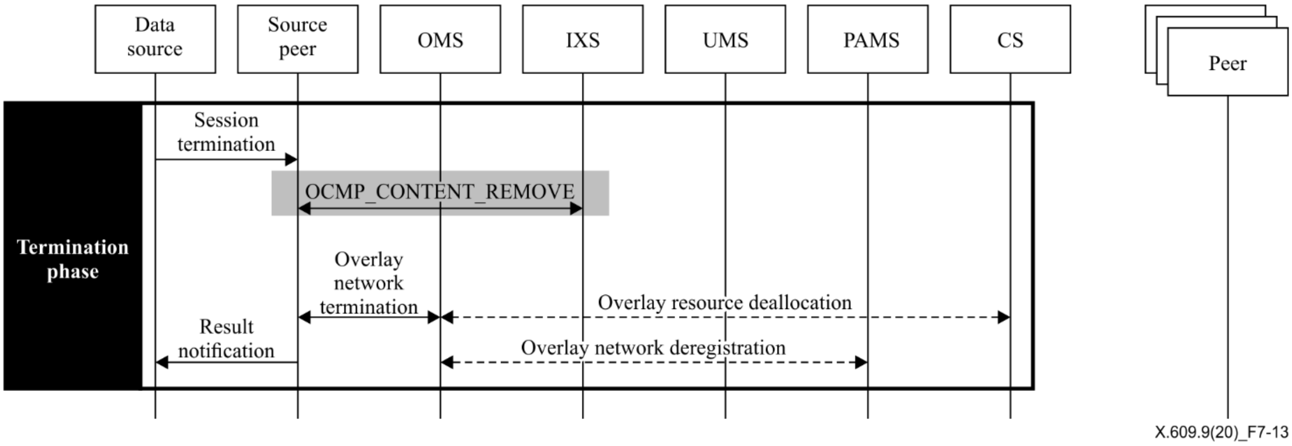


Figure 7-13 – Procedure for the data streaming service termination phase

8 Messages and parameters

This clause describes the format of OCMP messages. For extensibility, OCMP adopts the representational state transfer (REST) architecture, a style of software architecture for distributed systems. The most common encoding format used in REST is JSON and XML. This Recommendation uses the text type JSON [IETF RFC 7159] encoding.

It is possible to extend the elements or attributes to the resource element type to customize existing features and support new properties and capabilities for different operating environments. Extending the protocol implies adding new features without changing the protocol itself. The extension must not alter the existing protocol and must support backward capability.

8.1 Element types

This clause provides the format of element types used in this Recommendation. The grammar used in representing objects defined in this Recommendation is as follows:

– «STRING», «BOOLEAN», and «NUMBER» are types used to indicate string, Boolean, and number, respectively;

– An array of collective values are enclosed in brackets «[ ]» with values separated by a comma «,»;

– Selective options are separated by a vertical bar «|».

8.1.1 OVERLAY_CONTENT element

This OVERLAY_CONTENT element identifies the details of the content information. The generic definition of the element is as follows:

```
Object {
  NUMBER        version
  STRING        title
  STRING        content-id
  STRING        owner-id
  STRING        overlay-id
  STRING        oms-address
  NUMBER        fragment-size
  STRING        content-type {«file»|«stream»|other}
  NUMBER        fragment-num
  AUTH          auth
  FILES         files
  STRING        desc
  MSCHAR                multimedia-streaming-session-char
} OVERLAY_CONTENT
```

The description of the attributes is as follows:

– *version* indicates the version of the meta-information. If there is any change of the attributes of the content, the number is incremented by one;

– *title* indicates the title of the content;

– *content-id* indicates the identifier of the content. This is issued by the IXS on receiving a content information registration request from a source peer;

– *owner-id* indicates the identifier of the owner that has authority to control this content information such as creation, removal and update;

– *overlay-id* is the identifier of an overlay network;

- *oms-address* is the address of an overlay management server (OMS) that manages the overlay network for this overlay content;
- *piece-size* indicates the size of fragment in kibibytes (KiB). If this parameter is not present, it means that the size of a fragment is not fixed. However, if it is present on the registration of a content, this value is not changed;
- *content-type* indicates the type of content such as file, stream and other;
- *fragment-size* indicates the size of a fragment in KiB. If this parameter is not present, it means that the size of a fragment is not fixed. However, if it is present on the registration of a content, this value is not changed;
- *auth* includes the information for authentication on accessing a specific meta-information;
- *files* list up the files to be distributed. This parameter is used for the content distribution service;
- *desc* includes detailed descriptions on the contents to be distributed in human readable format;
- multimedia-*streaming-session-char* includes the information on characteristics of multimedia streaming session.

8.1.2 FILES element

The FILES element identifies the details of the contents to be distributed. The generic definition of the FILES element is as follows. In the case of multimedia streaming, this element is not used.

```
Object [{
  STRING        path
  STRING        filename
  NUMBER        file-size
  HASHES        hashes
},…] FILES;
```

The description of the attributes is as follows:

- *path* indicates a relative path of the file;
- *filename* indicates the name of the file;
- *file-size* indicates the size of the file in KiB;
- *hashes* indicate the hash value of all fragments of the file, and this is generated by a source peer before requesting a registration of a content.

8.1.3 HASHES element

The HASHES element includes hash values of the fragments to be distributed. The generic definition of the HASHES element is as follows:

```
Object [{
  STRING        fragment-id
  STRING        hash
},…] HASHES;
```

- *fragment-id* indicates the identifier of a particular fragment;
- *hash* contains the SHA-1 hash-value for the fragment.

8.1.4 AUTH element

The AUTH element identifies the type of authentication used by the overlay network. The generic definition of the AUTH element is as follows:

```
Object {
  STRING          closed = "YES"|"NO"|"KEY"
  STRING          key
} AUTH;
```

The description of the attributes is as follows:

– *closed* indicates whether the overlay network is closed, open or accessible with key;

– *key* indicates the key string used to join the overlay network. It is valid only if the attribute closed is set to the value «KEY».

8.1.5 MSCHAR element

This MSCHAR element identifies the details of a multimedia streaming session. The generic definition of the element is as follows:

```
Object {
  NUMBER          init-date
  NUMBER          init-time
  NUMBER          GMT
  NUMBER          keyFrame-interval-inSec
  NUMBER          keyFrame-interval-inFrame
  NUMBER          frameRate
  NUMBER          bitrate
  STRING          encoding {«CBR»|«VBR»|other}
} MSCHAR
```

The description of the attributes is as follows:

– *init-date* indicates the date in the form of yyyymmdd when the initial multimedia stream to be distributed was created;

– *init-time* indicates the time in the form of hhmmss when the initial multimedia stream to be distributed was created;

– *GMT* indicates Greenwich mean time of the location where the multimedia stream is created. The value can be set to a value between −12 and +12;

– *keyFrame-interval-inSec* indicates the keyframe interval in seconds. n means a keyframe will be created every n second;

– *keyFrame-interval-inFrame* indicates the keyframe interval in frame. n means a keyframe will be created every n frame;

– *frameRate* indicates the frame rate of the multimedia stream;

– *bitrate* indicates the bit rate of the multimedia stream. If encoding attribute is set to «VBR», bitrate indicates the maximum bit rate;

– *encoding* indicates the encoding method. The value can be constant bit rate (CBR) or variable bit rate (VBR) or other value.

8.2 Messages

This clause specifies the message formats and shows examples. This Recommendation has four primitives for controlling overlay content information. Each transaction consists of request and response, and this Recommendation uses hypertext transfer protocol (HTTP) messages for conveying the information.

8.2.1 OCMP_CONTENT_REGISTER

This primitive is used to register an overlay content. When a peer wants to register, it sends a request message with its meta-information describing the property of the overlay content. On successful registration, IXS returns a new identifier for the overlay contents.

8.2.1.1 Request

The request message format for this primitive is shown in Table 8-1.

Table 8-1 – Request message format for OCMP_CONTENT_REGISTER

Method	POST
URI	http://{IXS_ADDRESS} [a]/ixs
Body	OVERLAY_CONTENT (refer to clause 8.1.1)
[a] {IXS_ADDRESS} refers to the fully qualified domain name (FQDN) address of OMS	

An example HTTP request message for OCMP_CONTENT_REGISTER is as follows:

```
POST /ixs HTTP/1.1
Host: www.example_ixs.com
Content-Length: 528
Content-Type: application/json
Accept: application/json
{
   "version" : 1,
   "title" : "exampleContent",
   "owner-id" : "8djdhd",
   "overlay-id" : "12ekd4kd8",
   "oms-address" : "www.example_oms.com",
   "piece-size" : "2MB",
   "content-type" : "file",
   "piece-num" : 2,
   "files" : [
      {"path" : "",
      "file-size" : "4095234",
      "filename" : "exampeFile.dat",
      "hashes" : [
         {"fragment-id" : "0000000001",
         "hash" : "1w2e3r5t6y0ofksjh1=-"},
            {fragment-id" : "0000000002",
         "hash" : "0odlfj10d=-0978djh1`4"}]
   }]
}
```

8.2.1.2 Response

The response message format for this primitive is shown in Table 8-2.

Table 8-2 – Response code for OCMP_CONTENT_REGISTER

Response code and semantics		Body
200	OK The request is accepted and registration has succeeded.	version, content-id
400	Client Error The request has failed due to some reasons.	reasons of failure with plain text
401	Unauthorized The request requires user authentication.	n/a
500	Internal Server Error The request is denied due to some reason.	n/a
501	Not Implemented The request is denied when IXS does not support OCMP.	n/a

An example HTTP response message for OCMP_CONTENT_REGISTER is as follows:

```
HTTP/1.1 200 OK
Content-Length: 756
Content-Type: application/json
{
   "version": 1,
  "content-id": "855b623c-0cb5-40c1-8539-3fbb0fc1814a"
}
```

8.2.2 OCMP_CONTENT_UPDATE

This primitive is used to update an overlay content. When a source peer wants to update, it sends a request message with its meta-information describing the property of the modified overlay content. On sending the request message, the source peer increments the version number.

8.2.2.1 Request

The request message format for this primitive is shown in Table 8-3.

Table 8-3 – Request message format for OCMP_CONTENT_UPDATE

Method	PUT
URI	http://{IXS_ADDRESS} a)/ixs/{content-id}
Body	OVERLAY_CONTENT (refer to clause 8.1.1)
a) {IXS_ADDRESS} refers to the FQDN address of OMS	

An example HTTP request message for OCMP_CONTENT_UPDATE is as follows:

```
PUT /ixs/855b623c-0cb5-40c1-8539-3fbb0fc1814a HTTP/1.1
Host: www.example_ixs.com
Content-Length: 529
Content-Type: application/json
```

```
Accept: application/json
{
   "version" : 2,
   "title" : "exampleContent",
   "owner-id" : "8djdhd",
   "overlay-id" : "12ekd4kd8",
   "oms-address" : "www.example_oms.com",
   "piece-size" : "4MB",
   "content-type" : "file",
   "transport" : "tcp",
   "piece-num" : 2,
   "files" : [
       {"path" : "",
       "file-size" : "8191231",
       "filename" : "exampeFile2.dat",
       "hashes" : [
          {"fragment-id" : "0000000001",
          "hash" : "9w2eop5t6y0ofksj,/=-"},
             {fragment-id" : "0000000002",
          "hash" : "01flfj10d=-0978djh1ee"}]
   }]
}
```

8.2.2.2 Response

The response message format for this primitive is shown in Table 8-4.

Table 8-4 – Response code for OCMP_CONTENT_UPDATE

Response code and semantics		Body
200	OK The request is accepted and the update has succeeded.	version, content-id
400	Client Error The request has failed due to some reasons.	reasons of failure with plain text
401	Unauthorized The request requires user authentication.	n/a
500	Internal Server Error The request is denied due to some reason.	n/a
501	Not Implemented The request is denied when IXS does not support OCMP.	n/a

An example HTTP response message for OCMP_CONTENT_UPDATE is as follows:

```
HTTP/1.1 200 OK
Content-Length: 75
Content-Type: application/json
{
   "version": 2,
  "content-id": "855b623c-0cb5-40c1-8539-3fbb0fc1814a"
}
```

8.2.3 OCMP_CONTENT_REMOVE

This primitive is used to delete a specific overlay content registered in IXS. When a peer wants to delete an overlay content registered by the peer, it sends a request message with channel identifier.

8.2.3.1 Request

The request message format for this primitive is shown in Table 8-5.

Table 8-5 – Request message format for OCMP_CONTENT_REMOVE

Method	DELETE
URI	http://{IXS_ADDRESS} [a)]/ixs/{content-id}
Body	n/a

[a)] {IXS_ADDRESS} refers to the FQDN address of OMS

An example HTTP request message for OCMP_CONTENT_REMOVE is as follows:

```
DELETE /ixs/855b623c-0cb5-40c1-8539-3fbb0fc1814a HTTP/1.1
Host: www.example_ixs.com
Content-Length: 0
```

8.2.3.2 Response

The response message format for this primitive is shown in Table 8-6.

Table 8-6 – Response code for OCMP_CONTENT_REMOVE

Response code and semantics		Body
200	OK The request is accepted and the update has succeeded.	n/a
400	Client Error The request has failed due to some reasons.	reasons of failure with plain text
401	Unauthorized The request requires user authentication.	n/a
500	Internal Server Error The request is denied due to some reason.	n/a
501	Not Implemented The request is denied when IXS does not support OCMP.	n/a

An example HTTP response message for OCMP_CONTENT_REMOVE is as follows:

```
HTTP/1.1 200 OK
Content-Length: 0
```

8.2.4 OCMP_CONTENT_QUERY

This primitive is used for querying the properties of an overlay content. When a peer wants to query, it sends a request message with the content identifier.

8.2.4.1 Request

The request message format for this primitive is shown in Table 8-7.

Table 8-7 – Request message format for OCMP_CONTENT_QUERY

Method	PUT

URI	http://{IXS_ADDRESS} [a)]/ixs
	http://{IXS_ADDRESS} [a)]/ixs/content-id/{content-id}
	http://{IXS_ADDRESS} [a)]/ixs/title/{title}
	http://{IXS_ADDRESS} [a)]/ixs/owner-id/{owner-id}

Body	n/a
[a)] {IXS_ADDRESS} refers to the FQDN address of OMS	

The following OCMP_CONTENT_QUERY message shows an example of requesting the whole list of overlay contents on IXS.

```
GET /ixs HTTP/1.1
Host: www.example_ixs.com
Content-Length: 0
```

The following OCMP_CONTENT_QUERY message shows an example of querying the property of a specific overlay content.

```
GET /ixs/855b623c-0cb5-40c1-8539-3fbb0fc1814a HTTP/1.1
Host: www.example_ixs.com
Content-Length: 0
```

The following OCMP_CONTENT_QUERY message shows an example of querying the properties of overlay contents with a specific keyword in the title.

```
GET /ixs/title/peace HTTP/1.1
Host: www.example_ixs.com
Content-Length: 0
```

8.2.4.2 Response

The response message format for this primitive is shown in Table 8-8.

Table 8-8 – Response code for OCMP_CONTENT_QUERY

Response code and semantics		Body
200	OK The request is accepted and the update has succeeded.	OVERLAY_CONTENT
400	Client Error The request hasfailed due to some reasons.	reasons of failure with plain text
401	Unauthorized The request requires user authentication. CS/RS may repeat the request with suitable authorization in the HTTP header.	n/a
500	Internal Server Error The request is denied due to the following reason.	n/a
501	Not Implemented The request is denied when IXS does not support OCMP.	n/a

An example HTTP response message for OCMP_CONTENT_QUERY is as follows:

```
HTTP/1.1 200 OK
Content-Length: 75
Content-Type: application/json
{
   "version": 2,
  "content-id": "855b623c-0cb5-40c1-8539-3fbb0fc1814a"
}
```

Annex A

Use of the MSCHAR element for time-synchronized multimedia streaming

(This annex forms an integral part of this Recommendation.)

A.1 Overview

In multimedia streaming, time synchronization provides users with the multimedia stream in accordance with a time stamp. Strict synchronization may not be possible in multimedia streaming over overlay networks such as a managed peer-to-peer network. However, soft synchronization can be possible by providing characteristics of multimedia streaming session which can be provided by a signalling message defined in this Recommendation.

Figure A.1 shows a conceptual image of time-synchronized multimedia streaming over a managed peer-to-peer (MP2P) network. A source peer injects fragments into the MP2P network as soon as they are generated. Peers select the last injected fragment they can receive properly. If the peers select the same or a similar fragment, the multimedia stream to be played back by each peer will be synchronized.

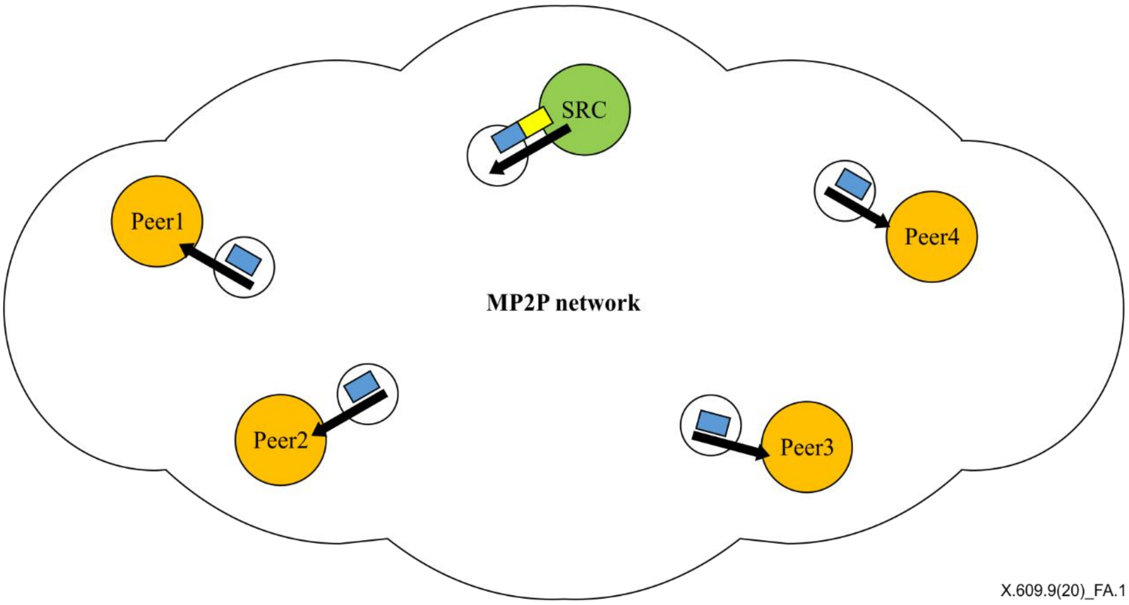


Figure A.1 – A conceptual image of time-synchronized multimedia streaming over MP2P network

To prevent the delay introduced by processing multimedia frames in each fragment, a source peer generates fragments which can be processed independently. Thus, a fragment contains a refresh frame, which is also known an instantaneous decode refresh (IDR)-frame and other frames referring the refresh frame. In addition, the size of each fragment depends on the encoding method. When an encoder uses constant bit rate (CBR), the size of all fragments will be same. However, each fragment may have a different size if an encoder uses variable bit rate (VBR), including constrained VBR (CVBR). Figures A.2 and A.3 show examples of fragment generation for two different cases.

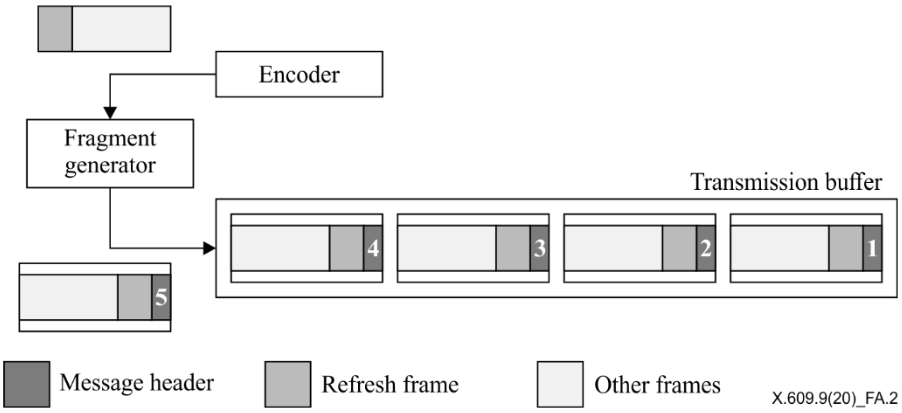


Figure A.2 – An example of fragment generation when the size of fragment is fixed

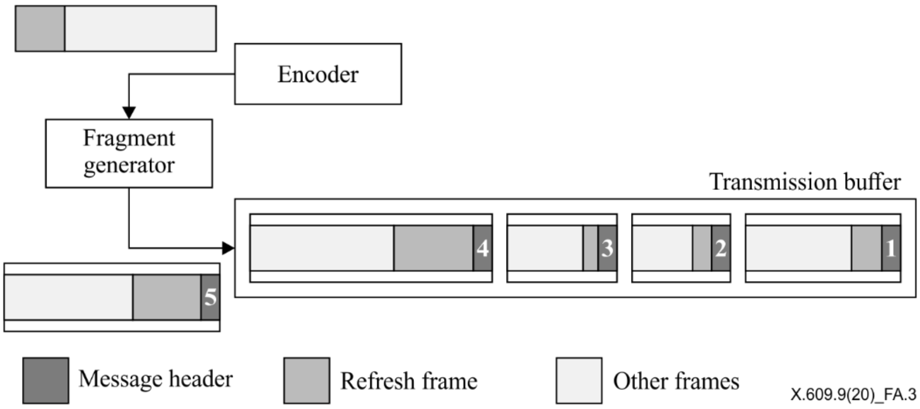


Figure A.3 – An example of fragment generation when the size of fragment is variable

A.2 Use of OCMP for time-synchronized multimedia streaming

To support time-synchronized multimedia streaming over MP2P networks, this Recommendation provides MSCHAR element for describing the characteristics of the multimedia streaming session. The element can be utilized by peers to determine an appropriate fragment to receive. Figure A.4 shows the procedures for selecting an appropriate fragment.

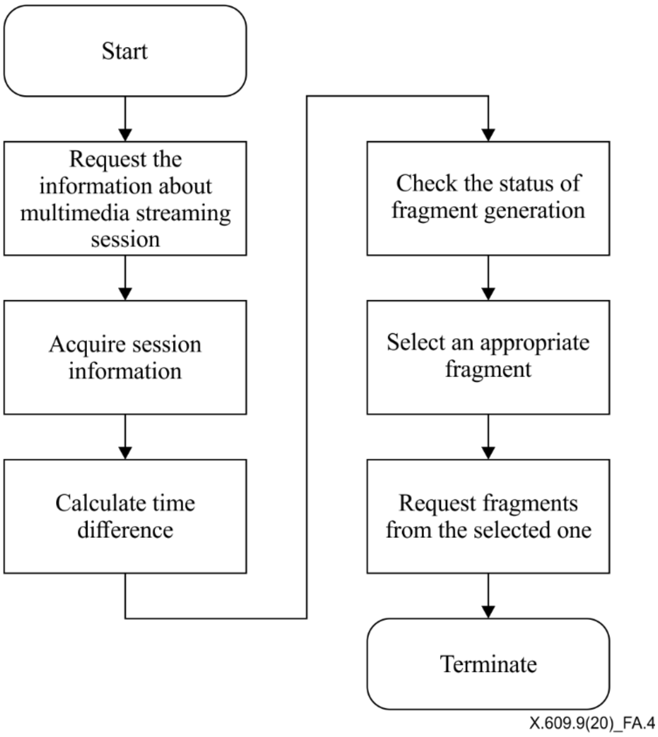

Start
Request the information about multimedia streaming session
Acquire session information
Calculate time difference
Check the status of fragment generation
Select an appropriate fragment
Request fragments from the selected one
Terminate
X.609.9(20)_FA.4

Figure A.4 – Procedures for selecting an appropriate fragment

To calculate the time difference, a peer uses information elements that make part of the session information. Especially, the peer compares the time when the initial stream was generated with its current time. Figure A.5 shows an overview of time difference calculation. A seed registers its streaming session by sending a request including the information on its streaming session. Then the index server (IXS) registers a new streaming session. Upon receiving a request from a peer, IXS sends session information to the peer. Then the peer can calculate the time difference by referring to the «init-date», «init-time» and «GMT» attributes in the session information. As an example, in Figure A.5, init-date is set to «20190101» and init-time is set to «135519». If the peer calculates time difference at 13:55:28 January first 2019, the calculation result will be 9 seconds. Note that the example assumes seed and peer are in same GMT.

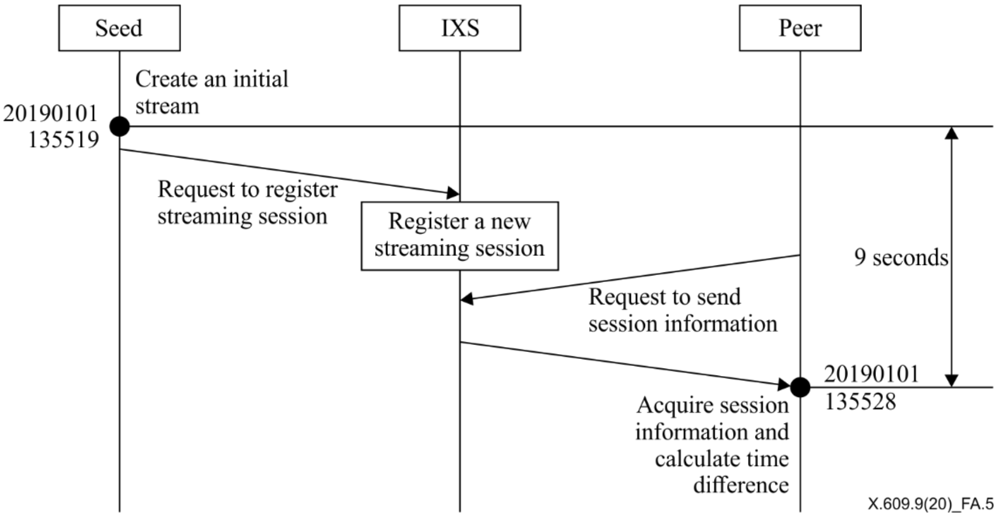

Seed
IXS
Peer
Create an initial stream
20190101
135519
Request to register streaming session
Register a new streaming session
9 seconds
Request to send session information
20190101
135528
Acquire session information and calculate time difference
X.609.9(20)_FA.5

Figure A.5 – Overview of time difference calculation

After calculating the time difference, a peer checks the status of fragment generation by referring to the session information. If the session information has *keyFrame-interval-inSec* and the value is *n*, the peer directly knows that a fragment is generated every *n* seconds. If the session information has *keyFrame-interval-inFrame* and *frameRate*, then the peer calculates the fragment interval. The interval can be derived by dividing *framerate* by *keyFrame-interval-inFrame*. For example, the interval is 2 seconds if *frameRate* is 60 and *keyFrame-interval-inFrame* is 30. After calculating the time difference and checking the status of fragment generation, a peer knows the total amount of fragments. In other words, the peer knows the fragment number of the latest fragment.

Then the peer selects a fragment to receive. To select an appropriate fragment as an initial point of multimedia streaming, the peer checks whether it can receive fragments after selecting a specific fragment. This Recommendation provides three different parameters for selecting the appropriate fragment; the size of a fragment, the period for generating a fragment and bit rate of the multimedia stream. For example, the size of a fragment may be used to calculate the expected delivery time of a fragment. By using the size of a fragment and available downlink capacity of a peer, the peer can calculate the expected delivery time. The available downlink capacity is already known by peer, but the peer needs to refer to session characteristics for acquiring the size of a fragment. If the *encoding* attribute is set to «CBR», then the peer can refer to the *piece-size* attribute directly. However, the *piece-size* attribute will not exist if the *encoding* attribute is set to «VBR». Then the peer estimates the size of a fragment by referring to the *bitrate* attribute and fragment generation period. How to calculate the fragment generation period is already described in the previous paragraph. The estimated fragment size may not be exact, but it can be used to estimate the expected delivery time of a fragment. Note that fragment #n+1 needs to arrive at least when fragment #n is being played back. Thus, the expected delivery time of a fragment needs to be smaller than the play back time of a fragment. The time for processing a received fragment also needs to be considered as shown in Figure A-6. As another example, the peer can compare its available downlink capacity and the bit rate of the multimedia stream. If its available downlink capacity is higher than the bit rate of the multimedia stream, the peer may request the latest fragment.

NOTE – This annex does not provide any specific method for selecting the appropriate fragment, because that depends on the implementation.

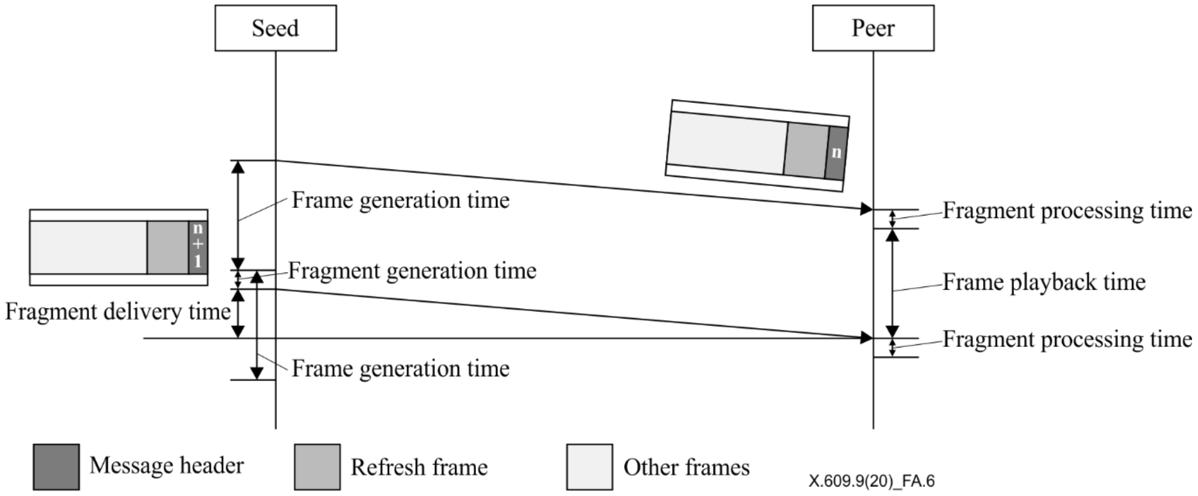


Figure A.6 – An example of time difference calculation

Bibliography

[b-ITU-T X.1161]	Recommendation ITU-T X.1161 (2008), *Framework for secure peer-to-peer communications*.
[b-ITU-T X.1162]	Recommendation ITU-T X.1162 (2008), *Security architecture and operations for peer-to-peer networks*.
[b-ITU-T Y.2206]	Recommendation ITU-T Y.2206 (2010), *Requirements for distributed service networking capabilities*.
[b-ISO/IEC TR 20002]	ISO/IEC TR 20002 (2012), *Information technology – Telecommunications and information exchange between systems – Managed P2P: Framework.* https://www.iso.org/standard/50950.html

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2020