
Source: SA5 (Telecom Management)
Title: Rel-5 draft TS 32.225 v150 (Charging data description for the IP Multimedia Subsystem) - for the 2nd time for Information
Document for: Information
Agenda Item: 7.5.3

Presentation of Technical Specification to TSG SA

Presentation to: TSG SA Meeting #16
Document for presentation: TS 32.225, Version 1.5.0
Presented for: the 2nd time to SA for Information
Abstract of document: This is a draft Technical Specification on Charging Management; Charging Data Description for IP Multimedia Subsystem;
Changes since last presentation to TSG-SA Meeting #15: Revised Draft adding e.g. on-line charging, Diameter protocol
Version 1.0.0 was presented for Information at SA#15 (03/2002)

Work done against the WID contained in SP-010654 (Work Item ID: OAM-CH).

Outstanding Issues:

1. This draft TS represents the status of the draft specification as of the end of SA5#28 (05/2002). It contains almost all the text for both off-line and on-line charging that needs to be included for stage 3 implementations and there are no action items regarding missing clauses.
2. In the parts of the documents that were reviewed and agreed upon by the entire SWGB all track marks have been accepted.
3. However, due to the lack of time, some parts of the document were not reviewed by the entire SWGB, but only by the contributing authors. This is indicated by clauses that still contain track marks.
4. At present, most of the basic principles, the use of the Diameter protocol, on-line and off-line flows and (off-line) CDR content have all been reviewed and agreed.
5. The message formats and the AVPs still need to be reviewed. A sections containing the CDR format is still waiting agreement (received but not yet broached). Alignments with 32.200 (Charging Principles) and few editorial notes still need to be checked and completed. The description of the CDR parameters and the mapping between the DIAMETER AVPs and the CDR parameters still need to be specified.

Contentious Issues: None

NOTE: SA5's target is to submit draft v200 to SA#17 (09-2002) for Approval and placement under TSG Change Control (CR regime); i.e. rising to v500.

3GPP TS 32.225 V1.5.0 (2002-05)

Technical Specification

**3rd Generation Partnership Project;
Technical Specification Group;
Telecommunication Management;
Charging Management;
Charging Data Description for IP Multimedia Subsystem;
(Release 5)**



The present document has been developed within the 3rd Generation Partnership Project (3GPP™) and may be further elaborated for the purposes of 3GPP.

The present document has not been subject to any approval process by the 3GPP Organizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPP only. The Organizational Partners accept no liability for any use of this Specification. Specifications and reports for implementation of the 3GPP™ system should be obtained via the 3GPP Organizational Partners' Publications Offices.

Keywords

MMS, charging

3GPP

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet

<http://www.3gpp.org>

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© 2002, 3GPP Organizational Partners (ARIB, CWTS, ETSI, T1, TTA, TTC).
All rights reserved.

Contents

Foreword.....	8
1 Scope	9
2 References	9
3 Definitions and abbreviations	10
3.1 Definitions.....	10
3.2 Symbols.....	10
3.3 Abbreviations	10
4 Offline and Online Charging	10
4.1 Implementation of Offline and Online Charging	10
4.1.1 Usage of Rf or Ro Interfaces	10
4.1.2 Usage of Rf or ISC Interfaces.....	11
4.2 DIAMETER protocol basic principles and use	11
4.2.1 Basic Principles	11
4.2.2 Application Requirement for the Base Protocol.....	11
4.2.2.1 Offline Specific Base Protocol Requirements	11
4.2.2.2 Online Specific Base Protocol Requirements	11
5 Offline Charging.....	12
5.1 Diameter Description on the Rf Interfaces.....	12
5.1.1 Basic Principles	12
5.1.2 Message Flows and Types.....	13
5.1.2.1 Message Flows - Successful Cases and Scenarios.....	13
5.1.2.1.1 Session Related Procedures	13
5.1.2.1.1.1 Session Establishment – Mobile Origination.....	13
5.1.2.1.1.2 Session Establishment – Mobile Termination.....	14
5.1.2.1.1.3 Mid-Session Procedures.....	15
5.1.2.1.1.4 Session Release – Mobile Initiated	16
5.1.2.1.1.5 Session Release – Network Initiated.....	17
5.1.2.1.1.6 Session Release – CCF initiated	17
5.1.2.1.2 Session-Unrelated Procedures.....	18
5.1.2.1.3 MRFC Related Procedures.....	19
5.1.2.1.3.1 Multi-Party Call	19
5.1.2.1.4 AS Related Procedures	21
5.1.2.1.4.1 AS as Redirect Server	21
5.1.2.2 Message Flows - Error Cases and Scenarios	22
5.1.2.2.1 Reception of SIP error messages.....	22
5.1.2.2.2 CCF connection failure	22
5.1.2.2.3 Duplicate detection	22
5.1.2.2.4 CCF detected failure	22
5.1.2.2.5 Error case – Registration not allowed	23
5.1.3 Message Formats.....	23
5.1.3.1 Summary of Offline Charging Message Formats	23
5.1.3.2 Structure for the Accounting-Request and Accounting-Answer Message Formats	24
• Accounting-Request Message	25
• Accounting-Answer Message	25
• Abort-Session-Request.....	26
• Abort-Session-Answer	26
5.1.3.3 Detailed S-CSCF Message Formats	27
• Start of Session.....	27
• Mid-Session Information.....	28
• End of Session.....	28
• Events	28
5.1.3.4 Detailed P-CSCF Message Formats	29
• Start of Session.....	29

• Mid-Session Information.....	30
• End of Session.....	30
• Events	31
5.1.3.5 Detailed I-CSCF Message Formats	32
• Start of Terminating Session Establishment.....	32
• End of Terminating Session Establishment.....	32
• Events	33
5.1.3.6 Detailed MRFC Message Formats.....	33
• Start of Session.....	34
• Mid-Session Information.....	34
• End of Session.....	35
• Events	35
5.1.3.7 Detailed MGCF Message Formats	36
• Start of Session.....	36
• Mid-Session Information.....	36
• End of Session.....	37
• Events	37
5.1.3.8 Detailed BGCF Message Formats	38
• Start of Session, Start of Media Component	38
• Mid-Session Information.....	39
• End of Session.....	39
• Events	40
5.1.3.9 Detailed AS Message Formats	40
• Start of Session.....	41
• Mid-Session Information.....	41
• End of Session.....	41
• Events	42
5.2 CDR Description on the Bi Interface	43
5.2.1 CDR Field Types.....	43
5.2.2 CDR Triggers	44
5.2.3 CDR Content	45
5.2.3.1 Charging Data in S-CSCF (S-CSCF-CDR)	45
5.2.3.2 Charging Data in P-CSCF (P-CSCF-CDR)	46
5.2.3.3 Charging Data in I-CSCF (I-CSCF-CDR).....	48
5.2.3.4 Charging Data in MRFC (MRFC-CDR)	49
5.2.3.5 Charging Data in MGCF (MGCF-CDR).....	50
5.2.3.6 Charging Data in BGCF (BGCF-CDR).....	52
5.2.3.7 Charging Data in AS (AS-CDR)	53
5.2.4 CDR Parameter Description	55
5.2.4.1 Application provided Called Party Address	55
5.2.4.2 Authorised QoS	55
5.2.4.3 BearerService	55
5.2.4.4 Called Party Address (Public User ID).....	55
5.2.4.5 Calling Party Address (Public User ID)	55
5.2.4.6 Cause (for Record Closure)	55
5.2.4.7 Correlation vector access part.....	55
5.2.4.8 Correlation Vector IMS part (ICID).....	55
5.2.4.9 Event Type	55
5.2.4.10 Inter Operator Identifier(s).....	55
5.2.4.11 List of Application Servers involved	55
5.2.4.12 List of Media Components.....	55
5.2.4.13 List of Media Components.....	55
5.2.4.14 Local Sequence Number	56
5.2.4.15 Media Component Initiator	56
5.2.4.16 Media Type	56
5.2.4.17 Private User ID (served party)	56
5.2.4.18 Node Address.....	56
5.2.4.19 Record Closure Time	56
5.2.4.20 Record Extensions	56
5.2.4.21 Record Opening Time.....	56

5.2.4.22	Record Output Reason	56
5.2.4.23	Record Sequence Number.....	56
5.2.4.24	Record Type.....	56
5.2.4.25	Role of Node.....	56
5.2.4.26	Session ID.....	56
5.2.4.27	Served party IP Address.....	56
5.2.4.28	Start of a Media component	57
5.2.4.29	Stop of a Media component	57
5.2.4.30	TrunkGroup ID Incoming/Outgoing.....	57
5.2.5	Bi interface Conventions	57
5.2.6	Abstract Syntax Description (tbd).....	57
5.2.7	Data Encoding Rules	57
6	Online Charging	57
6.1	Diameter Description on the Ro Interface.....	57
6.1.1	Basic Principles	57
6.1.2	Message Flows and Types.....	57
6.1.2.1	Immediate Event Charging.....	57
6.1.2.1.1	Message Flows - Successful Cases and Scenarios	58
6.1.2.1.1.1	Event - Direct Debiting	58
6.1.2.1.1.2	Event - Direct Debiting for Session-Unrelated Procedures.....	59
6.1.2.1.1.3	Transactions for Refund of the User's Account	60
6.1.2.1.1.4	Transactions for Service Price Enquiry.....	60
6.1.2.1.1.5	Transactions for Balance Check	61
6.1.2.1.2	Message Flows - Error Cases and Scenarios.....	62
6.1.2.1.2.1	Reception of SIP Error Messages	62
6.1.2.1.2.2	ECF Connection Failure	62
6.1.2.1.2.3	ECF Detected Failure.....	63
6.1.2.1.2.4	Transient, Permanent Failures and Protocol Errors	63
6.1.2.1.2.5	Duplicate Detection	63
6.1.2.1.2.6	Error Case – Failure in Termination Procedure	64
6.1.2.1.2.7	Error Case – Failure in Origination Procedure.....	65
6.1.2.1.2.8	Error Case – Credit Expired/Exhausted or User Unknown.....	67
6.1.2.2	Event Charging with Reservation	68
6.1.2.2.1	Message Flows - Successful Cases and Scenarios	68
6.1.2.2.1.1	Session Related Service – Credit Control and Reservation	68
6.1.2.2.1.2	Session-Unrelated Service – Credit Check And Reservation	69
6.1.2.2.1.3	Mid-Session Procedure – Intermediate Credit Check And Account Deduction	70
6.1.2.2.1.4	Service Release – ECF initiated.....	71
6.1.2.2.1.5	Stopping of Service Delivery – Mobile Initiated	72
6.1.2.2.1.6	AS/MRFC-Originated Multimedia Service	73
6.1.2.2.1.7	Transactions for Content Charging	74
6.1.2.2.2	Message Flows - Error Cases and Scenarios.....	76
6.1.2.2.2.1	Reception Of SIP Error Messages.....	76
6.1.2.2.2.2	ECF Connection Failure	76
6.1.2.2.2.3	ECF Detected Failure.....	77
6.1.2.2.2.4	Transient, Permanent Failures and Protocol Errors	77
6.1.2.2.2.5	Duplicate Detection	77
6.1.2.2.2.6	Error Case – Failure in Termination Procedure	77
6.1.2.2.2.7	Error Case – Failure in Origination Procedure.....	79
6.1.2.2.2.8	Error Case – Credit Expired/Exhausted or User Unknown.....	80
6.1.3	Message Formats.....	81
6.1.3.1	Summary of Online Charging Message Formats.....	81
6.1.3.2	Common Structure for the Accounting-Request and Accounting-Answer Message Formats.....	82
•	Accounting-Request Message	82
•	Accounting-Answer Message	83
7	Used AVPs and their Definitions for Offline and Online.....	84
7.1	Diameter Base Protocol AVPs (Offline and Online).....	84
7.1.1	Accounting-Interim-Interval.....	84
7.1.2	Accounting-Realtime-Required.....	84
7.1.3	Accounting-Record-Number	85

7.1.4	Accounting-Record-Type	85
7.1.5	Destination-Realm	86
7.1.6	Origin-Host	86
7.1.7	Origin-Realm	86
7.1.8	Origin-State-Id	86
7.1.9	Proxy-Info	87
7.1.10	Route-Record	87
7.1.11	Session-Id	87
7.1.12	User-Name	87
7.1.13	Vendor-Specific-Application-Id	88
7.2	Offline Charging AVPs	88
7.2.1	Re-Transmission	89
7.2.2	(IMS) Event Type	89
7.2.3	(User) Session ID	89
7.2.4	Application provided Called Party Address	89
7.2.5	Authorised QoS	89
7.2.6	Bearer Service	90
7.2.7	Called Party Address (Public User ID)	90
7.2.8	Calling Party Address (Public User ID)	90
7.2.9	Cause	90
7.2.10	Correlation Vector IMS part (ICID)	90
7.2.11	Inter Operator Identifier	90
7.2.12	List of Application Servers involved	90
7.2.13	Media-Action	90
7.2.14	Media Component	91
7.2.15	Media-Id	91
7.2.16	Media-Initiator	91
7.2.17	Media-Mode	91
7.2.18	Media-Type	91
7.2.20	Private User ID (served party)	92
7.2.21	Role of node	92
7.2.22	S-CSCF information	92
7.2.23	Served party IP Address	92
7.2.24	Service ID	92
7.2.25	SIP-Method	92
7.2.26	Time stamp	93
7.2.27	Start-Time-Stamp	93
7.2.28	Stop-Time-Stamp	93
7.2.29	TrunkGroup-ID-Incoming/Outgoing	93
7.3	Online Charging Specific AVPs	93
7.3.1	Abnormal-Termination-Reason AVP	94
7.3.2	Check-Balance-Result AVP	94
7.3.3	Cost AVP	94
7.3.4	Cost-Information AVP	94
7.3.5	Credit-Control-Failure-Handling AVP	95
7.3.6	Currency-Code AVP	95
7.3.7	Direct-Debiting-Failure-Handling AVP	95
7.3.8	Final-Unit-Indication AVP	96
7.3.9	Granted-Service-Unit AVP	96
7.3.10	Requested-Action AVP	96
7.3.11	Requested-Service-Unit AVP	97
7.3.12	Service-Parameter-Info AVP	97
7.3.13	Service-Parameter-Type AVP	97
7.3.14	Service-Parameter-Value AVP	97
7.3.15	Unit-Type AVP	98
7.3.16	Unit-Value AVP	98
7.3.17	Used-Service-Unit AVP	98
7.3.18	Result-Code AVP values	99
Annex A (informative): Offline Charging for an Event – Mobile Origination with one-time event accounting data transfer		99

Annex B (informative): Change history 101

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

This document covers both online and offline charging for the IMS.

For clarity, the terms Offline Charging and Online charging as applied to the IMS are defined here in Section 3. These definitions are the same as listed in TS 32.200 [2].

The IMS charging architecture details, requirements, definitions and principles are listed in TS 32.200 [2] and therefore are not repeated here.

In this document the charging data triggers, content and format are specified along and their transport using the DIAMETER protocol. Details about charging message flows and the definitions of the DIAMETER AVPs are included in this document. This information is divided into two main sections: Online Charging and Offline Charging.

2 References

The following documents contain provisions, which through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

- [1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".
- [2] 3GPP TS 32.200: "Telecommunication Management; Charging Management; Charging Principles".
- [3] IETF Internet-Draft Diameter Base Protocol, "Diameter Base Protocol "
<http://search.ietf.org/internet-drafts/draft-ietf-aaa-diameter-10.txt>
- [4] 3GPP TS 33.201: "Access domain security"
- [5] 3GPP TR 23.218: "IP Multimedia (IM) session handling; IM call model; Stage 2"
- [6] IETF RFC2486: "The Network Access Identifier"
- [7] IETF RFC 1890: "RTP Profile for Audio and Video Conferences with Minimal Control",
- [8] 3GPP TS 23.207: "End to end quality of service concept and architecture"
- [9] 3GPP TS 29.207: "Policy control over Gs interface"
- [10] ITU-T X.690 "Information Technology – ASN.1 Encoding Rules: Basic Encoding Rules (BER)"
- [11] ITU-T X.691 "Information Technology – ASN.1 Encoding Rules: Packet Encoding Rules (PER)"
- [12] ITU-T X.693 "Information Technology – ASN.1 Encoding Rules: XML encoding Rules (XER)"
- [13] 3GPP TS 24.228: "Signalling flows for the IP multimedia call control based on SIP and SDP; Stage 3"
- [14] IETF Internet-Draft Diameter Credit Control Application
<draft-hakala-Diameter-Credit-Control-02.txt>

3 Definitions and abbreviations

3.1 Definitions

Off-line charging is a charging mechanism where charging information **does not** affect, in real-time, the service rendered.

On-line charging is a charging mechanism where charging information can affect, in real-time, the service rendered and therefore a direct interaction is required with session/service control.

3.2 Symbols

For the purposes of the present document the following symbols apply:

Bi	The Interface between the IMS charging function and the BS
Rb	On-line Charging Reference Point between Session Charging Function and Correlation Function
Rc	On-line Charging Reference Point between ECF and Correlation Function
Re	On-line Charging Reference Point towards a Rating Server
Rf	Off-line Charging Reference Point between an IMS Network Entity or an AS and CCF
Ro	On-line Charging Reference Point between an AS or MRFC and the ECF

3.3 Abbreviations

For the purposes of the present document, the abbreviations defined in 3GPP TR 21.905 [1], 3GPP TS 32.200 [2] and the following apply:

ACA	Accounting Answer
ACR	Accounting Request
AS	Application Server
AVP	Attribute Value Pair
BGCF	Breakout Gateway Control Function
BS	Billing System
CCF	Charging Collection Function
CDR	Charging Data Record
CPCF	Content Provider Charging Function
ECF	Event Charging Function
CSCF	Call Session Control Function (I-Interrogating; P-Proxy; and S-Serving)
IMS	IP Multimedia Subsystem
ISC	IMS Service Control
MGCF	Media Gateway Control Function
MRFC	Media Resource Function Controller
OCS	Online Charging System
SCCF	Subscriber Content Charging Function
SIP	Session Initiation Protocol

4 Offline and Online Charging

4.1 Implementation of Offline and Online Charging

4.1.1 Usage of Rf or Ro Interfaces

The AS and MRFC are able to distinguish whether to apply offline or online charging, i.e., whether to send charging information on the Rf interface to the CCF or on the Ro interface to the ECF. This is decided based on the information (CCF and/or ECF address) the AS and MRFC receive in the SIP signalling. If the AS and MRFC only receive the CCF

address and do not receive an ECF address then they use only the Rf interface. If only the ECF address was provided then they use only the Ro interface. In cases where both CCF and ECF addresses are provided it is possible to use both interfaces simultaneously. The decision on which interface(s) to use is left for the operator configuration.

4.1.2 Usage of Rf or ISC Interfaces

All other IMS nodes (S-CSCF, P-CSCF, I-CSCF, BGCF and MGCF) always apply offline charging via the Rf interface using the CCF address as received via SIP signaling. The S-CSCF additionally supports online charging using the ISC interface, i.e. if the application server addressed over ISC is the Session Charging Function of the OCS.

4.2 DIAMETER protocol basic principles and use

This specification defines a 3GPP IMS charging Diameter application, which utilizes the Diameter Base Protocol [3]. This application is used for both online and offline charging. The generic description of the protocol is provided in the section below while the portions of the protocol application associated with offline and online charging are described in Section 5 and 6, respectively.

4.2.1 Basic Principles

The IMS charging Diameter application is based on the following general principles:

- The basic functionality of Diameter, as defined by the DIAMETER Base Protocol [3] is re-used in IMS.
- In case of offline charging, IMS network elements report accounting information to the Charging Collection Function (CCF). The CCF uses this information to construct and format CDRs.
- In case of online charging, IMS network elements report accounting information to the Event Charging Function (ECF). The ECF uses this information to support event based charging (content charging) function of the OCS.

4.2.2 Application Requirement for the Base Protocol

4.2.2.1 Offline Specific Base Protocol Requirements

In order to support the offline charging principles described in this document, the Diameter client and server must implement at least the following Diameter options listed in Ref. [3]:

- To send/receive *Abort-Session-Request*
- To send/receive *Abort-Session-Answer*

All other options of the Diameter Base Protocol are beyond the scope of this document.

4.2.2.2 Online Specific Base Protocol Requirements

If *Accounting-Interim-Interval* AVP is not used or its value field is set to 0, the timer Ts should have a configurable default value.

5 Offline Charging

5.1 Diameter Description on the Rf Interfaces

5.1.1 Basic Principles

The offline charging functionality is based on reporting accounting information at reception of various SIP methods or ISUP messages, as most of the relevant information is contained in these messages. This is achieved by sending Diameter *Accounting Requests* (ACR) (start, interim, stop and event).

It is operator configurable in the nodes for which SIP method or ISUP messages an *Accounting Request* shall be sent, with the exception that if accounting information is collected for sessions the ACR (Start) and ACR (Stop) messages are mandatory according to the tables below. The first table (5.1) describes all possible ACRs that might be sent from a P-CSCF, I-CSCF, S-CSCF, MGCF or BGCF. All options are not, however, required in all nodes and a more node specific description of the ACRs is described in section 5.1.3.

The ACRs to be sent from a MRFC is described in table 5.2 and the ACRs to be sent from an AS is described in table 5.3.

Table 5.1: Accounting Request messages following SIP Methods or ISUP Messages

Diameter message	SIP Method /ISUP message	Mandatory/Configurable
ACR [start]	SIP 200 OK acknowledging an initial SIP INVITE	Mandatory
	ISUP:ANM (applicable for the MGCF)	Mandatory
ACR [Interim]	SIP 200 OK acknowledging a SIP RE-INVITE [e.g. change in media components]	Configurable
	Expiration of AVP [Accounting-Interim-Interval]	Configurable
ACR [stop]	A SIP BYE message	Mandatory
	Termination of an ongoing session via a SIP Final Response with error codes 4xx, 5xx, 6xx	Mandatory
	ISUP:REL (applicable for the MGCF)	Mandatory
ACR [event]	SIP 200 OK acknowledging non-session related SIP messages, which are:	
	SIP NOTIFY	Configurable
	SIP MESSAGE	Configurable
	SIP REGISTER	Configurable
	SIP SUBSCRIBE	Configurable
	SIP UNSUBSCRIBE	Configurable
	Detection of an unsuccessful SIP session set-up via: SIP Final Response (4xx, 5xx, 6xx)	Configurable
	Detection of an unsuccessful SIP session set-up via: SIP CANCEL	Configurable

Table 5.2: Accounting Request messages following SIP Methods for the MRFC

Diameter message	SIP Method /ISUP message	Mandatory/Configurable
ACR [start]	SIP 200 OK acknowledging an SIP INVITE for initiating a multimedia ad hoc conferencing session	Mandatory
ACR [Interim]	SIP ACK acknowledging a SIP INVITE to connect an UE to the conferencing session	Configurable
	Expiration of AVP [Accounting-Interim-Interval]	Configurable
ACR [stop]	A SIP BYE message	Mandatory
	Termination of an ongoing session via a SIP Final Response with error codes 4xx, 5xx, 6xx	Mandatory

Table 5.3: Accounting Request messages following SIP Methods for an AS

Diameter message	SIP Method /ISUP message	Mandatory/Configurable
ACR [event]	SIP ACK acknowledging a SIP INVITE to	Configurable

The ability of SIP methods not listed in Tables 5.1 to 5.3 to trigger ACRs is for further study.

5.1.2 Message Flows and Types

5.1.2.1 Message Flows - Successful Cases and Scenarios

5.1.2.1.1 Session Related Procedures

5.1.2.1.1.1 Session Establishment – Mobile Origination

Figure 5.1 shows the transactions that are required between CSCF and CCF during session establishment.

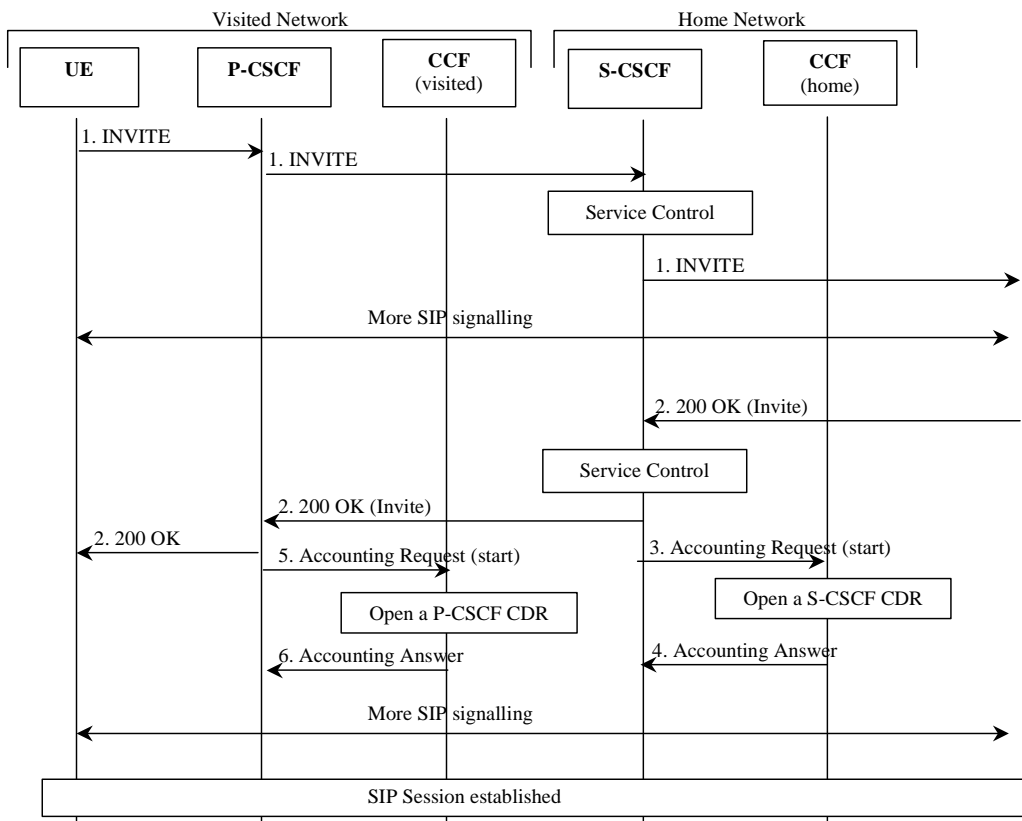


Figure 5.1: Message Sequence Chart for Offline Charging Session Establishment (Mobil Origination)

1. The session is initiated.
2. The destination party answers and a final response is received.
3. Upon reception of the final response, the S-CSCF may send an *Accounting-Request* with *Accounting-Record-Type* indicating *START_RECORD* to record start of a user session and start of a media component in the S-CSCF CDR.
4. The CCF acknowledges the reception of the data and opens a S-CSCF CDR.
5. Same as 3, but for P-CSCF.
6. Same as 4, but creating a P-CSCF CDR.

5.1.2.1.1.2 Session Establishment – Mobile Termination

Figure 5.2 shows the transactions that are required between CSCFs and CCF during the session establishment – mobile termination. The I-CSCF is only involved in the INVITE transaction.

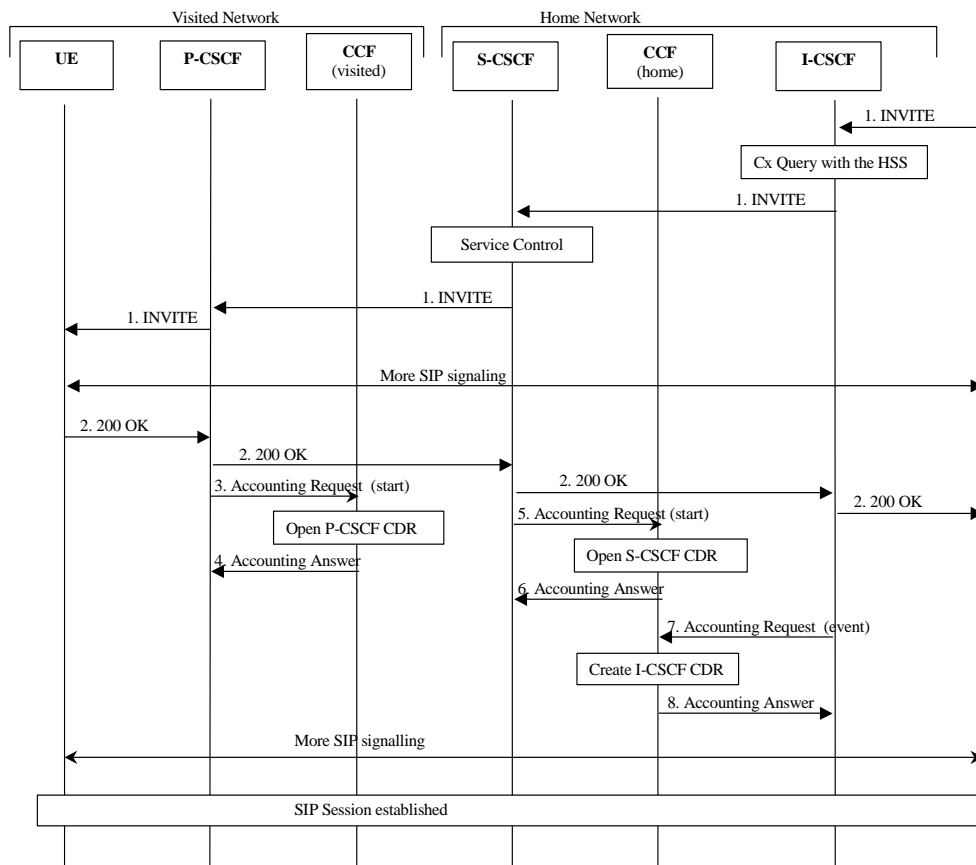


Figure 5.2: Message Sequence Chart for Offline Charging Session Establishment (Mobile Termination)

1. The session is initiated.
2. The destination party answers and a final response is sent.
- 3 – 6. These steps are identical to the corresponding steps described in Section 5.1.2.1.1.1.
7. Upon receiving the 200OK response to the INVITE transaction, the I-CSCF sends an *Accounting Request* with the *Accounting-Record-Type* set EVENT. As the subsequent SIP requests do not go through the I-CSCF, the I-CSCF is not involved in the IMS session after the closure of the initial INVITE transaction.
8. The CCF acknowledges the data received and creates an I-CSCF CDR.

5.1.2.1.1.3 Mid-Session Procedures

Figure 5.3 shows the transactions that are required between CSCF and CCF when a UE generates a re-invite in mid-session, e.g. in order to modify media component(s), or when the hold and resume procedure is executed.

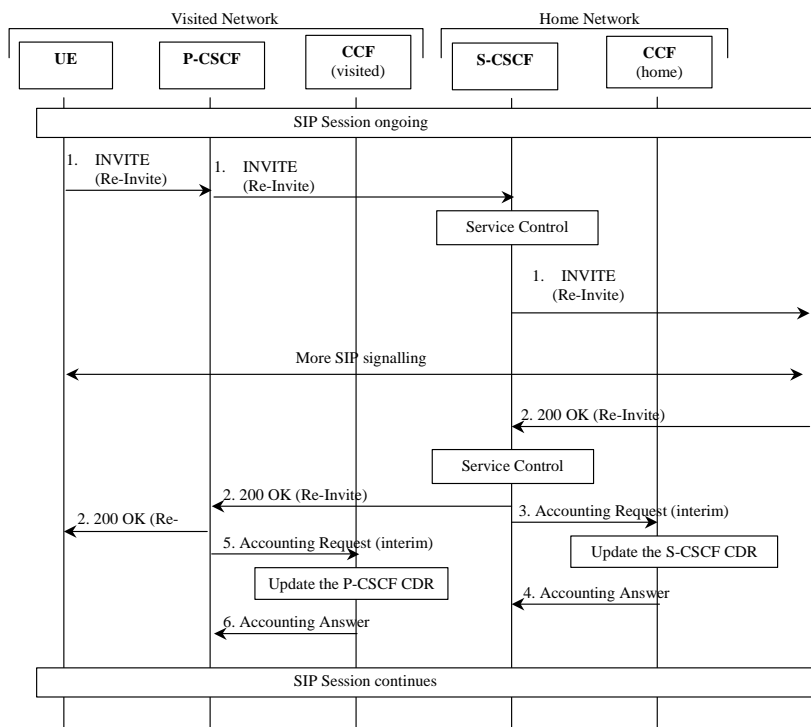


Figure 5.3: Message Sequence Chart for Offline Charging Media Modification

1. Modified media information is received from the subscriber.
2. The destination party acknowledges the media modification.
3. At modification of a media, the S-CSCF sends Accounting-Request with Accounting-Record-Type indicating INTERIM_RECORD to record modification of a media component in the S-CSCF CDR.
4. The CCF acknowledges the reception of the data and updates the S-CSCF CDR.
5. Same as 3, but for P-CSCF.
6. Same as 4, updating the P-CSCF CDR.

5.1.2.1.1.4 Session Release – Mobile Initiated

Figure 5.4 shows the transactions that are required between CSCF and CCF at session release.

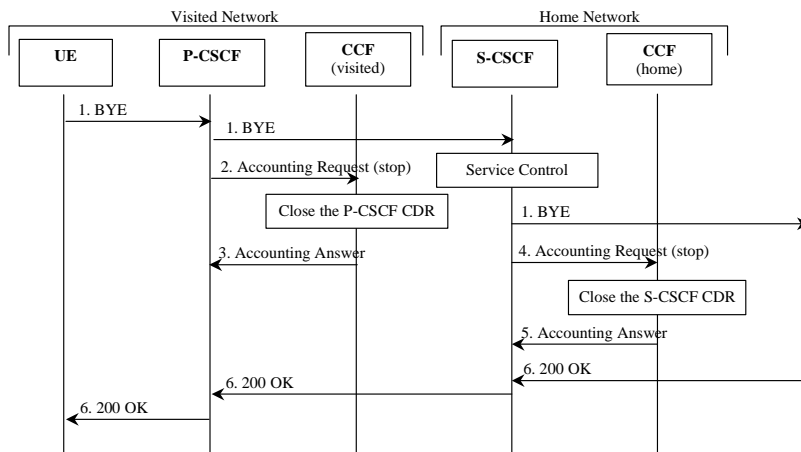


Figure 5.4: Message Sequence Chart for Offline Charging Session Release

1. The session is released.
2. At session termination the P-CSCF sends *Accounting-Request* with *Accounting-Record-Type* indicating *STOP_RECORD* to record stop of a session and stop of a media component in the P-CSCF CDR.
3. The CCF acknowledges the reception of the data and closes the P-CSCF CDR.
4. Same as 2, but for S-CSCF.
5. Same as 3, closing the S-CSCF CDR.
6. The release is acknowledged.

5.1.2.1.1.5 Session Release – Network Initiated

The message flow for this case is identical to the mobile initiated session release described in subsection 5.1.2.1.5. However before invoking the procedure, the UE receives a command requesting session release from the network.

Editor’s note: the correctness of the above assertion needs to be verified.

5.1.2.1.1.6 Session Release – CCF initiated

The IMS operator may request the release of SIP session(s) upon certain trigger conditions being met, for example as soon as a fraud is detected. The communication between CCF and external functions that convey that request to the CCF is not in the scope of this document.

Figure 5.5 shows the transactions that are required between CCF and S-CSCF in order to release an ongoing SIP session.

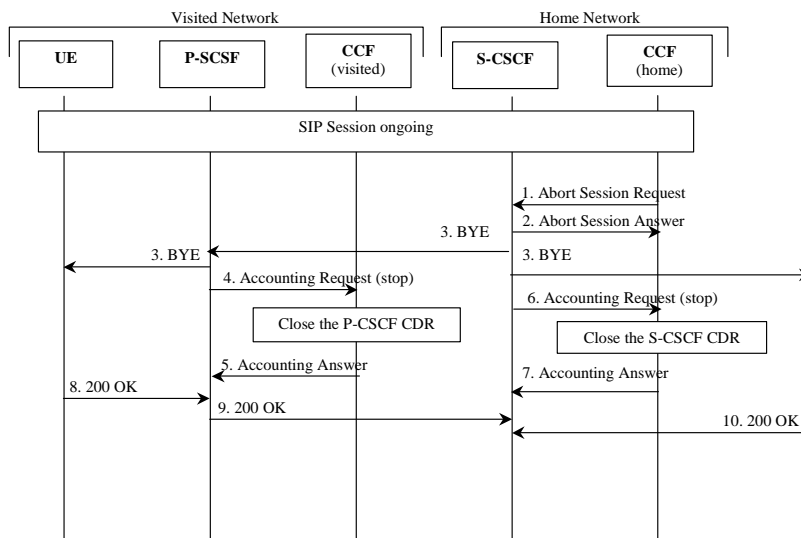


Figure 5.5: Message Sequence Chart for Offline Charging CCF Initiated Session Release

1. The CCF may initiate the SIP session release by sending *Abort-Session-Request* message to the S-CSCF.
2. The S-CSCF acknowledges the *Abort-Session-Request* by sending *Abort-Session-Answer* message to the CCF. The Result-Code is included in the answer and may be set to DIAMETER-SUCCESS to indicate successful termination. CCF closes the CDR. The record closure time in the CDR is the time when *Abort-Session-Answer* message has been received and the cause for closure reflect the cause why the session has been aborted.
3. The S-CSCF initiates the SIP session release by sending SIP BYE request to both the originating and the terminating parties, as specified in TS 23.218 [5].
4. At session termination the P-CSCF sends *Accounting-Request* with *Accounting-Record-Type* indicating STOP_RECORD to record stop of a session and stop of a media component in the P-CSCF CDR.
5. The CCF acknowledges the reception of the data and closes the P-CSCF CDR CDR.
6. Same as 4, but for S-CSCF.
7. Same as 5, but for S-CSCF CDR.
8. The S-CSCF receives the 200 OK responses from originating and terminating parties.

5.1.2.1.2 Session-Unrelated Procedures

Figure 5.6 shows the transactions that are required between CSCF and CCF for session-unrelated IMS procedures, i.e. those that relate to the DIAMETER ACR (event), as listed in table 5.1.

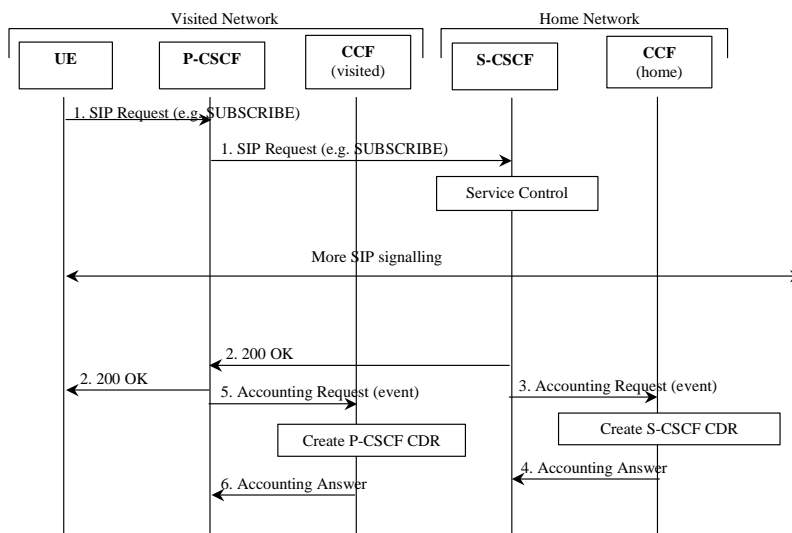


Figure 5.6: Message Sequence Chart for Offline Charging Session-Unrelated Procedure

1. A SIP request (e.g. SUBSCRIBE, MESSAGE or REGISTER) is received from the subscriber.
2. The SIP request is successfully acknowledged. Note: Depending on the used SIP method, there might be additional signalling between steps 1 and 2.
3. After the completion of the procedure, the S-CSCF sends *Accounting-Request* with *Accounting-Record-Type* indicating *EVENT_RECORD* to record transaction specific information in the S-CSCF CDR.
4. The CCF acknowledges the reception of the data and produces an S-CSCF CDR.
5. Same as 3, but for P-CSCF.
6. Same as 4, creating a P-CSCF CDR.

5.1.2.1.3 MRFC Related Procedures

5.1.2.1.3.1 Multi-Party Call

Figure 5.7 shows the establishment of an ad hoc conference (multiparty call). An AS (acting as B2BUA) performs third party call control with the MRFC, where the S-CSCF is in the signalling path. The Application Server that is in control of the ad hoc conference is aware of the MRFC capabilities and is also operating as a B2BUA performing third party call control.

Note: Only accounting information sent from the MRFC is showed in the figure.

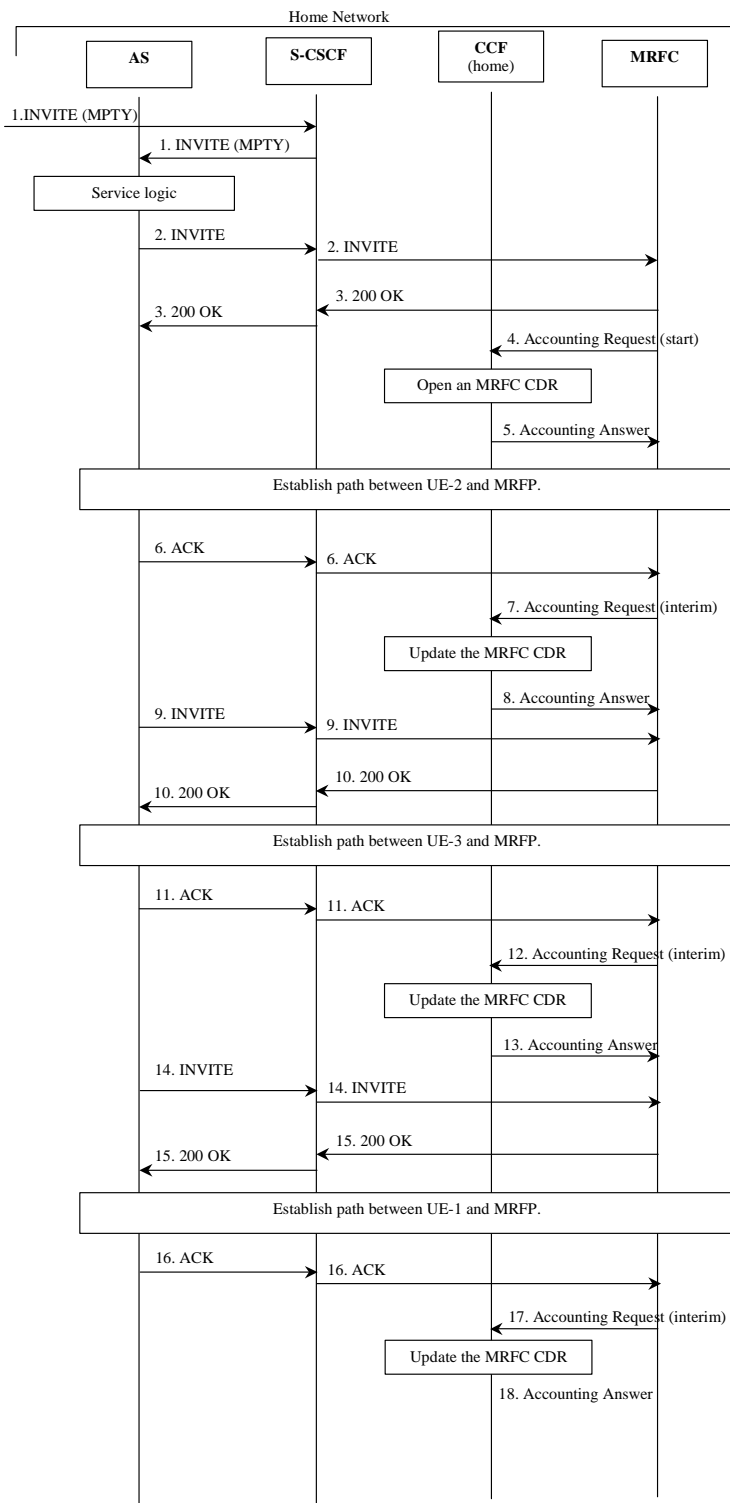


Figure 5.7: Message Sequence Chart for Offline Charging Multi-Party Call Establishment in MRFC.

1. Sessions exist between UE-1 and UE-2, and between UE-1 and UE-3. A request is received from UE-1 for putting all parties together to a multi-party call.
- 2-3. Request and acknowledgement to initiate multi-party call.
4. At session establishment the MRFC may send an Accounting-Request with Accounting-Record-Type indicating START_RECORD to record start of multi-party call in the MRFC CDR

5. The CCF acknowledges the reception of the data and creates the MRFC CDR.
6. Dialog between UE-2 and MRFP has been established.
7. The MRFC may send an Accounting-Request with Accounting-Record-Type indicating INTERIM_RECORD to report that UE-2 has been connected to the multi-party call.
8. The CCF acknowledges the reception of the data and updates the MRFC CDR.
9. New request sent to MRFC to prepare dialog for UE-3.
10. Request acknowledged.
11. Dialog between UE-3 and MRFP has been established.
12. The MRFC may send an Accounting-Request with Accounting-Record-Type indicating INTERIM_RECORD to report that UE-3 has been connected to the multi-party call.
13. The CCF acknowledges the reception of the data and updates the MRFC CDR.
14. New request sent to MRFC to prepare dialog for UE-1.
15. Request acknowledged.
16. Dialog between UE-1 and MRFP has been established.
17. The MRFC may send an Accounting-Request with Accounting-Record-Type indicating INTERIM_RECORD to report that UE-1 has been connected to the multi-party call.
18. The CCF acknowledges the reception of the data and updates the MRFC CDR.

5.1.2.1.4 AS Related Procedures

5.1.2.1.4.1 AS as Redirect Server

Figure 5.8 shows the case where an Application Server acts as a redirect server. In the figure below, UE-1 sets up a session towards UE-2 but due to Call Forwarding functionality located in the AS, a new number (to UE-3) is returned to UE-1. Finally UE-1 sets the session towards UE-3.

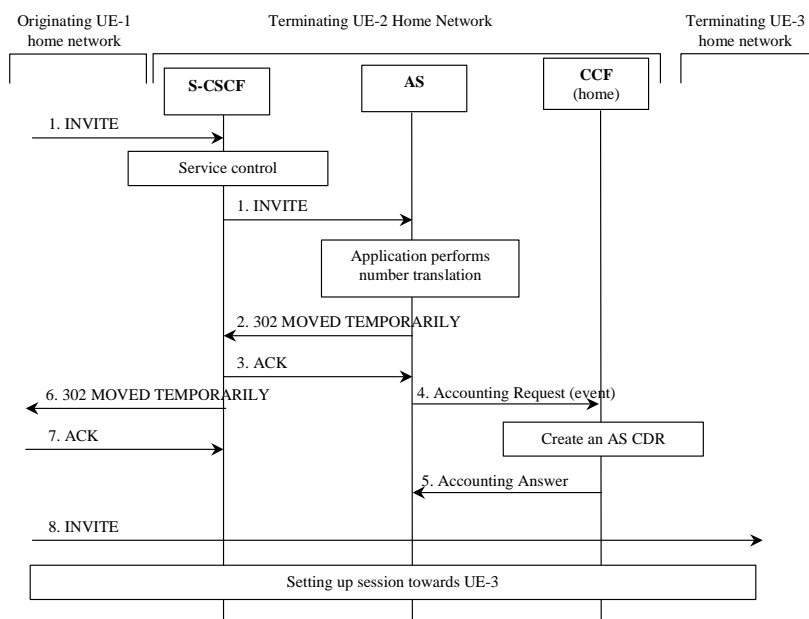


Figure 5.8: Message Sequence Chart for Offline Charging AS Based Redirection

1. Sessions initiated by UE-1 towards UE-2.
- 2-3. Response indicating that session should be redirected towards another number (UE-3).
4. After successful service execution, the AS sends *Accounting-Request* with *Accounting-Record-Type* indicating *EVENT_RECORD* to record service specific information in the AS CDR.
5. The CCF acknowledges the reception of the data and creates the AS CDR.
- 6-7. Response indicating that session should be redirected towards another number (UE-3).
8. Session is initiated by UE-1 towards UE-3.

5.1.2.2 Message Flows - Error Cases and Scenarios

This section describes various error cases and how these should be handled.

5.1.2.2.1 Reception of SIP error messages

Normally a SIP session is closed by the reception of the BYE message. There are, however, a few cases when no BYE message is received and the ACR [stop] message must be triggered by the reception of other messages.

The following events should trigger the sending of an ACR [stop] message:

- Reception of a 4xx, 5xx and 6xx response that triggers the termination of the SIP session in the receiving entity.

Note: This also covers the error handling in originating procedures, as a CANCEL request sent by the originating party to cancel a session invitation will trigger the terminating party to send a 487 final response to the INVITE.

5.1.2.2.2 CCF connection failure

When the connection towards the primary CCF is broken, the sending of accounting information should continue towards a secondary CCF (if configured). For further CCF connection failure functionality, see section '*Transport Failure Detection*' in [3].

If no CCF is reachable the network element may buffer the generated accounting data in non-volatile memory. Once the CCF connection is working again, all accounting messages stored in the buffer shall be sent to the CCF, in the order they were stored in the buffer. This behaviour is specified in [3].

5.1.2.2.3 Duplicate detection

A Diameter client marks possible duplicate request messages (e.g., retransmission due to the link failover process) with the *Re-Transmission AVP*.

If the CCF receives a message that is marked as retransmitted and this message was already received, than it discards the duplicate message. If the re-transmitted message was not yet received, its information is taken into account when generating the CDR. The CDRs are marked if information from a duplicated messages is used.

5.1.2.2.4 CCF detected failure

The CCF closes a CDR when it detects that expected Diameter ACRs for a particular SIP session have not been received for a period of time.

5.1.2.2.5 Error case – Registration not allowed

Figure 5.9 shows the transactions that are required between CSCF and CCF in order to include data in the CDR during the registration procedure, in case registration is not allowed. HSS is not shown to simplify the flow.

This sequence only shows one error message, but other error messages should be handled in a similar way.

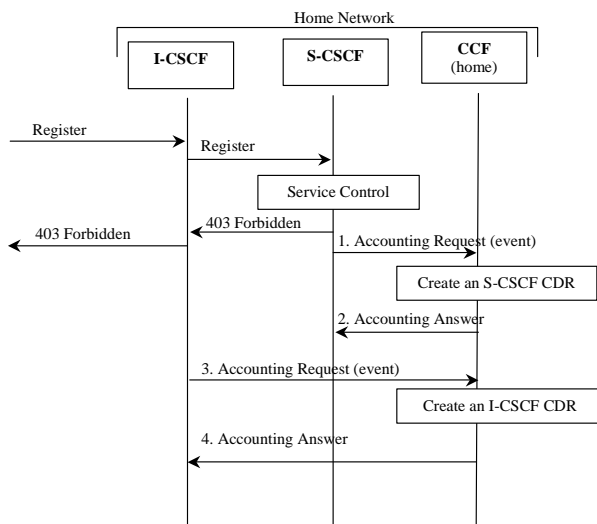


Figure 5.9: Message Sequence Chart for Offline Charging Error in Registration

1. After the registration procedure has been rejected, the S-CSCF sends *Accounting-Request* with *Accounting-Record-Type* indicating *EVENT_RECORD* to record registration specific information in the S-CSCF CDR.
2. The CCF acknowledges the reception of the data and creates a S-CSCF CDR.
3. Upon indication that the registration procedure has been rejected, the I-CSCF sends *Accounting-Request* with *Accounting-Record-Type* indicating *EVENT_RECORD* to record registration specific information in the I-CSCF CDR.

The CCF acknowledges the reception of the data and creates a I-CSCF CDR.

5.1.2.23 Message Formats

[Editor: This section is subject to further review regarding the specific fields of these messages. Ericsson's Action Item]

5.1.3.12.2.4 Summary of Offline Charging Message Formats

This section defines Command-Code values for this IMS Charging Diameter application.

The IMS Charging application re-uses the *Accounting-Request* and *Accounting-Answer* messages from the base Diameter protocol.

[The CCF may send an unsolicited message indicating to the S-CSCF to release the ongoing SIP session due for example to fraud detection. For this purpose the IMS Charging application re-uses the *Abort-Session-Request* and *Abort-Session-Answer* from the base Diameter protocol.](#)

Messages for IMS charging make use of the existing *Accounting-Request* and *Accounting-Answer* messages. Table 5.1.142 describes the use of these messages for offline charging.

Table 5.1.124: Rf and Command-Code values

Command-Name	Source	Destination	Abbreviation	Code
Accounting-Request	S-CSCF, I-CSCF, P-CSCF, MRFC, MGCF, BGCF, AS	CCF	ACR	271
Accounting –Answer	CCF	S-CSCF, I- CSCF, P- CSCF, MRFC, MGCF, BGCF, AS	ACA	271
Abort-Session-Request	CCF	S-CSCF	ASR	274
Abort-Session-Answer	S-CSCF	CCF	ASA	274

Following the base protocol specification, the following “types” of accounting data may be sent:

- start session accounting data
- interim session accounting data
- end session accounting data
- event accounting data

Event accounting data, is used for non-session related accounting data, such as a simple registration or interrogation. [A DIAMETER client implementation may choose to use Event accounting data also for session-related accounting data if a non-volatile memory for the safe storage of accounting records is available in the NE.](#) ~~Event accounting data may also be used for session-related accounting data if a non-volatile memory for the safe storage of accounting records is available in the NE.~~

The ~~S~~sections [that follow xxx](#) specify the accounting data ~~to be~~ [that are](#) sent by each of the IMS network elements:

- S-CSCF
- P-CSCF
- I-CSCF
- MRFC
- MGCF
- BGCF
- [AS](#)

5.1.3.2 Structure for the Accounting-Request and Accounting-Answer Message Formats

The following is the basic structure shared by all offline charging messages. This is based directly on the Accounting-Request and Accounting-Answer messages defined in the base Diameter protocol specification.

- Accounting-Request Message

The table below illustrates the basic structure of a Diameter Accounting-Request message as it could be used for IMS charging.

IMS Accounting-Request Message Contents
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{ Vendor-Specific-Application-Id }
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Sub-Session-Id]
[Accounting-RADIUS-Session-Id]
[Accounting-Multi-Session-Id]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
<IMS Accounting Data>
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[(User) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Start Time stamp]
[Stop Time stamp]
[List of Application Servers involved]
[Application provided Called Party Address]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type] What information do we actually want, the complete SDP data or only certain attributes?
[Cause]

IMS-specific parts are shown in bold. ~~“IMS Accounting Data” actually refers to several Attribute Value Pairs (AVPs), as will be seen in later sections. The following sections describes more detailed which AVPs to use for various cases.~~

NOTE: the Start Time stamp and Stop Time stamp AVPs are used only in case the DIAMETER client uses one-time event accounting data transfer at the end of the session. The Start Time stamp AVP reflects the time when the session started. The Stop Time stamp AVP reflects the time when the session ended.

- Accounting-Answer Message

The table below illustrates the basic structure of a Diameter Accounting-Answer message as it could be used for IMS charging.

IMS Accounting-Answer Message Contents
< Diameter Header: 271, PXY >
< Session-Id >
{ Acct-Application-Id }
{ Vendor-Specific-Application-Id }
{ Result-Code }
{ Origin-Host }
{ Origin-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Sub-Session-Id]
[Accounting-RADIUS-Session-Id]
[Accounting-Realtime-Required]
[Accounting-Multi-Session-Id]
[Error-Reporting-Host]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [AVP]
* [Proxy-Info]

- [Abort-Session-Request](#)

The table below illustrate the basic structure of a Diameter *Abort-Session-Request* message as it could be used for IMS charging.

IMS Abort-Session-Request Message Contents
< Diameter Header: 274, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Vendor-Specific-Application-Id }
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
[User-Name]
[Origin-State-Id]
[Re-Transmission]
*[AVP]
*[Proxy-Info]
*[Record-Route]

- [Abort-Session-Answer](#)

The table below illustrate the basic structure of a Diameter *Abort-Session-Answer* message as it could be used for IMS charging.

IMS Abort-Session-Answer Message Contents
< Diameter Header: 274, PXY >
< Session-Id >
{ Vendor-Specific-Application-Id }
{ Result-Code }
{ Origin-Host }
{ Origin-Realm }
[User-Name]
[Origin-State-Id]

[Re-Transmission]
* [AVP]
* [Proxy-Info]

5.1.3.3 Detailed S-CSCF Message Formats

Charging data may be sent by the S-CSCF at the following points:

- [Start of Session](#) (a “Start Session” record type)
- [Mid-session information](#) (an “Interim Session” record type), to be used in the following cases:
 - ▣ [start of media component](#)
 - ▣ [end of media component](#)
 - ▣ [start of user session](#)
 - ▣ [reception of access charging correlation information](#)
- [End of Session](#) (a “Stop Session” record type)
- [Events](#) (e.g. registration, One-time event accounting data transfer) (an “Event Record” record type)
The event messages may contain description of multiple media components.

- Start of Session

IMS S-CSCF Accounting-Request Message Contents for Start of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{Vendor-Specific-Application-Id}
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[(User) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[List of Application Servers involved]
[Application provided Called Party Address]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type] What information do we actually want, the complete SDP data or only certain attributes?

- Mid-Session Information

IMS S-CSCF Accounting-Request Message Contents <u>for Mid-Session Information</u>
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{ Vendor-Specific-Application-Id }
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
<u>[Re-Transmission]</u>
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]
*[Media Component Initiator]
[Media Type] What information do we actually want, the complete SDP data or only certain attributes?
[Correlation vector access part]

- End of Session

IMS S-CSCF Accounting-Request Message Contents <u>for End of Session</u>
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{ Vendor-Specific-Application-Id }
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
<u>[Re-Transmission]</u>
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]
[Cause]

- Events





IMS S-CSCF Accounting-Request Message Contents <u>for Events</u>
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{ Vendor-Specific-Application-Id }
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }

{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[(User) Session Id]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Start Time stamp]
[Stop Time stamp]
[List of Application Servers involved]
[Application provided Called Party Address]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
[Correlation vector access part]
*[Media Component Initiator]
[Media Type]
[Cause]

*NOTE: the [Start Time stamp](#) and [Stop Time stamp](#) AVPs are used only in case the DIAMETER client uses *one-time event accounting data transfer* at the end of the session. The [Start Time stamp](#) AVP reflects the time when the session started. The [Stop Time stamp](#) AVP reflects the time when the session ended.*

5.1.3.4 Detailed P-CSCF Message Formats

Charging data may be sent by the P-CSCF at the following points:

- [Start of Session](#) (a “Start Session” record type)
- [Mid-session information](#) (an “Interim Session” record type), to be used in the following cases:
 -  [start of media component](#)
 -  [end of media component](#)
 -  [start of user session](#)
 -  [reception of access charging correlation information](#)
- [End of Session](#) (a “Stop Session” record type)
- [Events](#) (e.g. registration, One-time event accounting data transfer) (an “Event Record” record type)
The event messages may contain description of multiple media components.

- Start of Session

IMS P-CSCF Accounting-Request Message Contents for Start of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{Vendor-Specific-Application-Id}
{ Origin-Host }

{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[(User) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Served party IP Address]
[Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type]
[Authorised QoS]

- Mid-Session Information

IMS P-CSCF Accounting-Request Message Contents for Mid-Session Information
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{Vendor-Specific-Application-Id}
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
*[Media Component Initiator]
[Correlation vector access part]
[Authorised QoS]

- End of Session

IMS P-CSCF Accounting-Request Message Contents for End of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } – IMS Charging
{Vendor-Specific-Application-Id}
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }

{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]
[Cause]

- Events

IMS P-CSCF Accounting-Request Message Contents for Events
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Acct-Application-Id } — IMS Charging
{Vendor-Specific-Application-Id}
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[Node Address]
[(User) Session ID]
[Called Party Address (Public User ID)]
[Served party IP Address]
[Time stamp]
[Start Time stamp]
[Stop Time stamp]
[Inter Operator Identifier(s)]
[Media Component Initiator]
[Media Type]
[Correlation Vector IMS part (ICID)]
[Correlation vector access part]
[Authorised QoS]
[Cause]

*NOTE: the [Start Time stamp](#) and [Stop Time stamp](#) AVPs are used only in case the DIAMETER client uses *one-time event accounting data transfer* at the end of the session. The [Start Time stamp](#) AVP reflects the time when the session started. The [Stop Time stamp](#) AVP reflects the time when the session ended.*

5.1.2.23.5 Detailed I-CSCF Message Formats

Charging data may be sent by the I-CSCF at the following points:

- Start of mobile terminating Session (a “Start Session” record type)
- End of mobile terminating Session establishment transaction (a “Stop Session” record type)
- Events (e.g. registration, One-time event accounting data transfer) (an “Event Record” record type)

- Start of Terminating Session Establishment

IMS I-CSCF Accounting-Request Message Contents for Start of Terminating Session Establishment
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{Vendor-Specific-Application-Id}{Acct-Application-Id} – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[Node Address]
[(User) Session Id]
[Calling Party Address (Public User ID)]
[Time stamp]
[Inter Operator Identifier]
[S-CSCF information]
[Correlation Vector IMS part (ICID)]

- End of Terminating Session Establishment

IMS I-CSCF Accounting-Request Message Contents for End of Terminating Session Establishment
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{Vendor-Specific-Application-Id}{Acct-Application-Id} – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]

[(IMS) Event Type]
[Time stamp]
[Cause]





- Events

IMS I-CSCF Accounting-Request Message Contents for Events
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] {Acct-Application-Id}—IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[Node Address]
[(User) Session Id]
[Calling Party Address (Public User ID)]
[Time stamp]
[Start Time stamp]
[Stop Time stamp]
[Cause]
[Inter Operator Identifier]
[S-CSCF information]
[Correlation Vector IMS part (ICID)]

NOTE: the Start Time stamp and Stop Time stamp AVPs are used only in case the DIAMETER client uses one-time event accounting data transfer at the end of the session. The Start Time stamp AVP reflects the time when the session started. The Stop Time stamp AVP reflects the time when the session ended.

5.1.32.2.6 Detailed MRFC Message Formats

Charging data may be sent by the MRFC at the following points:

- Start of Session (a “Start Session” record type)
- —Mid-session information (an “Interim Session” record type), to be used in the following cases:
 -  start of media component
 -  end of media component
 -  start of user session
 -  ...
- —End of Session (a “Stop Session” record type)

- Events (e.g. One-time event accounting data transfer) (an “Event Record” record type)
The event messages may contain description of multiple media components.

- Start of Session

IMS MRFC Accounting-Request Message Contents for Start of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[(User) Session ID]
[Service ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type]

- Mid-Session Information

IMS MRFC Accounting-Request Message Contents for Mid-session Information
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Service ID]
[Time stamp]
*[Media Component Initiator]
[Media Type]

- End of Session

IMS MRFC Accounting-Request Message Contents for End of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } -- IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Service ID]
[Time stamp]
[Cause]

- Events

IMS MRFC Accounting-Request Message Contents for Events
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } -- IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[(User) Session ID]
[Service ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Start Time stamp]
[Stop Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type]
[Cause]

NOTE: the Start Time stamp and Stop Time stamp AVPs are used only in case the DIAMETER client uses one-time event accounting data transfer at the end of the session. The Start Time stamp AVP reflects the time when the session started. The Stop Time stamp AVP reflects the time when the session ended.

5.1.2.23.7 Detailed MGCF Message Formats

Charging data may be sent by the MGCF at the following points:

- Start of Session (a “Start Session” record type)
- –Mid-session information (an “Interim Session” record type), to be used in the following cases:
 - start of media component
 - end of media component
 - start of user session
 - ...
- –End of Session (a “Stop Session” record type)
- Events (e.g. One-time event accounting data transfer) (an “Event Record” record type)
The event messages may contain description of multiple media components.

- Start of Session

IMS MGCF Accounting-Request Message Contents for Start of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id]{ Acct-Application-Id } – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[(User) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
[Media Component Initiator]
[Media Type]
[TrunkGroup ID Incoming/Outgoing]
[BearerService]

- Mid-Session Information

IMS MGCF Accounting-Request Message Contents for Mid-Session Information
< Diameter Header: 271, REQ, PXY >

< Session-Id > -- Diameter Session Id
{Vendor-Specific-Application-Id} {Acct-Application-Id} – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[Time stamp]
[Media Component Initiator]
[Media Type]
[TrunkGroup ID Incoming/Outgoing]
[BearerService]

- End of Session

IMS MGCF Accounting-Request Message Contents for End of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{Vendor-Specific-Application-Id} {Acct-Application-Id} – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]
[Cause]

- Events





IMS MGCF Accounting-Request Message Contents for Events
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{Vendor-Specific-Application-Id} {Acct-Application-Id} – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]

* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Role of node]
[(User) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Start Time stamp]
[Stop Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type]
[TrunkGroup ID Incoming/Outgoing]
[BearerService]
[Cause]

NOTE: the Start Time stamp and Stop Time stamp AVPs are used only in case the DIAMETER client uses one-time event accounting data transfer at the end of the session. The Start Time stamp AVP reflects the time when the session started. The Stop Time stamp AVP reflects the time when the session ended.

5.1.2.23.8 Detailed BGCF Message Formats

Charging data may be sent by the BGCF at the following points:

- Start of Session (a “Start Session” record type)
- —Mid-session information (an “Interim Session” record type), to be used in the following cases:
 -  start of media component
 -  end of media component
 -  start of user session
 -  ...
- —End of Session (a “Stop Session” record type)
- Events (e.g. One-time event accounting data transfer) (an “Event Record” record type)
The event messages may contain description of multiple media components.

- Start of Session, Start of Media Component

IMS BGCF Accounting-Request Message Contents for Start of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
{ Vendor-Specific-Application-Id } { Acct-Application-Id } — IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]

[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[(User) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type]

- Mid-Session Information

IMS BGCF Accounting-Request Message Contents for Mid-Session Information
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } -- IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]
*[Media Component Initiator]
[Media Type]

- End of Session

IMS BGCF Accounting-Request Message Contents for End of Session
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } -- IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]

[Cause]





- Events

IMS B2C Accounting-Request Message Contents for Events
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
<u>{Vendor-Specific-Application-Id}</u> {Acct-Application-Id} – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
<u>[Re-Transmission]</u>
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[(User) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
<u>[Start Time stamp]</u>
<u>[Stop Time stamp]</u>
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
<u>[Media Type]</u>
[Cause]

NOTE: the Start Time stamp and Stop Time stamp AVPs are used only in case the DIAMETER client uses one-time event accounting data transfer at the end of the session. The Start Time stamp AVP reflects the time when the session started. The Stop Time stamp AVP reflects the time when the session ended.

5.1.2.23.9 Detailed AS Message Formats

Charging data is sent by the AS at the following points:

- Start of Session (a “Start Session” record type)
- Mid-session information (an “Interim Session” record type), to be used in the following cases:
 -  start of media component
 -  end of media component
 -  mid-session credit check
 -  ...
- End of Session (a “Stop Session” record type)
- Event (e.g. direct debiting, one-time event accounting data transfer) (an “Event Record” record type)
The event messages may contain description of multiple media components.

- Start of Session

IMS AS Accounting-Request Message Contents <u>for Start of Session</u>
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
<u>{Vendor-Specific-Application-Id}</u> {Acct-Application-Id}—IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
<u>[Re-Transmission]</u>
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[(User) Session ID]
[Service ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
*[Media Component Initiator]
[Media Type]

- Mid-Session Information

IMS AS Accounting-Request Message Contents <u>for Mid-Session Information</u>
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
<u>{Vendor-Specific-Application-Id}</u> {Acct-Application-Id}—IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
<u>[Re-Transmission]</u>
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]
*[Media Component Initiator]
[Media Type]

- End of Session

IMS AS Accounting-Request Message Contents <u>for End of Session</u>
< Diameter Header: 271, REQ, PXY >

< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[Time stamp]
[Cause]
[Pricing-Enquiry-Indication]
[Requested-Service-Unit]
[Used-Service-Unit]
* [Service-Parameter-Info]

- Events

IMS <u>AS</u> Accounting-Request Message Contents <u>for Events</u>
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id] { Acct-Application-Id } – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [Proxy-Info]
* [Route-Record]
[(IMS) Event Type]
[(User) Session Id]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Start Time stamp]
[Stop Time stamp]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
* [Media Component Initiator]
[Media Type]
[Cause]

*NOTE: the [Start Time stamp](#) and [Stop Time stamp](#) AVPs are used only in case the DIAMETER client uses *one-time event accounting data transfer* at the end of the session. The [Start Time stamp](#) AVP reflects the time when the session started. The [Stop Time stamp](#) AVP reflects the time when the session ended.*

5.2 CDR Description ~~On~~on the Bi Interface

5.2.1 CDR Field Types

The following Standard CDR content and format are considered:

- S-CSCF-CDR ~~generated by the S-CSCF~~generated based on information from the S-CSCF
- I-CSCF-CDR ~~generated by the I-CSCF~~generated based on information from the I-CSCF
- P-CSCF-CDR ~~generated by the P-CSCF~~generated based on information from the P-CSCF
- BGCF-CDR ~~generated by the BGCF~~generated based on information from the BGCF
- MGCF-CDR ~~generated by the MGCF~~generated based on information from the MGCF
- MRFC-CDR ~~generated by the MRFC~~generated based on information from the MRFC
- AS-CDR generated based on information from the AS

~~The content of CDRs associated with services from Application Servers are currently not specified in this document.!!!
I ASSUME WE NEED TO DEFINE A CDR FOR THE AS, EVEN IF WE EARLIER DECIDED NOT TO DO THAT.
OR WHAT ARE WE SUPPOSED TO DO WITH THE ACCOUNTING DATA SENT FROM AN AS TO THE
CCF???~~

The content of each CDR type is defined in the tables that are part of this clause. For each CDR type the field definition includes the field name, description and category.

Equipment vendors shall be able to provide all of the fields listed in the CDR content table in order to claim compliance with the present document. However, since CDR processing and transport consume network resources, operators may opt to eliminate some of the fields that are not essential for their operation. This operator provisionable reduction is specified by the field category.

A field category can have one of two primary values:

- M** This field is **Mandatory** and shall always be present in the CDR.
- C** This field shall be present in the CDR only when certain **Conditions** are met. These **Conditions** are specified as part of the field definition.

All other fields are designated as **Operator (O)** provisionable. Note that previously the letter "**O**" represented the word "Optional". Using TMN management functions or specific tools provided by an equipment vendor, operators may choose if they wish to include or omit the field from the CDR. Once omitted, this field is not generated in a CDR. To avoid any potential ambiguity, a CDR generating element **MUST** be able to provide all these fields. Only an operator can choose whether or not these fields should be generated in their system.

Those fields that the operator wishes to be present are further divided into a mandatory and conditional categories into mandatory and conditional categories:

- O_M** This is a field that, if provisioned by the operator to be present, shall always be included in the CDRs. In other words, an **O_M** parameter that is provisioned to be present is a mandatory parameter.
- O_C** This is a field that, if provisioned by the operator to be present, shall be included in the CDRs when the required conditions are met. In other words, an **O_C** parameter that is configured to be present is a conditional parameter.

The ~~I~~IMS Servers~~CCF~~ shall be able to provide the CDRs at the ~~CCF~~Bi interface in the format and encoding described in the present document. Additional CDR formats and contents may be available at the interface to the billing system to meet the requirements of the billing system.

5.2.2 CDR Triggers

Reflecting the usage of multimedia sessions IMS CDRs shall be generated on a per session level. In the scope of this specification the term “session” refers always to a SIP session. The coherent media components are reflected inside the session CDR with a media component container comprising of all the information necessary for a media component.

A session CDR shall be opened for a successfully initiated SIP session once the ~~200 OK ???~~ SIP message is received in the node generating the CDR. ~~The 200 OK message shall be the one corresponding to the respective SIP INVITE message. If chargeable events already occur before the receipt of the SIP 200 OK message (e.g. obtained from the SIP INVITE message) this has to be reflected in the CDR as well.~~ A session CDR shall be closed once the session is terminated. For a SIP session this is considered to be the case once the SIP BYE message ~~or an error message~~ is received in the node generating the CDR. Partial session CDRs shall be able to be generated due to following circumstances/events:

- on a regular time basis (e.g. each 30 minutes)
- due to service change, e.g. change in a Media component related to this session (i.e. a component is added or removed or the QoS parameters changed for a component)

Charging data for media components associated with a session are handled inside the Session CDRs as follows:

A new media component container is added into a session CDR each time a media component is added to a session. A media component container is closed once the related media component is removed from a session. The following figure illustrates this principle:

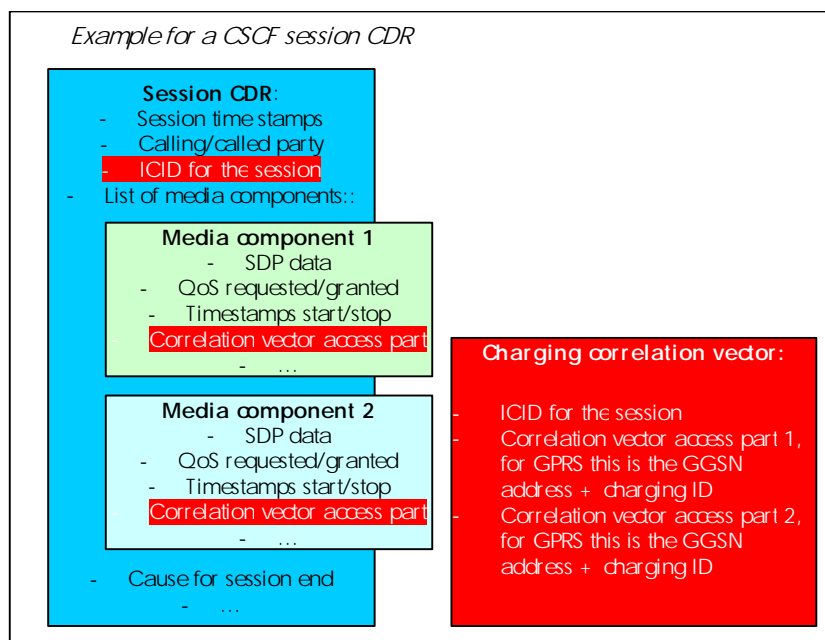


Figure 5.107: Charging Data Record Structure

A media component container is added into the session CDR when the associated SIP 200 OK message (the one corresponding to the appropriate SIP INVITE message) is received in the node generating the CDR. An appropriate media component start time stamp shall reflect the start of this media component. A media component is supposed to be removed from a session once either the SIP BYE message or a SIP 200 OK message (corresponding to the SIP INVITE message removing a media) is received in the node generating the CDR. The removal of a media component (either due to session release or due to a SIP INVITE/200 OK message pair during a session) shall be reflected with an appropriate time stamp inside the media components. If a media component is removed from an ongoing session, the related media component container shall not be carried forward to subsequent partial CDRs (if any).

The charging correlation vector is composed in the way that the IMS generated part (ICID) is put into the “session part” of the CDR and the access-generated part is associated with the coherent media components of a session. For GPRS being used to access IMS, the access part of the correlation vector comprises the GPRS charging ID and the GGSN address. For further information regarding the composition of the charging correlation vector refer to the appropriate section in TS 32.200 [2].

To reflect chargeable events not directly related to a session the S-CSCF may generate CDR(s) at the registration respectively de-registration event as well as for events related to the subscription/invocation of services at CSCF level.

(Editor's note: further elaboration is required on this subject. Open Issues: Should CDRs that do not lead to a successful session be generated? Define the data items to charge a user that has a payload in their signalling.)

5.2.3 CDR Content

5.2.3.1 Charging Data in S-CSCF (S-CSCF-CDR)

Table 5.2.453: S-CSCF Charging Data (S-CSCF-CDR)

Field	Category	Description
Record Type	O _M	Identifies the type of record: S-CSCF-CDR.
Event Type	O _M	Reflects the type of chargeable telecommunication service/event for which the CDR is generated, such as: "session", "register", "subscribe".
Role of node	O _M	Specifies the role of the CSCF if relevant for the chargeable telecommunication service/event, which is either: originating role (serving A) terminating role (serving B)
Node Address	O _M	The address of the node providing the information for the CDR
Session ID	O _M	The Session identification. For a SIP session the Session ID contains the SIP Call ID as defined in the Session Initiating Protocol.
Calling Party Address (Public User ID)	O _M	The address of the party initiating a session (SIP URL, E.164 ...)
Called Party Address (Public User ID)	O _M	The address of the party to whom a session is established (SIP URL, E.164 ...)
Private User ID (served party)	O _M	Holds the used Network Access Identifier according to RFC2486[6].
Record opening time	O _M	Time stamp reflecting either the start of a session when the Session is activated in the CSCF (If SIP is used this corresponds to the SIP 200 OK message received in the node generating the CDR) or the partial record opening time for subsequent partial records.
Record closure time	O _M	Time stamp reflecting the stop of a session when the Session is released in the CSCF (last record) (If SIP is used this corresponds to the SIP Bye message received in the node generating the CDR) or record closure time for partial records.
Cause	O _M	Cause for session disconnection
List of Application Servers involved	O _C	Holds a list ASs comprising the IP address/SIP URLs of the involved AS (if any) <i>(note: the format may be according to TS-23.218)</i>
Application provided Called Party Address	O _C	Holds the called party number if it is determined by an application (SIP URL, E.164).
Inter Operator Identifier(s)	O _C	Holds the identification of the network neighbours (incoming/outgoing) as exchanged via SIP signalling.

Field	Category	Description
Local Sequence Number	O _M	Contains a unique record number created by this node.
Record Sequence Number	O _C	Number of partial record if applicable.
Record Output Reason	O _M	Identifies the reason for CDR output, such as: timeLimit, serviceChange (e.g. change in media components), networkInternalReasons, lastCDR, timeChange
Correlation Vector IMS part (ICID)	O _M	Holds the ICID as either received via incoming SIP signalling or generated in the node generating the CDR
List of Media Components	O _C	Media components are added/removed to/from this list as requested by the user
Start of a Media component	O _C	Time stamp reflecting the start of a Media Component (If SIP is used this corresponds to the time stamp the respective 200 OK message is received). This data item is not carried forward into subsequent partial CDRs, it is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was installed.
Stop of a Media component	O _C	Time stamp reflecting either the stop of a Media Component (If SIP is used this corresponds either to the time stamp the respective 200 OK message or the SIP Bye message is received in the node generating the CDR). This data item is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was removed from a session.
Media Component Initiator	O _M	This item points to the user initiating the media component
Media Type	O _M	Holds the SDP data for a media component in case of a SIP session, the SDP data comprises: codec description (refer to IETF RFC 1890 [7])
Correlation vector access part	O _M	If GPRS is used to access IMS this item holds the GPRS charging ID and GGSN address for the associated PDP context
Record Extensions	O _C	A set of operator/manufacture specific extensions to the record, conditioned upon existence of an extension.

5.2.3.2 Charging Data in P-CSCF (P-CSCF-CDR)

Table 5.2.246: P-CSCF Charging Data (P-CSCF-CDR)

Field	Category	Description
Record Type	O _M	Identifies the type of record: P-CSCF-CDR.
Event Type	O _M	Reflects the type of chargeable telecommunication service/event for which the CDR is generated, such as: "session", "register/subscribe", ...

Field	Category	Description
Role of node	O _M	Specifies the role of the CSCF if relevant for the chargeable telecommunication service/event, which is either: originating role (serving A) terminating role (serving B)
Node Address	O _M	The address of the node providing the information for the CDR
Session ID	O _M	The Session identification. For a SIP session the Session ID contains the SIP Call ID as defined in the Session Initiating Protocol.
Calling Party Address (Public User ID)	O _M	The address of the party initiating a session (SIP URL, E.164 ...)
Called Party Address (Public User ID)	O _M	The address of the party to whom a session is established (SIP URL, E.164 ...)
Served party IP Address	O _M	Holds the IP address of either the calling or called party, depending on whether the proxy is in touch with the calling or the called party
Record opening time	O _M	Time stamp reflecting either the start of a session when the Session is activated in the CSCF (If SIP is used this corresponds to the SIP 200 OK message received in the node generating the CDR) or the partial record opening time for subsequent partial records.
Record closure time	O _M	Time stamp reflecting the stop of a session when the Session is released in the CSCF (last record) (If SIP is used this corresponds to the SIP Bye message received in the node generating the CDR) or record closure time for partial records.
Cause	O _M	Cause for session disconnection
Inter Operator Identifier	O _C	Holds the identification of the network neighbour (incoming or outgoing) as exchanged via SIP signalling.
Local Sequence Number	O _M	Contains a unique record number created by this node.
Record Sequence Number	O _C	Number of partial record if applicable.
Record Output Reason	O _M	Identifies the reason for CDR output, such as: timeLimit, serviceChange (e.g. change in media components), networkInternalReasons, lastCDR, timeChange
Correlation Vector IMS part (ICID)	O _M	Holds the ICID as either received via incoming SIP signalling or generated in the node generating the CDR
List of Media Components	O _C	Media components are added/removed to/from this list as requested by the user
Start of a Media component	O _C	Time stamp reflecting the start of a Media Component (If SIP is used this corresponds to the time stamp the respective 200 OK message is received). This data item is not carried forward into subsequent partial CDRs, it is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was installed.

Field		Category	Description
	Stop of a Media component	O _C	Time stamp reflecting either the stop of a Media Component (If SIP is used this corresponds either to the time stamp the respective 200 OK message or the SIP Bye message is received in the node generating the CDR). This data item is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was removed from a session.
	Media Component Initiator	O _M	This item points to the user initiating the media component
	Media Type	O _M	Holds the SDP data for a media component in case of a SIP session, the SDP data comprises: codec description (refer to IETF RFC 1890 [7])
	Correlation vector access part	O _M	If GPRS is used to access IMS this item holds the GPRS charging ID and GGSN address for the associated PDP context
	Authorised QoS	O _c	Authorised QoS as defined in TS 23.207 [8] / TS 29.207 [9] and applied via the Go interface
Record Extensions		O _C	A set of operator/manufacture specific extensions to the record, conditioned upon existence of an extension.

5.2.3.3 Charging Data in I-CSCF (I-CSCF-CDR)

Table 5.2.375: I-CSCF Charging Data (I-CSCF-CDR)

Field	Category	Description
Record Type	O _M	Identifies the type of record: I-CSCF-CDR.
Event Type	O _M	Reflects the type of chargeable telecommunication service/event for which the CDR is generated, such as: "session", "register/subscribe", ...
Node Address	O _M	The address of the node providing the information for the CDR
Session ID	O _M	The Session identification. For a SIP session the Session ID contains the SIP Call ID as defined in the Session Initiating Protocol.
Calling Party Address (Public User ID)	O _M	The address of the party initiating a session (SIP URL, E.164 ...)
Called Party Address (Public User ID)	O _M	The address of the party to whom a session is established (SIP URL, E.164 ...)
Transaction time stamp	O _M	Time stamp reflecting the time for transaction termination (Upon receiving/generating the final response for the SIP request)
Cause	O _M	Cause for session disconnection
Inter Operator Identifier	O _c	Holds the identification of the network neighbour (incoming) as exchanged via SIP signalling.
Local Sequence Number	O _M	Contains a unique record number created by this node.

Field	Category	Description
S-CSCF information	O _C	Information related to the serving CSCF, e.g. the S-CSCF capabilities upon registration event or the S-CSCF address upon the session establishment event.
Correlation Vector IMS part (ICID)	O _M	Holds the ICID as either received via incoming SIP signalling or generated in the node generating the CDR
Record Extensions	O _C	A set of operator/manufacturer specific extensions to the record, conditioned upon existence of an extension.

5.2.3.4 Charging Data in MRFC (MRFC-CDR)

Table 5.2.486: MRFC Charging Data (MRFC-CDR)

Field	Category	Description
Record Type	O _M	Identifies the type of record: MRFC-CDR.
Node Address	O _M	The address of the node providing the information for the CDR
Session ID	O _M	The Session identification. For a SIP session the Session ID contains the SIP Call ID as defined in the Session Initiating Protocol.
Service ID	O _M	Identifies the service the MRFC is hosting. For conferences the conference ID is used here.
Calling Party Address (Public User ID)	O _M	The address of the party initiating a session (SIP URL, E.164 ...)
Called Party Address (Public User ID)	O _M	The address of the party to whom a session is established (SIP URL, E.164 ...)
Record opening time	O _M	Time stamp reflecting either the start of a session when the Session is activated in the CSCF (If SIP is used this corresponds to the SIP 200 OK message received in the node generating the CDR) or the partial record opening time for subsequent partial records.
Record closure time	O _M	Time stamp reflecting the stop of a session when the Session is released in the CSCF (last record) (If SIP is used this corresponds to the SIP Bye message received in the node generating the CDR) or record closure time for partial records.
Cause	O _M	Cause for session disconnection
Inter Operator Identifier	O _c	Holds the identification of the network neighbour (incoming or outgoing) as exchanged via SIP signalling.
Local Sequence Number	O _M	Contains a unique record number created by this node.
Record Sequence Number	O _C	Number of partial record if applicable.
Record Output Reason	O _M	Identifies the reason for CDR output, such as: timeLimit, serviceChange (e.g. change in media components), networkInternalReasons, lastCDR, timeChange

Field	Category	Description
Correlation Vector IMS part (ICID)	O _M	Holds the ICID as either received via incoming SIP signalling or generated in the node generating the CDR
List of Media Components	O _C	Media components are added/removed to/from this list as requested by the user
Start of a Media component	O _C	Time stamp reflecting the start of a Media Component (If SIP is used this corresponds to the time stamp the respective 200 OK message is received). This data item is not carried forward into subsequent partial CDRs, it is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was installed.
Stop of a Media component	O _C	Time stamp reflecting either the stop of a Media Component (If SIP is used this corresponds either to the time stamp the respective 200 OK message or the SIP Bye message is received in the node generating the CDR). This data item is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was removed from a session.
Media Component Initiator	O _M	This item points to the user initiating the media component
Media Type	O _M	Holds the SDP data for a media component in case of a SIP session, the SDP data comprises: codec description (refer to IETF RFC 1890 [7])
Correlation vector access part	O _M	If GPRS is used to access IMS this item holds a list of the GPRS charging IDs and GGSN addresses for the associated PDP contexts
Record Extensions	O _C	A set of operator/manufacture specific extensions to the record, conditioned upon existence of an extension.

5.2.3.5 Charging Data in MGCF (MGCF-CDR)

Table 5.2.597: MGCF Charging Data (MGCF-CDR)

Field	Category	Description
Record Type	O _M	Identifies the type of record: MGCF-CDR.
Role of node	O _M	Specifies the role of the S-CSCF if relevant for the chargeable telecommunication service/event, which is either: originating role (serving A) terminating role (serving B)
Node Address	O _M	The address of the node providing the information for the CDR
Session ID	O _M	The Session identification. For a SIP session the Session ID contains the SIP Call ID as defined in the Session Initiating Protocol.
Calling Party Address (Public User ID)	O _M	The address of the party initiating a session (SIP URL, E.164 ...)

Field	Category	Description
Called Party Address (Public User ID)	O _M	The address of the party to whom a session is established (SIP URL, E.164 ...)
Record opening time	O _M	Time stamp reflecting either the start of a session when the Session is activated in the CSCF (If SIP is used this corresponds to the SIP 200 OK message received in the node generating the CDR) or the partial record opening time for subsequent partial records.
Record closure time	O _M	Time stamp reflecting the stop of a session when the Session is released in the CSCF (last record) (If SIP is used this corresponds to the SIP Bye message received in the node generating the CDR) or record closure time for partial records.
Cause	O _M	Cause for session disconnection
Inter Operator Identifier	O _C	Holds the identification of the network neighbour (incoming or outgoing) as exchanged via SIP signalling.
Local Sequence Number	O _M	Contains a unique record number created by this node.
Record Sequence Number	O _C	Number of partial record if applicable.
Record Output Reason	O _M	Identifies the reason for CDR output, such as: timeLimit, serviceChange (e.g. change in media components), networkInternalReasons, lastCDR, timeChange
Correlation Vector IMS part (ICID)	O _M	Holds the ICID as either received via incoming SIP signalling or generated in the node generating the CDR
List of Media Components	O _C	Media components are added/removed to/from this list as requested by the user
Start of a Media component	O _C	Time stamp reflecting the start of a Media Component (If SIP is used this corresponds to the time stamp the respective 200 OK message is received). This data item is not carried forward into subsequent partial CDRs, it is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was installed.
Stop of a Media component	O _C	Time stamp reflecting either the stop of a Media Component (If SIP is used this corresponds either to the time stamp the respective 200 OK message or the SIP Bye message is received in the node generating the CDR). This data item is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was removed from a session.
Media Component Initiator	O _M	This item points to the user initiating the media component
Media Type	O _M	Holds the SDP data for a media component in case of a SIP session, the SDP data comprises: codec description (refer to IETF RFC 1890 [7])
Correlation vector access part	O _M	If GPRS is used to access IMS this item holds the GPRS charging ID and GGSN address for the associated PDP context

Field	Category	Description
Record Extensions	O _C	A set of operator/manufacturer specific extensions to the record, conditioned upon existence of an extension.
TrunkGroup ID Incoming/Outgoing	O _M	PSTN leg: Contains the outgoing trunkgroup ID for an outgoing session/call Contains the incoming trunkgroup ID for an incoming session/call
BearerService	O _M	Holds the used bearer service for the PSTN leg

5.2.3.6 Charging Data in BGCF (BGCF-CDR)

Table 5.2.6810: BGCF Charging Data (BGCF-CDR)

Field	Category	Description
Record Type	O _M	Identifies the type of record: BGCF-CDR.
Node Address	O _M	The address of the node providing the information for the CDR
Session ID	O _M	The Session identification. For a SIP session the Session ID contains the SIP Call ID as defined in the Session Initiating Protocol.
Calling Party Address (Public User ID)	O _M	The address of the party initiating a session (SIP URL, E.164 ...)
Called Party Address (Public User ID)	O _M	The address of the party to whom a session is established (SIP URL, E.164 ...)
Record opening time	O _M	Time stamp reflecting either the start of a session when the Session is activated in the CSCF (If SIP is used this corresponds to the SIP 200 OK message received in the node generating the CDR) or the partial record opening time for subsequent partial records.
Record closure time	O _M	Time stamp reflecting the stop of a session when the Session is released in the CSCF (last record) (If SIP is used this corresponds to the SIP Bye message received in the node generating the CDR) or record closure time for partial records.
Cause	O _M	Cause for session disconnection
Inter Operator Identifier	O _C	Holds the identification of the network neighbour (outgoing) as exchanged via SIP signalling. This data item identifies the network where the breakout actually is performed
Local Sequence Number	O _M	Contains a unique record number created by this node.
Record Sequence Number	O _C	Number of partial record if applicable.
Record Output Reason	O _M	Identifies the reason for CDR output, such as: timeLimit, serviceChange (e.g. change in media components), networkInternalReasons, lastCDR, timeChange
Correlation Vector IMS part (ICID)	O _M	Holds the ICID as either received via incoming SIP signalling or generated in the node generating the CDR

Field	Category	Description
List of Media Components	O _C	Media components are added/removed to/from this list as requested by the user
Start of a Media component	O _C	Time stamp reflecting the start of a Media Component (If SIP is used this corresponds to the time stamp the respective 200 OK message is received). This data item is not carried forward into subsequent partial CDRs, it is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was installed.
Stop of a Media component	O _C	Time stamp reflecting either the stop of a Media Component (If SIP is used this corresponds either to the time stamp the respective 200 OK message or the SIP Bye message is received in the node generating the CDR). This data item is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was removed from a session.
Media Component Initiator	O _M	This item points to the user initiating the media component
Media Type	O _M	Holds the SDP data for a media component in case of a SIP session, the SDP data comprises: codec description (refer to IETF RFC 1890 [7])
Correlation vector access part	O _M	If GPRS is used to access IMS this item holds the GPRS charging ID and GGSN address for the associated PDP context
Record Extensions	O _C	A set of operator/manufacture specific extensions to the record, conditioned upon existence of an extension.

5.2.3.7 Charging Data in AS (AS-CDR)

Table 5.119: AS Charging Data (AS-CDR)

Field	Category	Description
Record Type	O _M	Identifies the type of record: AS-CDR.
Node Address	O _M	The address of the node providing the information for the CDR
Session ID	O _M	The Session identification. For a SIP session the Session ID contains the SIP Call ID as defined in the Session Initiating Protocol.
Calling Party Address (Public User ID)	O _M	The address of the party initiating a session (SIP URL, E.164 ...)
Called Party Address (Public User ID)	O _M	The address of the party to whom a session is established (SIP URL, E.164 ...)
Record opening time	O _M	Time stamp reflecting either the start of a session when the Session is activated in the CSCF (If SIP is used this corresponds to the SIP 200 OK message received in the node generating the CDR) or the partial record opening time for subsequent partial records.

<u>Field</u>	<u>Category</u>	<u>Description</u>
<u>Record closure time</u>	O _M	Time stamp reflecting the stop of a session when the Session is released in the CSCF (last record) (If SIP is used this corresponds to the SIP Bye message received in the node generating the CDR) or record closure time for partial records.
<u>Cause</u>	O _M	Cause for session disconnection
<u>Inter Operator Identifier</u>	O _c	Holds the identification of the network neighbour (outgoing) as exchanged via SIP signalling. This data item identifies the network where the breakout actually is performed
<u>Local Sequence Number</u>	O _M	Contains a unique record number created by this node.
<u>Record Sequence Number</u>	O _C	Number of partial record if applicable.
<u>Record Output Reason</u>	O _M	Identifies the reason for CDR output, such as: <u>timeLimit</u> , <u>serviceChange</u> (e.g. change in media components), <u>networkInternalReasons</u> , <u>lastCDR</u> , <u>timeChange</u>
<u>Correlation Vector IMS part (ICID)</u>	O _M	Holds the ICID as either received via incoming SIP signalling or generated in the node generating the CDR
<u>List of Media Components</u>	O _C	Media components are added/removed to/from this list as requested by the user
<u>Start of a Media component</u>	O _C	Time stamp reflecting the start of a Media Component (If SIP is used this corresponds to the time stamp the respective 200 OK message is received). This data item is not carried forward into subsequent partial CDRs, it is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was installed.
<u>Stop of a Media component</u>	O _C	Time stamp reflecting either the stop of a Media Component (If SIP is used this corresponds either to the time stamp the respective 200 OK message or the SIP Bye message is received in the node generating the CDR). This data item is only available in the partial CDR (if partial CDRs are generated) covering the time frame in which the media component was removed from a session.
<u>Media Component Initiator</u>	O _M	This item points to the user initiating the media component
<u>Media Type</u>	O _M	Holds the SDP data for a media component in case of a SIP session, the SDP data comprises: <u>codec description</u> (refer to IETF RFC 1890 [7])
<u>Correlation vector access part</u>	O _M	If GPRS is used to access IMS this item holds the GPRS charging ID and GGSN address for the associated PDP context
<u>Service Specific Data</u>	O _C	Contains service specific data.
<u>Record Extensions</u>	O _C	A set of operator/manufacturer specific extensions to the record, conditioned upon existence of an extension.

5.2.4 CDR Parameter Description

[Editor Note: This is a description of all the fields defined in the tables of ~~section 4~~ [above](#) in alphabetical order. Action Item to Alcatel]

5.2.4.1 Application provided Called Party Address

5.2.4.2 Authorised QoS

5.2.4.3 BearerService

5.2.4.4 Called Party Address (Public User ID)

5.2.4.5 Calling Party Address (Public User ID)

5.2.4.6 Cause (for Record Closure)

5.2.4.7 Correlation vector access part

5.2.4.8 Correlation Vector IMS part (ICID)

5.2.4.9 Event Type

5.2.4.10 Inter Operator Identifier(s)

5.2.4.11 List of Application Servers involved

5.2.4.12 List of Media Components

5.2.4.13 List of Media Components

- 5.2.4.14 Local Sequence Number
- 5.2.4.15 Media Component Initiator
- 5.2.4.16 Media Type
- 5.2.4.17 Private User ID (served party)
- 5.2.4.18 Node Address
- 5.2.4.19 Record Closure Time
- 5.2.4.20 Record Extensions
- 5.2.4.21 Record Opening Time
- 5.2.4.22 Record Output Reason
- 5.2.4.23 Record Sequence Number
- 5.2.4.24 Record Type
- 5.2.4.25 Role of Node
- 5.2.4.26 Session ID
- 5.2.4.27 Served party IP Address

5.2.4.28 Start of a Media component

5.2.4.29 Stop of a Media component

5.2.4.30 TrunkGroup ID Incoming/Outgoing

5.2.5 Bi interface Conventions

[Editor Note: Specify the protocols allowed on Bi: ftp, etc.]

5.2.6 Abstract Syntax Description (tbd)

[Editor Notes: ASN.1 code is needed. Action item for Nortel]

5.2.7 Data Encoding Rules

Data encoding rules are described in Ref [12-10] for BER, in Ref. [13-11] for PER, or in Ref [14-12] for XER.

[Editor Note: Nortel/Alcatel to submit a proposal (i.e., 156) if further clarification is needed.]

6 Online Charging

6.1 Diameter Description on the Ro Interface

6.1.1 Basic Principles

IMS online charging essentially uses the same protocol as is used for offline charging. The online charging protocol, however, may include additional attribute-value pairs (AVPs) within the existing messages, which are not used for offline.

Two cases for online event charging can be distinguished: immediate event charging and event charging with unit reservation. In the case of immediate event charging, granting units to the AS is performed in a single operation that also includes the deduction of the corresponding monetary units from the subscriber's account. In contrast, event charging with unit reservation includes also the process of requesting, reserving and possibly returning units. The deduction of the corresponding monetary units then occurs upon conclusion of the event charging transaction.

The online charging server that is part of the OCS implements the state machine described in [3] for the 'Server, Stateful Accounting' in order to support event charging with unit reservation.

6.1.2 Message Flows and Types

This section describes the message flows for the event charging procedures on the Ro interface.

6.1.2.1 Immediate Event Charging

This section provides the details of the "Debit Unit Request" operation that is specified in 32.200[2].

[Editor's note: Align terminology of the session titles and text with 32.200]

6.1.2.1.1 Message Flows - Successful Cases and Scenarios

6.1.2.1.1.1 Event - Direct Debiting

Figure 6.1 shows the transactions that are required on the Ro interface in order to perform credit control prior to service execution.

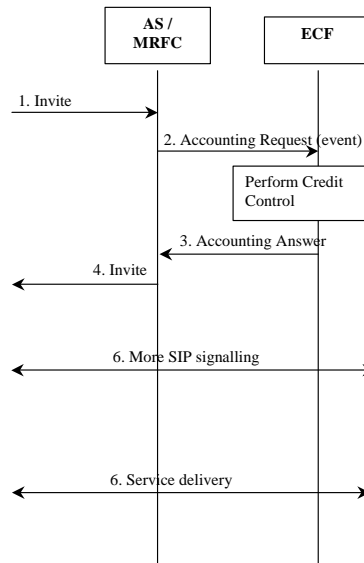


Figure 6.1: Message Sequence Chart for Online Charging Direct Debiting for Session Services

1. The AS/MRFC receives a SIP INVITE request from S-CSCF.
2. The AS/MRFC performs one time event credit control prior to service execution. AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to *EVENT_RECORD* to indicate service specific information to the ECF and include the *ICID* to enable correlation with the related SIP session. The *Requested-Action* shall be set to *DIRECT_DEBITING*. If known, the AS/MRFC may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
3. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF deducts then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly deducts the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *EVENT_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit* and possibly *Cost-Information* indicating the cost of the service are included in the *Accounting-Answer* message).
4. The AS/MRFC sends back the INVITE message to the S-CSCF.
5. More SIP signaling take place to negotiate the media types and codecs between the parties involved in the session as defined in TS 24.228 [13].
6. Service is being delivered.

Note: In addition to the flow described above, the "Debit Unit" operation may alternatively be carried out concurrently with service delivery, or after completion of service delivery.

6.1.2.1.1.2 Event - Direct Debiting for Session-Unrelated Procedures

Figure 6.2 shows the transactions that are required on the Ro interface for one time event without service confirmation, which may be used when there is no need to maintain service state in the ECF. The following example is not thought to be exhaustive; rather it depicts how the mechanism works.

This scenario applies to those cases where service execution is always successful in the service environment. The AS/MRFC must be sure that the requested service event execution will be successful; the exact transaction timing is dependent on the nature and implementation of the service. The one time event without service confirmation may be used for direct debiting where money can be deducted directly from the user's account and confirmation of successful execution is not required.

Only AS/MRFC and ECF are shown in the signaling chart.

[Editor's note: should be further elaborated and aligned with the session unrelated scenario for offline. We need to investigate how to charge for user data that is piggyback on an INVITE. It is probably not relevant here since this section relates only to AS ECF communication.]

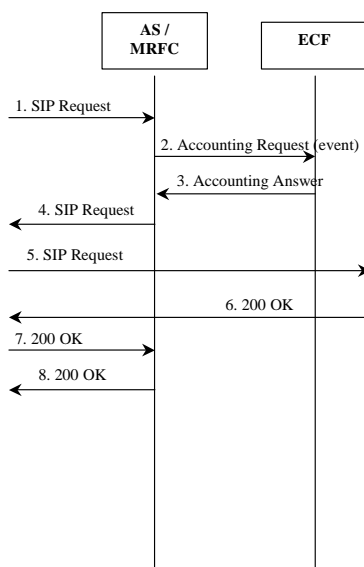


Figure 6.2: Message Sequence Chart for Online Charging – Direct Debiting for Session-Unrelated Procedures

1. The AS/MRFC receives a SIP Request from S-CSCF.
2. The AS/MRFC performs one time event credit control prior to service execution. AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to *EVENT_RECORD* to indicate service specific information to the ECF. The *Requested-Action* shall be set to *DIRECT_DEBITING*. If known, the AS/MRFC may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
3. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF deducts then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly deducts the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *EVENT_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit* and possibly *Cost-Information* indicating the cost of the service are included in the *Accounting-Answer* message).
4. The AS/MRFC sends back the SIP Request to the S-CSCF.

5. The S-CSCF forwards the SIP Request to the destination.
6. The 200OK final response is received from the destination.
7. The 200OK final response is forwarded to the AS/MRFC.
8. The 200OK final response is forwarded back to the S-CSCF.

6.1.2.1.1.3 Transactions for Refund of the User's Account

[Editor's Note: Should be studied in accordance with action items in 32.200]

Figure 6.3 shows the transactions that are required on the Ro interface in order to perform the refund of the end user's account. In the example a subscriber communicates with a top-up application, which refund the user's account with the specified amount of money. The same type of accounting transaction can be used by other services to refund the user's account (e.g. gaming service upon entering a predefined level in the game).

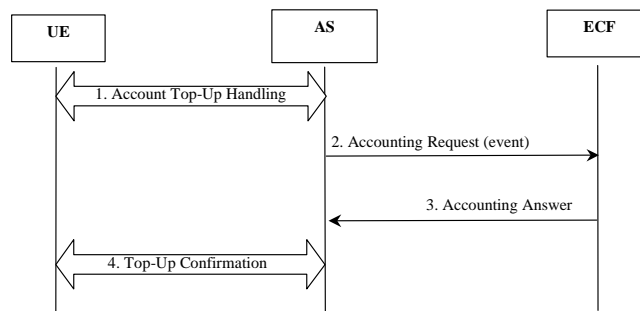


Figure 6.3: Message Sequence Chart for Online Charging Refund Account

1. The subscriber (UE) requests to a top-up application (AS) to refund the account with a certain amount of money.
2. The AS sends *Accounting-Request* with *Accounting-Record-Type* set to *EVENT_RECORD* and the *Requested-Action* set to *REFUND* to ECF. If known, the AS includes the monetary amount to be refunded in the *Requested-Service-Unit*. If the AS does not know the amount of money to be refunded, then the Service specific information will be included in the request.
3. If the monetary amount to be refunded is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF refunds then the corresponding monetary amount to the user's account. If the monetary amount to be refunded is included in the request received from the AS, ECF directly refunds the specified monetary amount to the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *EVENT_RECORD* to the AS. The *Granted-Service-Unit* may be included in the ECF's answer to indicate the refunded units.
4. The top-up application (AS) initiates the confirmation of the successful transaction to the UE.

6.1.2.1.1.4 Transactions for Service Price Enquiry

[Editor's Note: Should be studied in 32.200]

Figure 6.4 shows the transactions that are required on the Ro interface in order to perform the Service price enquiry. In this example, a subscriber communicates with an AS that implements an application to enquiry the price of a requested service. The same types of accounting transaction can be used by any AS to request the price of a service.

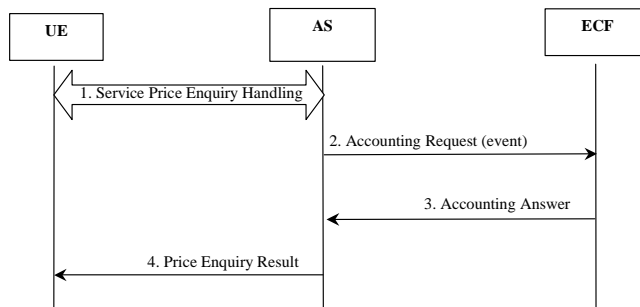


Figure 6.4: Message Sequence Chart for Online Charging Service Price Inquiry

1. The subscriber (UE) requests the price of a determined service(s) to the AS.
2. The AS sends *Accounting-Request* with *Accounting-Record-Type* set to EVENT_RECORD and the *Requested-Action* set to PRICE_ENQUIRY to ECF. The Service specific information is included in the request to indicate the service(s) to be rated.
3. ECF determines the price of the service(s) according to the Service specific information received by issuing a rating request to the Rating Function, but it does not perform any account balance check or credit reservation. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to EVENT_RECORD to the AS. The *Cost-Information* is included in the ECF's answer to indicate the price of the requested service(s).
4. The AS sends the price of the requested service(s) to the UE.

6.1.2.1.1.5 Transactions for Balance Check

[Editor's Note: Should be studied in 32.200]

Figure 6.5 shows the transactions that are required on the Ro interface in order to perform the subscriber's balance check. In the example a subscriber communicates with an AS that implement an application to check if there is enough credit in the subscriber's account to cover the cost of particular service. The same types of accounting transaction can be used by any AS to request the price of a service.

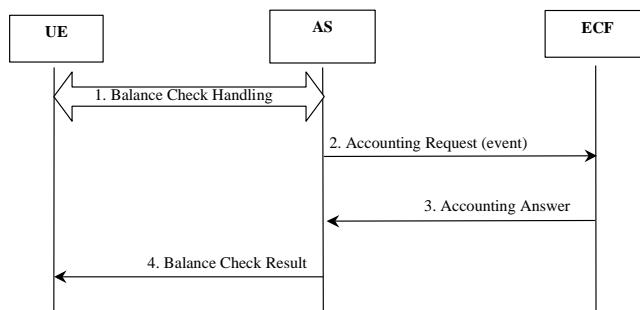


Figure 6.5: Message Sequence Chart for Online Charging Balance Check

1. The subscriber (UE) requests if there is enough credit in the account to cover the cost of a determined service to the AS.

2. The AS sends *Accounting-Request* with *Accounting-Record-Type* set to EVENT_RECORD and the *Requested-Action* set to CHECK_BALANCE to ECF. The Service specific information is included in the request to indicate the service to be rated.
3. ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF performs then balance check to verify if the credit left in the user's account can cover the cost of the rated service without reserving any units. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to EVENT_RECORD to the AS. The *Check-Balance-Result* (ENOUGH_CREDIT/NO_CREDIT) is included in the answer message and the *Cost-Information* may be included to indicate the price of the requested service.
4. The AS sends the balance check result to the UE.

6.1.2.1.2 Message Flows - Error Cases and Scenarios

This section describes various error cases and how these should be handled.

The failure handling behavior should be locally configurable in the credit control client. If *Credit-Control-Failure-Handling* and/or the *Direct-Debiting-Failure-Handling* AVPs are not used, the local configured values will be used.

6.1.2.1.2.1 Reception of SIP Error Messages

In most of the cases a service is considered successfully executed only when the 200OK final answer of the initial SIP request has been received by the AS/MRFC. There are, however, cases when the service is not successful due to SIP error messages received by the AS/MRFC. The following SIP error messages are possible:

4xx, 5xx and 6xx final response received from the terminating side to any SIP request (INVITE, SUBSCRIBE, MESSAGE, etc...)

Note: This also covers the error handling in originating procedures, as a CANCEL request sent by the originating party to cancel a session invitation will trigger the terminating party to send a 487 final response to the INVITE.

6.1.2.1.2.2 ECF Connection Failure

The service node (e.g. AS/MRFC) detects an ECF connection failure when Tx expires or a transport failure is detected as defined in [3]. The Credit Control Client failure handling depends on the use case category.

For Credit Control session:

If the failure occurs during an ongoing credit control session the Credit Control Client will terminate or continue the service depending on the value set for the *Credit-Control-Failure-Handling*. The *Credit-Control-Failure-Handling* AVP may be sent from the authorization server (i.e. UMS) and in the *Accounting-Answer* from the Credit Control Server.

For new credit control session failover to an alternate server should be performed if possible. The *Accounting-Request* with *Accounting-Record-Type* set to START_RECORD should be retransmitted with a new *Session-Id*. If also this new attempt will fail the Credit Control Client will terminate or continue the service depending on the value set for the *Credit-Control-Failure-Handling*.

For one time event:

If the requested action is DIRECT_DEBITING the Credit Control Client should re-send the request message to an alternate server if possible. The Credit Control Client shall mark the re-transmitted request message as possible duplicate with the *Re-Transmission* AVP. If also this new attempt will fail the Credit Control Client will continue the service and will buffer the request or not depending on the value set for the *Direct-Debiting-Failure-Handling*. The *Direct-Debiting-Failure-Handling* AVP may be sent from the authorization server (i.e. UMS). If it is not possible to successfully send the request message to any server and the *Direct-Debiting-Failure-Handling* is set to CONTINUE_AND_BUFFER, the Credit Control Client should store and re-send the message, marked as possible

duplicate, to the server as appropriate. Note that such recovery mechanism does not guarantee that there would be credit left to cover the cost of the service at the time the event will be retransmitted.

If the requested action is REFUND the request message should always be re-sent to an alternate server if possible. The Credit Control Client shall mark the re-transmitted request message as possible duplicate with the *Re-Transmission* AVP. If it is not possible to successfully send the request message to any server the message, marked as possible duplicate, should be stored and re-sent as appropriate.

If the requested action is PRICE_ENQUIRY or CHECK_BALANCE the request message may be forwarded to an alternate server, if possible.

6.1.2.1.2.3 ECF Detected Failure

The ECF detects a failure when the credit control session supervision timer T_s expires [3].

When a failure is detected by the Credit Control Server (i.e. ECF) during an ongoing credit control session, the server shall release the money reservation (i.e. the reserved units are returned to the user's account). The resources reserved in the Credit Control Server for the credit control session shall be released.

6.1.2.1.2.4 Transient, Permanent Failures and Protocol Errors

The service node (e.g. AS/MRFC) detects transient and permanent failures as defined in [3] when:

The *Result-Code* in the accounting answer indicates permanent server failure (50xx)

Errors that fall within permanent failure category are used to inform the peer that the request failed, and should not be attempted again.

The *Result-Code* in the accounting answer indicates transient server failure (40xx)

Errors that fall within transient failure category are used to inform the peer that the request could not be satisfied at the time it was received, but may be able to satisfy the request in the future.

The service node (e.g. AS/MRFC) detects a protocol error as defined in [3] when the *Result-Code* in the accounting answer indicates protocol error (30xx).

A rejected *Accounting-Request* message should cause the user's service to be terminated.

An exception to this rule is the request event message with requested action REFUND. In this latest case, if the server sends back a transient failure or temporary protocol error indicating DIAMETER_OUT_OF_SPACE or DIAMETER_TOO_BUSY an implementation may choose to buffer and retransmit the request message as appropriate. The possible duplicate marking is not needed as the previous request failed in the server.

6.1.2.1.2.5 Duplicate Detection

For credit control sessions the duplicate detection is not needed, as retransmission of accounting request is not allowed.

For one-time events the detection of duplicate request is needed and must be enabled. To speed up and simplify as much as possible the duplicate detection, the all-against-all record checking should be avoided and just the marked potential duplicates need to be checked against other received requests (within a reasonable time window) by the receiver entity.

The service node (e.g. AS/MRFC) shall mark the request messages retransmitted after a link failover as possible duplicates with the *Re-Transmission* AVP, as uniqueness checking against other received requests is mandatory only for those records. The receiver shall do the uniqueness checking of those requests marked with the *Re-Transmission* AVP, against the other requests received within a reasonable time window. The focused check for optimizing the duplicate detection procedure must be based on the inspection of the *Session-Id* and *Accounting-Record-Number* AVP pairs.

For one-time events it has to be noted that the duplicate detection needs to be performed in the Correlation Function. ECF receiving possible duplicate request should mark as possible duplicate the corresponding request sent over Rc interface.

6.1.2.1.2.6 Error Case – Failure in Termination Procedure

Figure 6.6 shows the transactions that are required on the Ro interface in case of failure in termination procedure. This scenario applies to those cases where confirmation of the successful service execution is required in order to deduct money from the user's account and it is also applicable to AS/MRFC originated multimedia services. In the latest case the AS/MRFC will initiate the SIP INVITE and will terminate the SIP error message, upon reception of the SIP error message the AS/MRFC will release the money reservation to the ECF as shown in Figure 6.6. Only AS/MRFC and ECF are shown in the signalling chart.

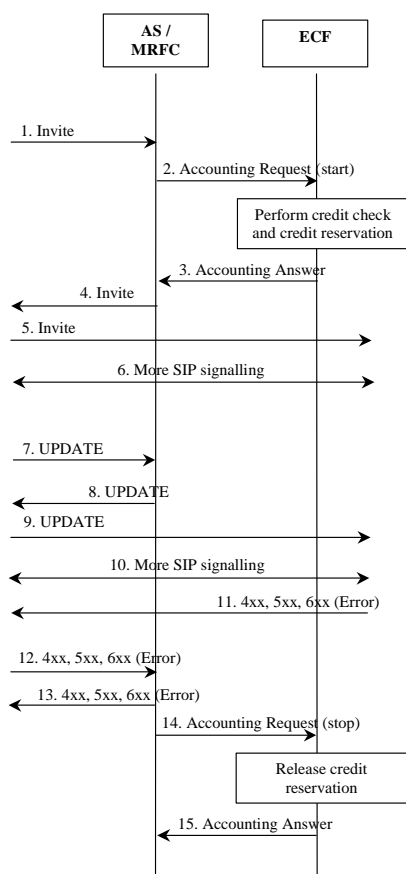


Figure 6.6: Message Sequence Chart for Online Charging – Error in Termination Procedure

- 1 The AS/MRFC receives a SIP INVITE request from S-CSCF.
- 2 The AS/MRFC performs credit control prior to service execution. AS sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF and include the *ICID* to enable correlation with the related SIP session events. If known, the AS may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
- 3 If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication* and

- possibly *Cost-Information* are included in the *Accounting-Answer* message). The AS/MRFC may initiate service execution.
- 4 The AS/MRFC sends back the INVITE message to the S-CSCF.
 - 5 The S-CSCF forwards the INVITE to the destination.
 - 6 More SIP signalling take place to negotiate the media types and codecs between the parties involved in the session as defined in TS 24.228 [13].
 - 7 The AS/MRFC may receives the UPDATE message from S-CSCF when the originating party has successfully reserved the needed resources.
 - 8 The AS/MRFC sends back the UPDATE message to S-CSCF.
 - 9 The S-CSCF forwards the UPDATE message to the destination.
 - 10 More SIP signalling may take place if the optional alerting procedure is performed according to TS 24.228 [13].
 - 11 The termination procedure detected some error situation, and returned a SIP error response (e.g. 4xx).
 - 12 The S-CSCF forwards the SIP error response to the AS/MRFC.
 - 13 The AS/MRFC performs the needed actions to free the resources and send back the SIP error response to the S-CSCF.
 - 14 The AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to STOP-RECORD to indicate unsuccessful service execution. The *Abnormal-Termination-Reason* is included in the request and set to CONNECTION TO END USER BROKEN, *Used-Service-Unit* should be included in the request.
 - 15 The ECF releases the monetary amount reserved in step 3 and acknowledges the reception of the data by sending *Accounting-Response* message with *Accounting-Record-Type* indicating STOP_RECORD.

6.1.2.1.2.7 Error Case – Failure in Origination Procedure

Figure 6.7 shows the transactions that are required on the Ro interface in case of failure in originating procedure. The subscriber that initiated a session either abandoned the attempt, or was unable to obtain the resources necessary for the session.

If the session is aborted due to failure to obtain resources, it will occur right after the UPDATE and steps 10-13 will not be present. If the session is abandoned due to user command, it can happen at any point after the SIP INVITE.

This scenario applies to those cases where confirmation of the successful service execution is required in order to deduct money from the user's account. Only AS/MRFC and ECF are shown in the signalling chart.

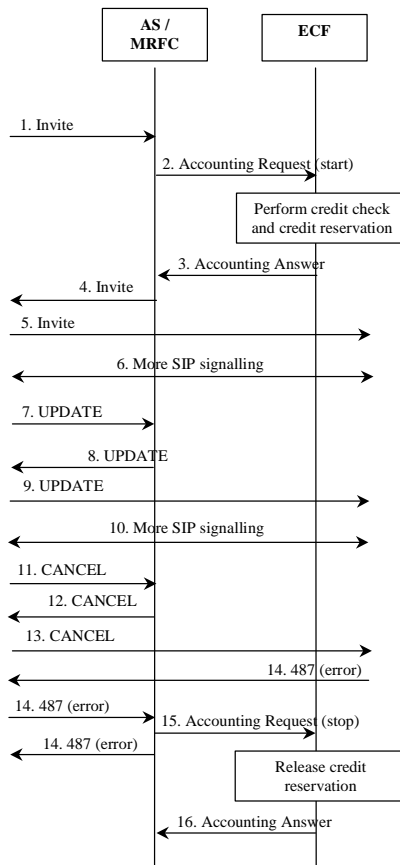


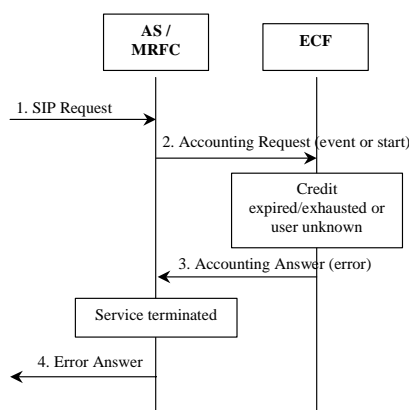
Figure 6.7: Message Sequence Chart for Online Charging – Error in Origination Procedure

- 1 The AS/MRFC receives a SIP INVITE request from S-CSCF.
- 2 The AS/MRFC performs credit control prior to service execution. AS sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF and include the ICID to enable correlation with the related SIP session events. If known, the AS may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
- 3 If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication* and possibly *Cost-Information* are included in the *Accounting-Answer* message). The AS/MRFC may initiate service execution.
- 4 The AS/MRFC sends back the INVITE message to the S-CSCF.
- 5 The S-CSCF forwards the INVITE to the destination.
- 6 More SIP signalling take place to negotiate the media types and codecs between the parties involved in the session as defined in TS 24.228 [13].
- 7 The AS/MRFC may receive the UPDATE message from S-CSCF indicating unsuccessful resource reservation.
- 8 The AS/MRFC sends back the UPDATE message to S-CSCF.
- 9 The S-CSCF forwards the UPDATE message to the destination.

- 10 [More SIP signalling may take place if the optional alerting procedure is performed according to TS 24.228 \[13\].](#)
- 11 [The originator, through the MO procedure, cancelled the original INVITE request.](#)
- 12 [The AS/MRFC send back the CANCEL request to the S-CSCF.](#)
- 13 [The S-CSCF forwards the CANCEL request to the destination.](#)
- 14 [The 487 *Request Terminated* final response is received from the terminating party.](#)
- 15 [The AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to STOP-RECORD to indicate unsuccessful service execution. The *Abnormal-Termination-Reason* is included in the request and set to CONNECTION_TO_END_USER_BROKEN, *Used-Service-Unit* should be included in the request.](#)
- 16 [The ECF releases the monetary amount reserved in step 3 and acknowledges the reception of the data by sending *Accounting-Response* message with *Accounting-Record-Type* indicating STOP_RECORD.](#)

[6.1.2.1.2.8 Error Case – Credit Expired/Exhausted or User Unknown](#)

[Figure 6.8 shows the error handling in case the subscriber's account is expired, exhausted or the user is unknown in the ECF.](#)



[Figure 6.8: Message Sequence Chart for Online Charging – Error Case Credit Expired/Exhausted or User Unknown](#)

- 1 [The AS/MRFC receives a SIP Request from S-CSCF.](#)
- 2 [The AS/MRFC performs one time event credit control or credit check and money reservation prior to service execution. AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to EVENT_RECORD or set to START_RECORD to indicate service specific information to the ECF.](#)
- 3 [The ECF determines that the credit is expired/exhausted or the user is unknown. If the credit is expired or exhausted the ECF returns *Accounting-Answer* with *Result-Code* set to DIAMETER_END_USER_SERVICE_DENIED. If the user is unknown the ECF returns *Accounting-Answer* with *Result-Code* set to DIAMETER_END_USER_NOT_FOUND.](#)
- 4 [The AS/MRFC terminate the service and may return an error answer to the S-CSCF.](#)

6.1.2.2 Event Charging with Reservation

6.1.2.2.1 Message Flows - Successful Cases and Scenarios

6.1.2.2.1.1 Session Related Service – Credit Control and Reservation

Figure 6.9 shows the transactions that are required on the Ro interface in order to perform credit control prior to service execution. This scenario applies to those cases where confirmation of the successful service execution is required in order to deduct money from the user's account. Only AS/MRFC and ECF are shown in the signalling chart.

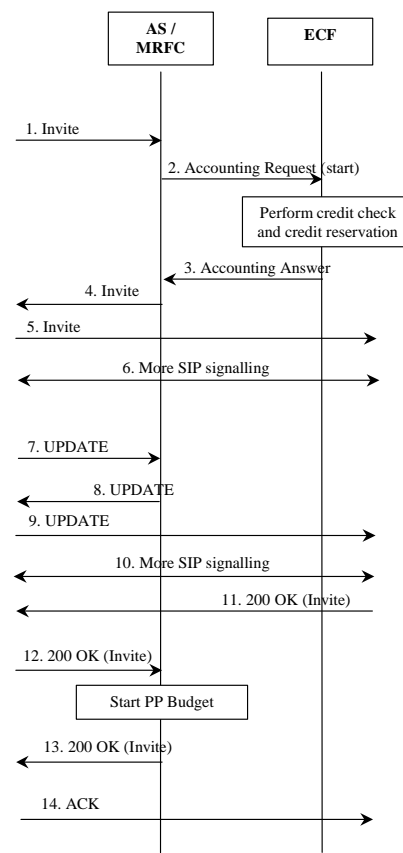


Figure 6.9: Message Sequence Chart for Online Charging Session Establishment

1. The AS/MRFC receives a SIP INVITE request from S-CSCF.
2. The AS/MRFC performs credit control prior to service execution. AS sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF and include the ICID to enable correlation with the related SIP session events. If known, the AS may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
3. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication* and possibly *Cost-Information* are included in the *Accounting-Answer* message). The AS/MRFC may initiate service execution.

4. The AS/MRFC sends back the INVITE message to the S-CSCF.
5. The S-CSCF forwards the INVITE to the destination.
6. More SIP signalling take place to negotiate the media types and codecs between the parties involved in the session as defined in TS 24.228 [13].
7. The AS/MRFC may receive the UPDATE message from S-CSCF when the originating party has successfully reserved the needed resources.
8. The AS/MRFC sends back the UPDATE message to S-CSCF.
9. The S-CSCF forwards the UPDATE message to the destination.
10. More SIP signalling may take place if the optional alerting procedure is performed according to TS 24.228 [13].
11. The 200OK final response is received from the destination.
12. The S-CSCF forwards the 200OK final response to the AS/MRFC, which start decreasing the prepaid budget received in step 3.
13. The AS/MRFC send back the 200OK final response to the S-CSCF.
14. The originating S-CSCF forwards the SIP ACK message to the destination.

6.1.2.2.1.2 Session-Unrelated Service – Credit Check And Reservation

Figure 6.10 shows the transactions that are required on the Ro interface in order to perform credit control prior to execution of a session-unrelated service. This scenario applies to those cases where confirmation of the successful service execution is required in order to deduct money from the user’s account. Only AS/MRFC and ECF are shown in the signalling chart.

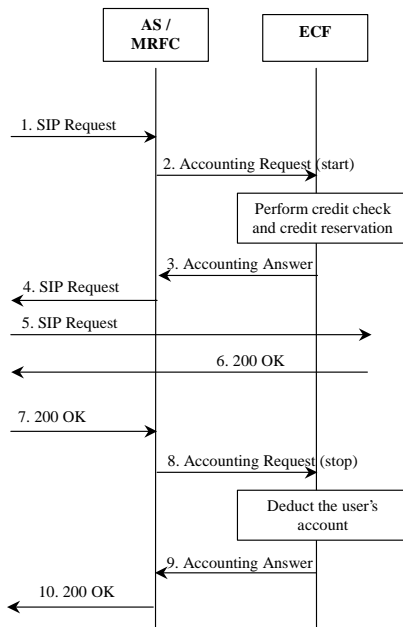


Figure 6.10: Message Sequence Chart for Online Charging Session-Unrelated Service

1. The AS/MRFC receives a SIP Request (e.g. SUBSCRIBE or MESSAGE) from S-CSCF.

2. The AS/MRFC performs credit control prior to service execution. AS sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF. If known, the AS may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
3. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication* and possibly *Cost-Information* are included in the *Accounting-Answer* message). The AS/MRFC may initiate service execution.
4. The AS/MRFC sends back the SIP Request to the S-CSCF.
5. The S-CSCF forwards the SIP Request to the destination.
6. The 200OK final response is received from the destination.
7. The S-CSCF forwards the 200OK final response to the AS/MRFC, which decreases the prepaid budget received in step 3.
8. The AS/MRFC sends an *Accounting-Request* message with *Accounting-Record-Type* set to *STOP_RECORD* to report the *Used-Service-Unit* and successful service transaction execution to the ECF.
9. The ECF deducts the used monetary amount from the user's account. ECF sends *Accounting-Answer* message with *Accounting-Record-Type* set to *STOP_RECORD*. The answer message may include the *Cost-Information* containing the final cost for the service.
10. The AS/MRFC send back the 200OK final response to the S-CSCF.

6.1.2.2.1.3 Mid-Session Procedure – Intermediate Credit Check And Account Deduction

Figure 6.311 shows the transactions that are required on the Ro interface in order to perform intermediate credit check during service execution. The units used since last credit check are deducted from the subscribers account. Only AS/MRFC and ECF are shown in the signalling chart.

When all the granted service units are spent by the service or the interim interval is expired the AS/MRFC shall send a new *Accounting-Request* to the ECF. In case the *Accounting-Interim-Interval* is used it is always up to AS/MRFC to send a new request well in advance before the expiration of the previous request in order to avoid interruption in the service element. Even if the granted service units reserved by the ECF have not been spent upon expiration of the accounting interim interval, the AS/MRFC shall send a new *Accounting-Request* to the ECF. The ECF refunds the reserved credit amount not used to the end user's account and deduct the used monetary amount from the end user's account.

When the used units are reported to the ECF the AS/MRFC will not have any units in its possession before new granted units are received from the ECF. When the new granted units are received from the ECF these units apply from the point where the measurement of the reported used units stopped.

Note that service delivery is ongoing all the time during the session; the accounting interactions shall not cause any discontinuity. The topological discontinuity of the "*Service delivery ongoing*" (big arrows) is performed for figure layout purposes only.

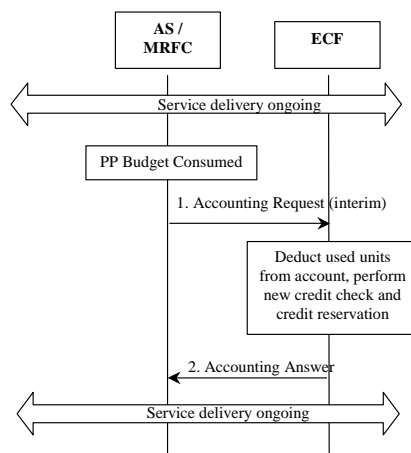


Figure 6.311: Message Sequence Chart for Online Charging Intermediate Credit Check

1. During service delivery, when the prepaid budget has been consumed by the service, the AS/MRFC sends *Accounting-Request* message with *Accounting-Record-Type* set to INTERIM_RECORD to report the *Used-Service-Unit* to the ECF. If known the AS may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
2. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF deducts the used monetary amount from the user's account. ECF sends *Accounting-Answer* message with *Accounting-Record-Type* set to INTERIM_RECORD and includes the *Granted-Service-Unit* (prepaid budget). The answer message may include the *Cost-Information* containing the accumulated cost for the service.

6.1.2.2.1.4 Service Release – ECF initiated

Figure 6.412 shows the transactions that are required on the Ro interface when the subscriber is running out of credit and the service delivery must be stopped. The units used since last credit check are deducted from the subscribers account. Only AS/MRFC and ECF are shown in the signalling chart.

Note that service delivery is ongoing all the time during the session; the accounting interactions shall not cause any discontinuity. The topological discontinuity of the "Service delivery ongoing " (big arrows) is performed for figure layout purposes only.

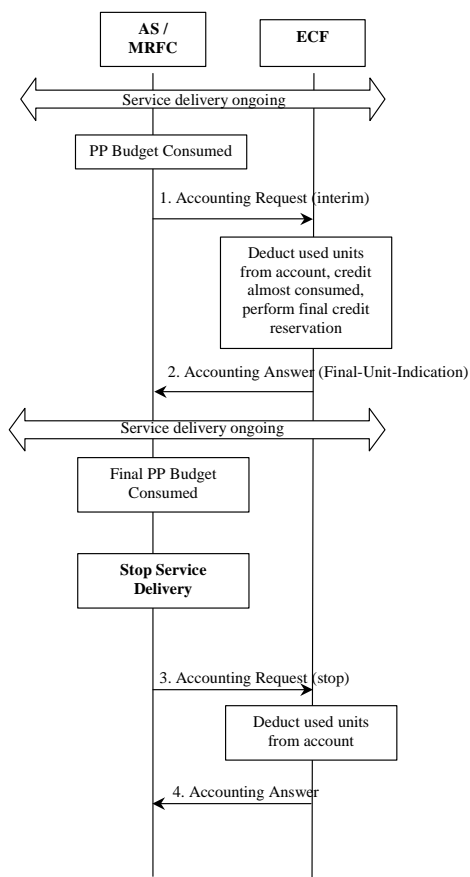


Figure 6.412: Message Sequence Chart for Online Charging ECF Initiated Release

1. During service delivery, when the prepaid budget has been consumed by the service (e.g. Content Server), the AS/MRFC sends *Accounting-Request* message with *Accounting-Record-Type* set to *INTERIM_RECORD* to report the *Used-Service-Unit* to the ECF. If known the AS/MRFC may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
2. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF deducts the used monetary amount from the user's account. ECF sends *Accounting-Answer* message with *Accounting-Record-Type* set to *INTERIM_RECORD* and includes the *Granted-Service-Unit* (prepaid budget) and *Final-Unit-Indication* to indicate the final units for the rendered service. The answer message may include the *Cost-Information* containing the accumulated cost for the service.
3. When the final prepaid budget has been consumed by the service, the AS/MRFC stop the service delivery (e.g. initiate a SIP BYE request) and sends *Accounting-Request* message with *Accounting-Record-Type* set to *STOP_RECORD* to report the *Used-Service-Unit* to the ECF.
4. The ECF deducts the used monetary amount from the user's account. ECF sends *Accounting-Answer* message with *Accounting-Record-Type* set to *STOP_RECORD*. The answer message may include the *Cost-Information* containing the final cost for the service.

6.1.2.2.1.5 Stopping of Service Delivery – Mobile Initiated

Figure 6.135 shows the transactions that are required on the Ro interface when the UE performs the stopping of service delivery. The units used since last credit check are deducted from the subscribers account. Only AS/MRFC and ECF are shown in the signalling chart.

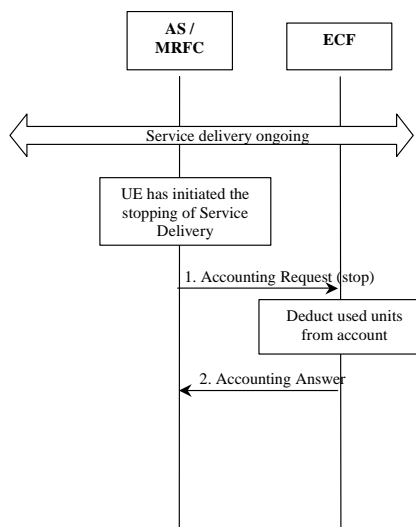


Figure 6.513: Message Sequence Chart for Online Charging Mobile Initiated Release

1. When the subscriber stops service delivery (e.g. initiate a SIP BYE request), AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* indicating *STOP_RECORD* to report back the last *Used-Service-Unit*.
2. The ECF deducts the used monetary amount from the user’s account and returns *Accounting-Response* with *Accounting-Record-Type* indicating *STOP_RECORD*. The answer message may include the *Cost-Information* containing the final cost for the service.

6.1.2.2.1.6 AS/MRFC-Originated Multimedia Service

Figure 6.814 shows the transactions that are required on the Ro interface in order to perform credit control prior to service execution. It is assumed for AS/MRFC-originated multimedia service that confirmation of the successful service execution is always required in order to deduct money from the user’s account. In this example the service establishes a SIP session between AS/MRFC and the end user.

Only AS/MRFC and ECF are shown in the signaling chart.

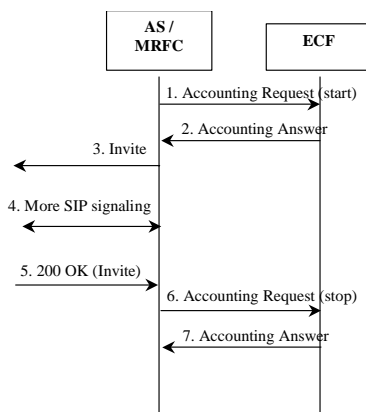


Figure 6.148: Message Sequence Chart for online charging – AS/MRFC Originated Service

1. The AS/MRFC performs credit control prior to service execution. AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF and include the ICID to enable correlation with the related SIP session events. If known the AS/MRFC may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
2. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication* and possibly *Cost-Information* to indicate the cost of the service are included in the *Accounting-Answer* message).
3. The AS/MRFC initiates an INVITE request to the destination.
4. More SIP signaling take place to complete the session set-up as defined in TS 24.228 [13].
5. The AS/MRFC receives 200OK final answer from the destination.
6. The AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to *STOP-RECORD* to indicate successful service execution. *Used-Service-Unit* should be included in the request. Note: the successful execution of a service is dependent on the nature and implementation of the service and may happen upon reception of any SIP message after the INVITE.
7. The ECF commits the money reservation and acknowledges the reception of the data by sending *Accounting-Response* message with *Accounting-Record-Type* indicating *STOP_RECORD*. The answer message may include the *Cost-Information* containing the final cost for the service.

6.1.2.2.1.7 Transactions for Content Charging

Figure 6.915 shows the transactions that are required on the Ro interface in order to charge for content delivery to the end user. For the sake of simplicity the IMS nodes are not shown in the figure, SIP signaling for session establishment take place as defined in TS 24.228 [13].

Note that Content Server may be located within or outside the home network operator domain. For secure transport of Diameter messages see TS 33.201 [4].

..

[Editor note: if Content Server outside home network domain security must be considered. May be an SA3 issue?]

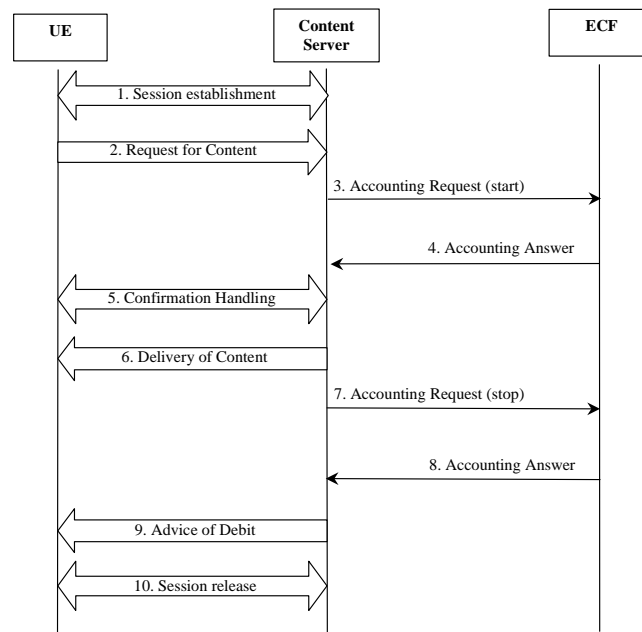


Figure 6.159: Message Sequence Chart for Online Charging Content Charging

1. The session is established as defined in TS 24.228 [13].
2. UE requests the desired content from the content server.
3. The Content Server performs credit control prior to delivering content. Content Server sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF and include the *ICID* to enable correlation with the related SIP session events. The Content Server may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
4. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the Content Server, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication*, and possibly *cost information* are included in the *Accounting-Answer* message).
5. The Content Server may optionally initiate a confirmation handling to the UE.
6. As requested by the UE, the content server delivers the desired content.
7. When the content has been successfully delivered, Content Server sends *Accounting-Request* with *Accounting-Record-Type* set to *STOP-RECORD* to indicate successful service execution. *Used-Service-Unit* should be included in the request.
8. The ECF commits the money reservation and acknowledges the reception of the data by sending *Accounting-Response* message with *Accounting-Record-Type* indicating *STOP_RECORD*. The answer message may include the *Cost-Information* containing the final cost for the service.
9. Optionally, the Content Server may send an advice of debit to the subscriber (UE).
10. The session is released

6.1.2.2.2 Message Flows - Error Cases and Scenarios

This section describes various error cases and how these should be handled.

The failure handling behavior should be locally configurable in the credit control client. If *Credit-Control-Failure-Handling* and/or the *Direct-Debiting-Failure-Handling* AVPs are not used, the local configured values will be used.

6.1.2.2.2.1 Reception Of SIP Error Messages

In most of the cases a service is considered successfully executed only when the 200OK final answer of the initial SIP request has been received by the AS/MRFC. There are, however, cases when the service is not successful due to SIP error messages received by the AS/MRFC. The following SIP error messages are possible:

4xx, 5xx and 6xx final response received from the terminating side to any SIP request (INVITE, SUBSCRIBE, MESSAGE, etc...)

Note: This also covers the error handling in originating procedures, as a CANCEL request sent by the originating party to cancel a session invitation will trigger the terminating party to send a 487 final response to the INVITE.

6.1.2.2.2.2 ECF Connection Failure

The service node (e.g. AS/MRFC) detects an ECF connection failure when Tx expires or a transport failure is detected as defined in [3]. The Credit Control Client failure handling depends on the use case category.

For Credit Control session:

If the failure occurs during an ongoing credit control session the Credit Control Client will terminate or continue the service depending on the value set for the *Credit-Control-Failure-Handling*. The *Credit-Control-Failure-Handling* AVP may be sent from the authorization server (i.e. UMS) and in the *Accounting-Answer* from the Credit Control Server.

For new credit control session failover to an alternate server should be performed if possible. The *Accounting-Request* with *Accounting-Record-Type* set to START_RECORD should be retransmitted with a new *Session-Id*. If also this new attempt will fail the Credit Control Client will terminate or continue the service depending on the value set for the *Credit-Control-Failure-Handling*.

For one time event:

If the requested action is DIRECT_DEBITING the Credit Control Client should re-send the request message to an alternate server if possible. The Credit Control Client shall mark the re-transmitted request message as possible duplicate with the *Re-Transmission* AVP. If also this new attempt will fail the Credit Control Client will continue the service and will buffer the request or not depending on the value set for the *Direct-Debiting-Failure-Handling*. The *Direct-Debiting-Failure-Handling* AVP may be sent from the authorization server (i.e. UMS). If it is not possible to successfully send the request message to any server and the *Direct-Debiting-Failure-Handling* is set to CONTINUE_AND_BUFFER, the Credit Control Client should store and re-send the message, marked as possible duplicate, to the server as appropriate. Note that such recovery mechanism does not guarantee that there would be credit left to cover the cost of the service at the time the event will be retransmitted.

If the requested action is REFUND the request message should always be re-sent to an alternate server if possible. The Credit Control Client shall mark the re-transmitted request message as possible duplicate with the *Re-Transmission* AVP. If it is not possible to successfully send the request message to any server the message, marked as possible duplicate, should be stored and re-sent as appropriate.

If the requested action is PRICE_ENQUIRY or CHECK_BALANCE the request message may be forwarded to an alternate server, if possible.

6.1.2.2.2.3 ECF Detected Failure

The ECF detects a failure when the credit control session supervision timer T_s expires [3].

When a failure is detected by the Credit Control Server (i.e. ECF) during an ongoing credit control session, the server shall release the money reservation (i.e. the reserved units are returned to the user's account). The resources reserved in the Credit Control Server for the credit control session shall be released.

6.1.2.2.2.4 Transient, Permanent Failures and Protocol Errors

The service node (e.g. AS/MRFC) detects transient and permanent failures as defined in [3] when:

The *Result-Code* in the accounting answer indicates permanent server failure (50xx)

Errors that fall within permanent failure category are used to inform the peer that the request failed, and should not be attempted again.

The *Result-Code* in the accounting answer indicates transient server failure (40xx)

Errors that fall within transient failure category are used to inform the peer that the request could not be satisfied at the time it was received, but may be able to satisfy the request in the future.

The service node (e.g. AS/MRFC) detects a protocol error as defined in [3] when the *Result-Code* in the accounting answer indicates protocol error (30xx).

A rejected *Accounting-Request* message should cause the user's service to be terminated.

An exception to this rule is the request event message with requested action REFUND. In this latest case, if the server sends back a transient failure or temporary protocol error indicating *DIAMETER_OUT_OF_SPACE* or *DIAMETER_TOO_BUSY* an implementation may choose to buffer and retransmit the request message as appropriate. The possible duplicate marking is not needed as the previous request failed in the server.

6.1.2.2.2.5 Duplicate Detection

For credit control sessions the duplicate detection is not needed, as retransmission of accounting request is not allowed.

For one-time events the detection of duplicate request is needed and must be enabled. To speed up and simplify as much as possible the duplicate detection, the all-against-all record checking should be avoided and just the marked potential duplicates need to be checked against other received requests (within a reasonable time window) by the receiver entity.

The service node (e.g. AS/MRFC) shall mark the request messages retransmitted after a link failover as possible duplicates with the *Re-Transmission* AVP, as uniqueness checking against other received requests is mandatory only for those records. The receiver shall do the uniqueness checking of those requests marked with the *Re-Transmission* AVP, against the other requests received within a reasonable time window. The focused check for optimizing the duplicate detection procedure must be based on the inspection of the *Session-Id* and *Accounting-Record-Number* AVP pairs.

For one-time events it has to be noted that the duplicate detection needs to be performed in the Correlation Function. ECF receiving possible duplicate request should mark as possible duplicate the corresponding request sent over *Rc* interface.

6.1.2.2.2.6 Error Case – Failure in Termination Procedure

Figure 6.163 shows the transactions that are required on the Ro interface in case of failure in termination procedure. This scenario applies to those cases where confirmation of the successful service execution is required in order to deduct money from the user's account and it is also applicable to AS/MRFC originated multimedia services. In the latest

case the AS/MRFC will initiate the SIP INVITE and will terminate the SIP error message, upon reception of the SIP error message the AS/MRFC will release the money reservation to the ECF as shown in Figure 6.163. Only AS/MRFC and ECF are shown in the signalling chart.

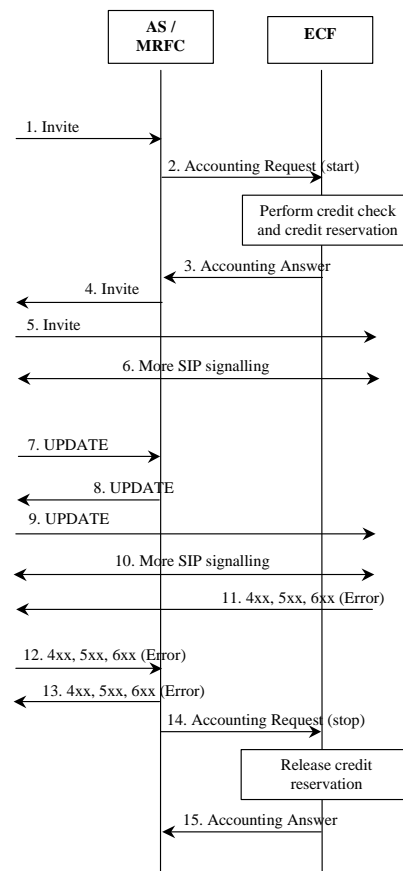


Figure 6.163: Message Sequence Chart for Online Charging – Error in Termination Procedure

1. The AS/MRFC receives a SIP INVITE request from S-CSCF.
2. The AS/MRFC performs credit control prior to service execution. AS sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF and include the ICID to enable correlation with the related SIP session events. If known, the AS may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
3. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication* and possibly *Cost-Information* are included in the *Accounting-Answer* message). The AS/MRFC may initiate service execution.
4. The AS/MRFC sends back the INVITE message to the S-CSCF.
5. The S-CSCF forwards the INVITE to the destination.
6. More SIP signalling take place to negotiate the media types and codecs between the parties involved in the session as defined in TS 24.228 [13].
7. The AS/MRFC may receives the UPDATE message from S-CSCF when the originating party has successfully reserved the needed resources.

8. The AS/MRFC sends back the UPDATE message to S-CSCF.
9. The S-CSCF forwards the UPDATE message to the destination.
10. More SIP signalling may take place if the optional alerting procedure is performed according to TS 24.228 [13].
11. The termination procedure detected some error situation, and returned a SIP error response (e.g. 4xx).
12. The S-CSCF forwards the SIP error response to the AS/MRFC.
13. The AS/MRFC performs the needed actions to free the resources and send back the SIP error response to the S-CSCF.
14. The AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to STOP-RECORD to indicate unsuccessful service execution. The *Abnormal-Termination-Reason* is included in the request and set to CONNECTION_TO_END_USER_BROKEN, *Used-Service-Unit* should be included in the request.
15. The ECF releases the monetary amount reserved in step 3 and acknowledges the reception of the data by sending *Accounting-Response* message with *Accounting-Record-Type* indicating STOP_RECORD.

6.1.2.2.7 Error Case – Failure in Origination Procedure

Figure 6.174 shows the transactions that are required on the Ro interface in case of failure in originating procedure. The subscriber that initiated a session either abandoned the attempt, or was unable to obtain the resources necessary for the session.

If the session is aborted due to failure to obtain resources, it will occur right after the UPDATE and steps 10-13 will not be present. If the session is abandoned due to user command, it can happen at any point after the SIP INVITE.

This scenario applies to those cases where confirmation of the successful service execution is required in order to deduct money from the user’s account. Only AS/MRFC and ECF are shown in the signalling chart.

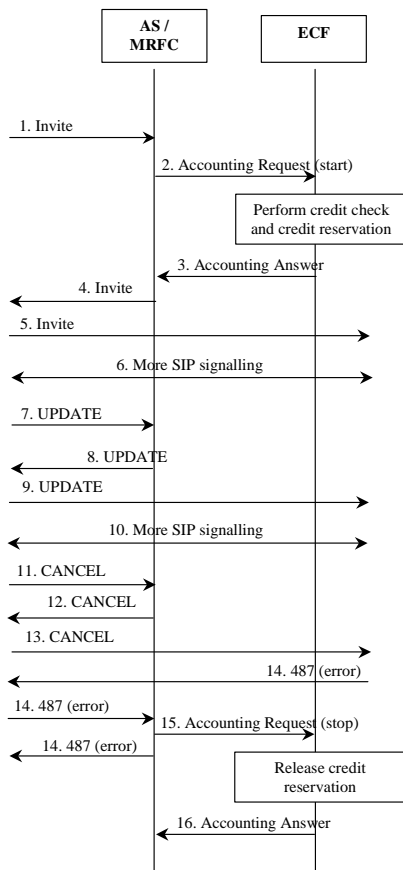


Figure 6.147: Message Sequence Chart for Online Charging – Error in Origination Procedure

1. The AS/MRFC receives a SIP INVITE request from S-CSCF.
2. The AS/MRFC performs credit control prior to service execution. AS sends *Accounting-Request* with *Accounting-Record-Type* set to *START_RECORD* to indicate service specific information to the ECF and include the *ICID* to enable correlation with the related SIP session events. If known, the AS may include *Requested-Service-Unit* (monetary or non monetary value) in the request message.
3. If the service cost information is not received by the ECF, ECF determines the price of the service according to the Service specific information received by issuing a rating request to the Rating Function. ECF reserves then the corresponding monetary amount from the user's account. If the cost of the service is included in the request received from the AS/MRFC, ECF directly reserves the specified monetary amount from the user's account. The ECF returns *Accounting-Answer* message with *Accounting-Record-Type* set to *START_RECORD* to the AS/MRFC in order to authorize the service execution (*Granted-Service-Unit*, *Final-Unit-Indication* and possibly *Cost-Information* are included in the *Accounting-Answer* message). The AS/MRFC may initiate service execution.
4. The AS/MRFC sends back the INVITE message to the S-CSCF.
5. The S-CSCF forwards the INVITE to the destination.
6. More SIP signalling take place to negotiate the media types and codecs between the parties involved in the session as defined in TS 24.228 [13].
7. The AS/MRFC may receive the UPDATE message from S-CSCF indicating unsuccessful resource reservation.
8. The AS/MRFC sends back the UPDATE message to S-CSCF.
9. The S-CSCF forwards the UPDATE message to the destination.
10. More SIP signalling may take place if the optional alerting procedure is performed according to TS 24.228 [13].
11. The originator, through the MO procedure, cancelled the original INVITE request.
12. The AS/MRFC send back the CANCEL request to the S-CSCF.
13. The S-CSCF forwards the CANCEL request to the destination.
14. The 487 *Request Terminated* final response is received from the terminating party.
15. The AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to *STOP-RECORD* to indicate unsuccessful service execution. The *Abnormal-Termination-Reason* is included in the request and set to *CONNECTION_TO_END_USER_BROKEN*, *Used-Service-Unit* should be included in the request.
16. The ECF releases the monetary amount reserved in step 3 and acknowledges the reception of the data by sending *Accounting-Response* message with *Accounting-Record-Type* indicating *STOP_RECORD*.

6.1.2.2.2.83 Error Case – Credit Expired/Exhausted or User Unknown

Figure 6.158 shows the error handling in case the subscriber's account is expired, exhausted or the user is unknown in the ECF.

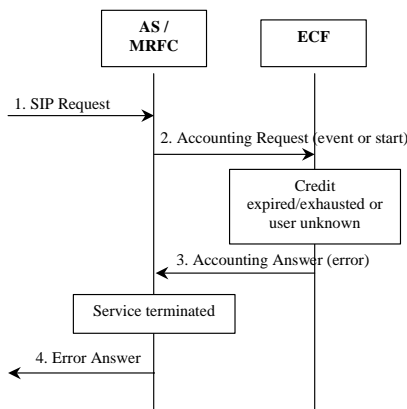


Figure 6.158: Message Sequence Chart for ONLINE CHARGING – Error Case Credit Expired/Exhausted or User Unknown

1. The AS/MRFC receives a SIP Request from S-CSCF.
2. The AS/MRFC performs one time event credit control or credit check and money reservation prior to service execution. AS/MRFC sends *Accounting-Request* with *Accounting-Record-Type* set to *EVENT_RECORD* or set to *START_RECORD* to indicate service specific information to the ECF.
3. The ECF determines that the credit is expired/exhausted or the user is unknown. If the credit is expired or exhausted the ECF returns *Accounting-Answer* with *Result-Code* set to *DIAMETER_END_USER_SERVICE_DENIED*. If the user is unknown the ECF returns *Accounting-Answer* with *Result-Code* set to *DIAMETER_END_USER_NOT_FOUND*.
4. The AS/MRFC terminate the service and may return an error answer to the S-CSCF.

6.1.3 Message Formats

6.1.2.1 Summary of Online Charging Message Formats

The existing Diameter credit control extension internet-draft [14] proposes an approach based on series of “interrogations”:

- initial interrogation (extending the start-session accounting report message)
- zero, one or more interim interrogations (extending the interim accounting report message)
- final interrogation (extending the stop-session accounting report message)

In addition to a series of interrogations, also a one time event (interrogation) can be used e.g. in the case when service execution is always successful.

All of these interrogations make use of the same basic *Accounting-Request* and *Accounting-Answer* messages in the base Diameter protocol. As extensions to the offline charging messages, additional attribute-value pairs (AVPs) shall be introduced to specify a request to reserve credit/service units in the *Accounting-Request* and to report the actual amount reserved in AVPs in the *Accounting-Answer*.

The *Accounting-Request* for the “interim interrogation” and “final interrogation” shall report the actual amount of “units” used, from what was reserved. This will determine an actual amount debited from the subscriber’s account.

Such an approach has the benefit of a common basic message structure and accounting data reporting mechanism. Online charging becomes simply additional AVPs within the existing offline charging protocol.

The table 6.1.4 below describes the use of these messages for online charging.

Table 6.1.4: Ro (Online Charging) Command-Code values

Command-Name	Source	Destination	Abbreviation	Code
Accounting-Request	MRFC, AS	ECF	ACR	271
Accounting –Answer	ECF	MRFC, AS	ACA	271

6.1.2.23.2 Common Structure for the Accounting-Request and Accounting-Answer Message Formats

The following is the basic structure shared by all offline and online charging messages. This is based directly on the *Accounting-Request* and *Accounting-Answer* messages defined in the base Diameter protocol specification.

- Accounting-Request Message

The table below illustrates the basic structure of a Diameter *Accounting-Request* message as it could be used for IMS charging.

IMS Accounting-Request Message Contents
< Diameter Header: 271, REQ, PXY >
< Session-Id > -- Diameter Session Id
[Vendor-Specific-Application-Id]{ Acct-Application-Id } – IMS Charging
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Sub-Session-Id]
[Accounting-RADIUS-Session-Id]
[Accounting-Multi-Session-Id]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
<IMS Accounting Data>
[<IMS Online charging parameters>: only for MRFC and AS messages on the Ro interface.]
* [Proxy-Info]
* [Route-Record]
[(AS) Event Type] Is this needed at all?
[(User/AS) Session ID]
[Calling Party Address (Public User ID)]
[Called Party Address (Public User ID)]
[Time stamp]
[Application provided Called Party Address]
[Inter Operator Identifier(s)]
[Correlation Vector IMS part (ICID)]
* [Media Component]
[Cause]

IMS-specific parts are shown in bold. “*IMS Accounting Data*” actually refers to several Attribute Value Pairs (AVPs), as described in sub-sections of section 7.4.

When this message is used on the Ro interface, the IMS Online Charging parameters include the following:

[Pricing-Enquiry-Indication]
[Abnormal-Termination-Reason]
[Credit-Control-Failure-Handling]
[Direct-Debiting-Failure-Handling]
[Requested-Service-Unit]
[Requested-Action]
[Used-Service-Unit]
* [Service-Parameter-Info]

- Accounting-Answer Message

The table below illustrates the basic structure of a Diameter *Accounting-Answer* message as it could be used for IMS charging.

IMS Accounting-Answer Message Contents
< Diameter Header: 271, PXY >
< Session-Id >
[Vendor-Specific-Application-Id]{ Acct-Application-Id }
{ Result-Code }
{ Origin-Host }
{ Origin-Realm }
{ Accounting-Record-Type }
{ Accounting-Record-Number }
[User-Name]
[Accounting-Sub-Session-Id]
[Accounting-RADIUS-Session-Id]
[Accounting-Realtime-Required]
[Accounting-Multi-Session-Id]
[Error-Reporting-Host]
[Accounting-Interim-Interval]
[Origin-State-Id]
[Re-Transmission]
* [AVP]
* [Proxy-Info]
[Conditional Valid only in online cases<IMS Online charging parameters>: only for MRFC and AS messages on the Ro interface.]
[Editor Note: Verify if Conditional is an accepted DIAMETER term to be used in AVP definition as above]

IMS-specific parts are shown in bold. “*IMS Accounting Data*” actually refers to several Attribute Value Pairs (AVPs), as described in sub-sections of section 7.4.

When this message is used on the Ro interface, the IMS Online charging parameters include the following:

[Check-Balance-Result]
[Cost-Information]
[Credit-Control-Failure-Handling]
[Granted-Service-Unit]
[Cost-Information]
[Final-Unit-Indication]

7 Used AVPs and their Definitions for Offline and Online

7.1 Diameter Base Protocol AVPs (Offline and Online)

Table 7.1 identifies the Attribute Value Pairs (AVPs) utilized by the IMS Charging Application, which are defined in the Diameter Base Protocol [3]. Detailed specification of these AVPs is available in [3]. The table applies to both offline and online charging.

Table 7.1: Diameter Base Protocol AVPs used for both Offline and Online Charging

Attribute Name	AVP Code	Value Type	AVP Flag rules				
			Must	May	Should not	Must not	May Encr.
Accounting-Interim-Interval	482	Unsigned32	M			V	N
Accounting-Realtime-Required	TBD	Enumerated	M			V	N
Accounting-Record-Number	485	Unsigned32	M			V	N
Accounting Record Type	480	Enumerated	M			V	N
Acct-Application-Id	TBD	Unsigned32	M			V	N
Destination-Realm	283	UTF8String	M			V	N
Origin-Host	264	DiameterIdentity	M			V	N
Origin-Realm	296	UTF8String	M			V	N
Origin-State-Id	278	Unsigned32	M			V	N
Proxy-Info	284	Grouped	M			V	N
Route-Record	282	Grouped	M			V	N
Session-Id	263	UTF8String	M			V	N
User-Name	1	UTF8String	M			V	N
Vendor-Specific-Application-Id	260	Grouped	M			V	N

7.1.1 Accounting-Interim-Interval

The *Accounting-Interim-Interval* AVP (AVP Code 482) is of type Unsigned32 and is sent from the CCF to the network element (NE). The NE uses information in this AVP to decide how and when to produce time triggered accounting records.

The following accounting record production behavior is directed by the inclusion of this AVP:

1. The omission of the *Accounting-Interim-Interval* AVP or its inclusion with Value field set to 0 means that no time triggered accounting records are generated.
2. The inclusion of the AVP with Value field set to a non-zero value means that INTERIM_RECORD records MUST be produced between the START_RECORD and STOP_RECORD records. The Value field of this AVP is the nominal interval between these records in seconds. The NE that originates the accounting information MUST produce the first INTERIM_RECORD record roughly at the time when this nominal interval has elapsed from the START_RECORD, the next one again as the interval has elapsed once more, and so on until the session ends and a STOP_RECORD record is produced.

The client MUST ensure that the interim record production times are randomized so that large accounting message storms are not created either among records or around a common service start time.

NOTE: Even if the AVP is missing or indicates value 0, the client must be prepared to send INTERIM_RECORDs based on received SIP messages, as indicated in section 5.1.1.

7.1.2 Accounting-Realtime-Required

The *Accounting-Realtime-Required* AVP (AVP Code 483) is of type Enumerated and is sent from the Diameter home authorization server to the Diameter client or in the *Accounting-Answer* from the accounting server. The client uses

information in this AVP to decide what to do if the sending of accounting records to the accounting server has been temporarily prevented due to, for instance, a network problem.

DELIVER_AND_GRANT 1

The AVP with Value field set to DELIVER_AND_GRANT means that the service **MUST** only be granted as long as there is a connection to an accounting server. Note that the set of alternative accounting servers are treated as one server in this sense. Having to move the accounting record stream to a backup server is not a reason to discontinue the service to the user.

NOTE: This value should not be used for offline charging in 3GPP.

GRANT_AND_STORE 2

The AVP with Value field set to GRANT_AND_STORE means that service **SHOULD** be granted if there is a connection, or as long as records can still be stored as described in Ref. [3]-~~section 9.4.~~

GRANT_AND_LOSE 3

The AVP with Value field set to GRANT_AND_LOSE means that service **SHOULD** be granted even if the records can not be delivered or stored. Upon loss of connection to the accounting server, the records should be stored.

All accounting clients should have a configurable default value.

7.1.3 Accounting-Record-Number

The *Accounting-Record-Number* AVP (AVP Code 485) is of type Unsigned32 and identifies this record within one session. As *Session-Id* AVPs are globally unique, the combination of *Session-Id* and *Accounting-Record-Number* AVPs is also globally unique, and can be used in matching accounting records with confirmations. An easy way to produce unique numbers is to set the value to 0 for records of type EVENT_RECORD and START_RECORD, and set the value to 1 for the first INTERIM_RECORD, 2 for the second, and so on until the value for STOP_RECORD is one more than for the last INTERIM_RECORD.

7.1.4 Accounting-Record-Type

The *Accounting-Record-Type* AVP (AVP Code 480) is of type Enumerated and contains the type of accounting record being sent. The following values are currently defined for the *Accounting-Record-Type* AVP:

EVENT_RECORD 1

An *Accounting Event Record* is used to indicate that a one-time event has occurred (meaning that the start and end of the event are simultaneous). This record contains all information relevant to the service, and is the only record of the service.

START_RECORD 2

An *Accounting Start, Interim, and Stop Records* are used to indicate that a service of a measurable length has been given. An *Accounting Start Record* is used to initiate an accounting session, and contains accounting information that is relevant to the initiation of the session.

INTERIM_RECORD 3

An *Interim Accounting Record* contains cumulative accounting information for an existing accounting session. *Interim Accounting Records* **SHOULD** be sent every time a re-authentication or re-authorization occurs. Further, additional interim record triggers **MAY** be defined by application-specific Diameter applications. The selection of whether to use INTERIM_RECORD records is done by the *Accounting-Interim-Interval* AVP.

STOP_RECORD 4

An Accounting Stop Record is sent to terminate an accounting session and contains cumulative accounting information relevant to the existing session.

7.1.5 Destination-Realm

The Destination-Realm AVP (AVP Code 283) is of type UTF8String, and contains the realm the message is to be routed to. The Destination-Realm AVP SHOULD NOT be present in Answer messages. Diameter Clients insert the realm portion of the User-Name AVP. Diameter servers initiating a request message use the value of the Origin-Realm AVP from a previous message received from the intended target host (unless it is known a priori). When present, the Destination-Realm AVP is used to perform message routing decisions.

Request messages whose ABNF does not list the Destination-Realm AVP as a mandatory AVP are inherently non-routable messages.

This AVP SHOULD be placed as close to the Diameter header as possible.

7.1.6 Origin-Host

The Origin-Host AVP (AVP Code 264) is of type DiameterIdentity, and MUST be present in all Diameter messages. This AVP identifies the endpoint that originated the Diameter message i.e. the address of the node providing the charging data. Relay agents MUST NOT modify this AVP.

The value of the Origin-Host AVP is guaranteed to be unique within a single host. Note that the Origin-Host AVP may resolve to more than one address as the Diameter peer may support more than one address.

This AVP SHOULD be placed as close to the Diameter header as possible

7.1.7 Origin-Realm

The Origin-Realm AVP (AVP Code 296) is of type UTF8String. This AVP contains the Realm of the originator of any Diameter message and MUST be present in all messages.

This AVP SHOULD be placed as close to the Diameter header as possible.

7.1.8 Origin-State-Id

The Origin-State-Id AVP (AVP Code 278), of type Unsigned32, is a monotonically increasing value that is advanced whenever a Diameter entity restarts with loss of previous state, for example upon reboot. Origin-State-Id MAY be included in any Diameter message, including CER.

A Diameter entity issuing this AVP, MUST create a higher value for this AVP each time its state is reset. A Diameter entity MAY set Origin-State-Id to the time of startup, or it MAY use an incrementing counter retained in non-volatile memory across restarts.

The Origin-State-Id, if present, MUST reflect the state of the entity indicated by Origin-Host. If a proxy modifies Origin-Host, it MUST either remove Origin-State-Id or modify it appropriately as well.

Typically, Origin-State-Id is used by an access device that always starts up with no active sessions; that is, any session active prior to restart will have been lost. By including Origin-State-Id in a message, it allows other Diameter entities to infer that sessions associated with a lower Origin-State-Id are no longer active. If an access device does not intend for such inferences to be made, it MUST either not include Origin-State-Id in any message, or set its value to 0.

7.1.9 Proxy-Info

The *Proxy-Info* AVP (AVP Code 284) is of type Grouped. The Grouped Data field has the following ABNF grammar:

```

Proxy-Info ::= < AVP Header: 284 >
_____
_____ { Proxy-Host }
_____
_____ { Proxy-State }
_____
_____ * [ AVP ]

```

7.1.10 Route-Record

The *Route-Record* AVP (AVP Code 282) is of type DiameterIdentity. The identity added in this AVP, **MUST** be the same as the one received in the *Origin-Host* of the *Capabilities Exchange* message.

7.1.11 Session-Id

The *Session-Id* AVP (AVP Code 263) is of type UTF8String and is used to identify a specific session (see section 8). All messages pertaining to a specific session **MUST** include only one *Session-Id* AVP and the same value **MUST** be used throughout the life of a session.

When present, the *Session-Id* **SHOULD** appear immediately following the Diameter Header (see Ref. [3]).

The *Session-Id* **MUST** be globally and eternally unique, as it is meant to uniquely identify a user session without reference to any other information, and may be needed to correlate historical authentication information with accounting information. The *Session-Id* includes a mandatory portion and an implementation-defined portion; a recommended format for the implementation-defined portion is outlined below.

The *Session-Id* **MUST** begin with the sender's identity encoded in the DiameterIdentity type (see Ref [3]). The remainder of the *Session-Id* **MAY** be any sequence that the client can guarantee to be eternally unique; however, the following format is recommended, (square brackets [] indicate an optional element):

<DiameterIdentity>;<high 32 bits>;<low 32 bits>[;<optional value>] <high 32 bits> and <low 32 bits> are decimal representations of the high and low 32 bits of a monotonically increasing 64-bit value. The 64-bit value is rendered in two part to simplify formatting by 32-bit processors. At startup, the high 32 bits of the 64-bit value **MAY** be initialized to the time, and the low 32 bits **MAY** be initialized to zero. This will for practical purposes eliminate the possibility of overlapping Session-Ids after a reboot, assuming the reboot process takes longer than a second. Alternatively, an implementation **MAY** keep track of the increasing value in non-volatile memory.

<optional value> is implementation specific but may include a modem's device Id, a layer 2 address, timestamp, etc.

Example, in which there is no optional value: accesspoint7.acme.com;1876543210;523

Example, in which there is an optional value: accesspoint7.acme.com;1876543210;523;mobile@200.1.1.88

The *Session-Id* is created by the Diameter device that initiates the session, which, in most cases, is done by the client.

Note: The *Session-Id* is not connected to any authorization functionality. [Editor Note: Could it be the same used by Cx?]

7.1.12 User-Name

The *User-Name* AVP (AVP Code 1) [RADIUS] is of type UTF8String, which contains the *User-Name* (i.e. *Private User Identity*), in a format consistent with the NAI specification [NAI].

7.1.13 Vendor-Specific-Application-Id

The *Vendor-Specific-Application-Id* AVP (AVP Code 260) is of type Grouped and is used to advertise support of a vendor-specific Diameter Application. Either the *Auth-Application-Id* or the *Acct-Application-Id* AVP MAY be present. Exactly one of the *Auth-Application-Id* and *Acct-Application-Id* AVPs MAY be present.

This AVP MUST also be present in all vendor-specific commands defined in the vendor-specific application.

This AVP SHOULD be placed as close to the Diameter header as possible.

AVP Format

<Vendor-Specific-Application-Id> ::= < AVP Header: 260 >

1* [Vendor-Id]

0*1{ Auth-Application-Id }

0*1{ Acct-Application-Id }

The 3GPP IMS Charging Application shall use value 10415 (3GPP) as *Vendor-Id*, and value (TBD) as *Acct-Application-Id*.

7.2 Offline Charging AVPs

Table 7.2 identifies the Attribute Value Pairs (AVPs) utilized by the IMS Charging Application for offline charging and are not part of the Diameter Base Protocol [3].

Table 7.2: IMS Specific AVPs used for Offline Charging

Attribute Name	AVP Code	Section defined	Value Type	AVP Flag rules				
				Must	May	Should not	Must not	May Encr.
Re-Transmission	TBD	7.2.1	Unsigned32	M			V	N
(IMS) Event Type	TBD	7.2.2	Grouped	M			V	N
(User) Session ID	TBD	7.2.3	TBD	M			V	N
Application provided Called Party Address	TBD	7.2.4	TBD	M			V	N
Authorised QoS	TBD	7.2.5	TBD	M			V	N
Bearer Service	TBD	7.2.6	TBD	M			V	N
Called Party Address (Public User ID)	TBD	7.2.7	TBD	M			V	N
Calling Party Address (Public User ID)	TBD	7.2.8	TBD	M			V	N
Cause	TBD	7.2.9	TBD	M			V	N
Correlation Vector IMS part (ICID)	TBD	7.2.10	TBD	M			V	N
Inter Operator Identifier	TBD	7.2.11	TBD	M			V	N
List of Application Servers involved	TBD	7.2.12	TBD	M			V	N
Media-Action	TBD	7.2.13	Enumerated	M			V	N
Media Component	TBD	7.2.14	Grouped	M			V	N
Media-Id	TBD	7.2.15	TBD	M			V	N
Media-Initiator	TBD	7.2.16	TBD	M			V	N
Media-Mode	TBD	7.2.17	TBD	M			V	N
Media-Type	TBD	7.2.18	TBD	M			V	N
Node Address	TBD	7.2.19	TBD	M			V	N

Private User ID (served party)	TBD	7.2.20	TBD	M		V	N
Role of node	TBD	7.2.21	Enumerated	M		V	N
S-CSCF information	TBD	7.2.22	TBD	M		V	N
Served party IP Address	TBD	7.2.23	TBD	M		V	N
Service ID	TBD	7.2.24	TBD	M		V	N
SIP-Method	TBD	7.2.25	TBD	M		V	N
Time stamp	TBD	7.2.26	TBD	M		V	N
Start Time stamp	TBD	7.2.27	TBD	M		V	N
Stop Time stamp	TBD	7.2.28	TBD	M		V	N
TrunkGroup ID	TBD	7.2.29	TBD	M		V	N
Incoming/Outgoing							

7.2.1 Re-Transmission

The *Re-Transmission* AVP (AVP code TBD) is of type Unsigned32 and it indicates that the request is a possible duplicate sent right after failover procedure activation, in case of transmission failure.

If included in a command, the value of this AVP is always 1.

7.2.2 (IMS) Event Type

Reflects the type of chargeable telecommunication service/event for which the accounting-request message is generated, such as: "session", "register", "subscribe".

[Editor Note: Should it be Enumrated with a table of possible values?]

The *IMS-Event-Type* AVP is of type Grouped and contains information about the type of chargeable telecommunication service/event for which the accounting-request message is generated.

It has the following ABNF grammar:

```

<IMS-Event-Type> ::= <AVP Header: TBD >
_____ [ SIP-Method ]
_____ [ ... ]

```

7.2.3 (User) Session ID

The *Session-ID* AVP is of type (TBD) and holds the session identifier. For a SIP session the *Session ID* contains the SIP Call ID as defined in the Session Initiating Protocol.

7.2.4 Application provided Called Party Address

The *Application-Provided-Called-Party-Address* is of type (TBD) and holds the called party number if it is determined by an application (SIP URL, E.164).

7.2.5 Authorised QoS

The *Authorised-QoS* AVP is of type (TBD) and holds the Authorised QoS as defined in TS 23.207 [8] /TS 29.207 [9] and applied via the Go interface.

7.2.6 Bearer Service

The *Bearer-Service* AVP is of type (TBD) and holds the used bearer service for the PSTN leg.

7.2.7 Called Party Address (Public User ID)

The *Called-Party-Address* is of type (TBD) and holds the address (Public User ID) of the party to whom a session is established (SIP URL, E.164 ...).

7.2.8 Calling Party Address (Public User ID)

The *Calling-Party-Address* is of type (TBD) and holds the address (Public User ID) of the party initiating a session (SIP URL, E.164 ...).

7.2.9 Cause

The *Cause* AVP is of type (TBD) and specifies the cause for session disconnection, as specified in ...

7.2.10 Correlation Vector IMS part (ICID)

The *Correlation-Vector-IMS-Part* is of type (TBD) and holds the ICID as either received via incoming SIP signalling or generated in the node generating the accounting-request message.

7.2.11 Inter Operator Identifier

The *Inter-Operator-Identifier* AVP is of type (TBD) and holds the identification of the network neighbours (incoming/outgoing) as exchanged via SIP signalling. The format of IOI is specified in ...

7.2.12 List of Application Servers involved

The *List-of-Application-Servers-Involved* AVP is of type (TBD) and holds a list ASs comprising the IP address of the involved AS (if any).

The format may be according to TS 23.218 [5].

7.2.13 Media-Action

The *Media-Action* AVP is of type Enumerated and indicates the type of action applied to the media component.

The identifier can be one of the following:

START

Indicates the start of a new media component.

STOP

Indicates the stop of a media component.

CHANGE

Indicates that an existing media component has been changed.

7.2.14 Media Component

The *Media Component* AVP is of type Grouped and contains information about media used for a IMS session.

It has the following ABNF grammar:

<Media-Component>::=<AVP Header: TBD >

 { Media-Id }
 [Media-Type]
 [Media-Mode]
 [Media-Initiator]
 [Media-Action]
 [GPRS-Charging-Id]
 [GGSN-Address]

7.2.15 Media-Id

The *Media-Id* AVP is of type (TBD) and holds the number of the media description within the SDP data i.e. the first media (m-line) is identified with number '1', the second media with number '2' etc.

7.2.16 Media-Initiator

The *Media-Initiator* AVP is of type Enumerated and indicates which party initiated the media component.

The identifier can be one of the following:

ORIGINATING

indicates that the initial originating party initiated the media component.

TERMINATING

indicates that the initial terminating party initiated the media component.

7.2.17 Media-Mode

The *Media-Mode* AVP is of type TBD and describes the transfer mode of the used media i.e. 'SENDONLY', 'RECVONLY', 'SENDRECV', 'INACTIVE'.

7.2.18 Media-Type

The *Media-Type* AVP is of type TBD and describes the type of media used e.g. 'AUDIO', 'VIDEO' etc.

7.2.19 Node Address

The *Node Address* AVP is of type (TBD) and holds the address of the node providing the information for the accounting data.

7.2.20 Private User ID (served party)

The *Private-User-ID* AVP is of type (TBD) and holds the Private User ID of the served party and is described as a Network Access Identifier according to RFC2486 [6].

7.2.21 Role of node

The *Role-Of-Node* AVP is of type Enumerated and specifies the role of the CSCF, as relevant for the chargeable telecommunication service/event.

The identifier can be one of the following:

ORIGINATING_ROLE

The CSCF is applying a originating role, serving the calling subscriber.

TERMINATING_ROLE

The CSCF is applying a terminating role, serving the called subscriber. Specifies the role of the CSCF if relevant for the chargeable telecommunication service/event, which is either:

→ Originating role (serving A)

→ Terminating role (serving B)

7.2.22 S-CSCF information

The *S-CSCF-Information* AVP is of type (TBD) and holds information related to the serving CSCF, e.g. the S-CSCF capabilities upon registration event or the S-CSCF address upon the session establishment event.

7.2.23 Served party IP Address

The *Served-Party-IP-Address* is of type (TBD) and holds the IP address of either the calling or called party, depending on whether the proxy is in touch with the calling or the called party.

7.2.24 Service ID

The *Service-ID* AVP is of type (TBD) and identifies the service the MRFC is hosting. For conferences the conference ID is used as the value of this parameter.

[Editors note: we probably need to come up with a better description here]

7.2.25 SIP-Method

The *SIP-Method* AVP is of type (TBD) and holds the name of the SIP Method (INVITE, COMET etc.) causing an accounting request to be sent to the CCF.

7.2.26 Time stamp

The *Time-Stamp* AVP is of type (TBD) and holds the time stamp reflecting the time when the *Accounting_Request* is sent to the CCF. Depending on the traffic case, this time stamp can reflect the start of session, start of a media component, stop of session, stop of media component etc.

7.2.27 Start-Time-Stamp

The *Start-Time-Stamp* AVP is of type (TBD). Depending on the traffic case, this time stamp can reflect the start of session, start of a media component, etc. when one time event accounting data transfer is used.

7.2.28 Stop-Time-Stamp

The *Stop-Time-Stamp* AVP is of type (TBD). Depending on the traffic case, this time stamp can reflect the stop of session, stop of a media component, etc. when one time event accounting data transfer is used.

7.2.29 TrunkGroup-ID-Incoming/Outgoing

The *TrunkGroup-ID-Incoming/Outgoing* AVP is of type (TBD) and identifies the incoming and outgoing PSTN leg, as specified in ...

For the PSTN leg:

→ Contains the outgoing trunkgroup ID for an outgoing session/call

→ Contains the incoming trunkgroup ID for an incoming session/call

7.3 Online Charging Specific AVPs

Table 7.3 identifies the Attribute Value Pairs (AVPs) utilized by the IMS Charging Application, which are specific to online charging.

Table 7.3: IMS Specific AVPs used for Online Charging

<u>Attribute Name</u>	<u>AVP Code</u>	<u>Section defined</u>	<u>Value Type</u>	<u>AVP Flag rules</u>				
				<u>Must</u>	<u>May</u>	<u>Should not</u>	<u>Must not</u>	<u>May Encr.</u>
<u>Abnormal Termination Reason</u>	TBD	<u>7.3.1</u>	<u>Enumerated</u>					
<u>Check-Balance-Result</u>	TBD	<u>7.3.2</u>	<u>Enumerated</u>					
<u>Cost</u>	TBD	<u>7.3.3</u>	<u>Float64</u>					
<u>Cost-Information</u>	TBD	<u>7.3.4</u>	<u>Grouped</u>					
<u>Credit-Control-Failure-Handling</u>	TBD	<u>7.3.5</u>	<u>Enumerated</u>					
<u>Currency-Code</u>	TBD	<u>7.3.6</u>	<u>Unsigned32</u>					
<u>Direct-Debiting-Failure-Handling</u>	TBD	<u>7.3.7</u>	<u>Enumerated</u>					
<u>Final-Unit-Indication</u>	TBD	<u>7.3.8</u>	<u>Unsigned32</u>					
<u>Granted-Service-Unit</u>	TBD	<u>7.3.9</u>	<u>Grouped</u>					
<u>Requested-Action</u>	TBD	<u>7.3.10</u>	<u>Enumerated</u>					
<u>Requested-Service-Unit</u>	TBD	<u>7.3.11</u>	<u>Grouped</u>					
<u>Service-Parameter-Info</u>	TBD	<u>7.3.12</u>	<u>Grouped</u>					

Service-Parameter-Type	TBD	7.3.13	Unsigned32					
Service-Parameter-Value	TBD	7.3.14	UTF8String					
Unit-Type	TBD	7.3.15	Enumerated					
Unit-Value	TBD	7.3.16	Float64					
Used-Service-Unit	TBD	7.3.17	Grouped					
Result-Code	268	7.3.18	Unsigned32					

Note: *Result-Code* AVP is defined in Diameter Base Protocol [3], only new values are defined for this application in section 7.3.18.

7.3.1 Abnormal-Termination-Reason AVP

The *Abnormal-Termination-Reason* AVP (AVP Code TBD) is of type Enumerated and contains information about the reason for an abnormal service termination in a service element (e.g AS/MRFC).

The following reasons are defined:

[SERVICE_ELEMENT_TERMINATION](#) 0

[An error occurred in the service element.](#)

[CONNECTION_TO_END-USER_BROKEN](#) 1

[The connection to the end-user is broken.](#)

7.3.2 Check-Balance-Result AVP

The *Check-Balance-Result* AVP (AVP code TBD) is of type Enumerated and contains the result of the balance check. This AVP is applicable only when the *Requested-Action* AVP indicates CHECK_BALANCE in the *Accounting-Request* command.

The following values are defined for the *Check-Balance-Result* AVP.

[ENOUGH_CREDIT](#) 0

[There is enough credit in the account to cover the requested service.](#)

[NO_CREDIT](#) 1

[There is not enough credit in the account to cover the requested service.](#)

7.3.3 Cost AVP

The *Cost* AVP (AVP Code TBD) is of type Float64 and contains the cost estimate of the service in case of price enquiry or the accumulate cost in the case of credit control session.

7.3.4 Cost-Information AVP

The *Cost-Information* AVP (AVP Code TBD) is of type Grouped and is used to return the cost information of a service in the *Accounting-Answer* command. The included *Cost* AVP contains the cost of the service event and the *Currency-Code* specifies in which currency the cost was given.

When the *Requested-Action* AVP with value PRICE_ENQUIRY is included in the *Accounting-Request* command the *Cost-Information* AVP sent in the succeeding *Accounting-Answer* command contains the cost estimation of the requested service, without any reservation being made.

The *Cost-Information* AVP included in the *Accounting-Answer* command with the *Accounting-Record-Type* set to INTERIM_RECORD contains the accumulated cost for the session without taking any credit- reservation into account.

The *Cost-Information* AVP included in the *Accounting-Answer* command with the *Accounting-Record-Type* set to EVENT_RECORD or STOP_RECORD contains the total cost for the requested service. It has the following ABNF grammar

<Cost-Information>: :=<AVP Header: TBD>

{ Cost }

{ Currency-Code }

7.3.5 Credit-Control-Failure-Handling AVP

The *Credit-Control-Failure-Handling* AVP (AVP Code TBD) is of type Enumerated. The Credit Control Client uses information in this AVP to decide what to do if the sending of credit control messages to the Credit Control Server has been temporarily prevented due to, for instance, a network problem.

TERMINATE _____ 0

When the *Credit-Control-Failure-Handling* AVP is set to TERMINATE the service shall only be granted as long as there is a connection to the Credit Control Server. If the Credit Control Client does not receive any *Accounting-Answer* message within the Tx timer the transport connection is regarded failed. The moving of already started credit control session to alternative server is not allowed.

This is the default behaviour if the AVP isn't included in the reply from the authorization server, from credit control server or the local configuration is not available.

CONTINUE _____ 1

When the *Credit-Control-Failure-Handling* AVP is set to CONTINUE the service should be granted even if credit control messages can't be delivered.

7.3.6 Currency-Code AVP

The *Currency-Code* AVP (AVP Code TBD) is of type Unsigned32 and contains a currency code that specifies in which currency the values of AVPs containing monetary units were given. It is specified using the numeric values defined in the ISO 4217 standard.

7.3.7 Direct-Debiting-Failure-Handling AVP

When the Credit Control Server receives and successfully process one time event with requested action *DIRECT_DEBITING*, it deducts the costs of the service directly from the user's account. If the Credit Control Client does not receive the accounting answer within Tx timer, it cannot be sure that money has not been deducted from the user's account. Termination of the service in these conditions is not suitable as it may happen that service is not granted but still the user is charged, which is not legal in many countries.

The *Direct-Debiting-Failure-Handling* AVP (AVP Code TBD) is of type Enumerated. The Credit Control Client uses information in this AVP to decide what to do if the sending of credit control messages to the Credit Control Server has been temporarily prevented due to, for instance, a network problem.

CONTINUE AND BUFFER _____ 0

When the *Direct-Debiting-Failure-Handling* AVP is set to CONTINUE AND BUFFER the service should be granted, the Credit Control Server should store and re-send the failed request, marked as possible duplicate, to the server as appropriate. If the Credit Control Client does not receive any *Accounting-Answer* message within the Tx timer the transport connection is regarded failed. Re-sending of request messages is allowed.

This is the default behaviour if the AVP isn't included in the reply from the authorization server or local configuration is not available.

CONTINUE _____ 1

When the *Direct-Debiting-Failure-Handling* AVP is set to CONTINUE the service should be granted even if credit control messages can't be delivered.

7.3.8 Final-Unit-Indication AVP

The *Final-Unit-Indication* AVP (AVP Code TBD) is of type Unsigned32 and indicates that an accounting command contains the final units for the service. After these units have expired, the service element (e.g. AS/MRFC) is responsible for terminating the service and sending the STOP_RECORD to the ECF.

If included in a command, the value of this AVP is always 1.

7.3.9 Granted-Service-Unit AVP

The *Granted-Service-Unit* AVP (AVP Code TBD) is of type Grouped and contains the amount of units that the service element (e.g. AS/MRFC) can provide to the end user until the service must be released or the new *Accounting-Request* must be sent. The *Unit-Value* AVP contains the granted *Unit-Value* and the *Unit-Type* AVP defines the type of the unit. If the unit type is money, a *Currency-Code* AVP SHOULD be included.

It has the following ABNF grammar:

```

_____ <Granted-Service-Unit>::=< AVP Header: TBD >
_____ { Unit-Type }
_____ { Unit-Value }
_____ [ Currency-Code ]

```

7.3.10 Requested-Action AVP

The *Requested-Action* AVP (AVP Code TBD) is type of Enumerated and contains the requested action being sent by *Accounting-Request* command where the *Accounting-Record-Type* is set to EVENT_RECORD. The following values are defined for the *Requested-Action* AVP:

DIRECT DEBITING _____ 0

Direct debiting indicates that the request is to decrease the end user's account according to information specified in the *Requested-Service-Unit* AVP and/or *Service-Parameter-Info* AVP. The *Granted-Service Unit* AVP in the *Accounting-Answer* command contains the debited units.

REFUND ACCOUNT _____ 1

Refund account indicates that the request is to increase the end user's account according to information specified in the *Requested-Service-Unit* AVP and/or *Service-Parameter-Info* AVP. The *Granted-Service Unit* AVP in the *Accounting-Answer* command contains the refunded units.

CHECK_BALANCE _____ 2

Check balance indicates that the request is a balance check request. In this case the checking of the account balance is done without any credit reservation from the account. The *Check-Balance-Result* AVP in the *Accounting-Answer* command contains the result of the Balance Check.

PRICE ENQUIRY _____ 3

Price Enquiry indicates that the request is a price enquiry request. In this case neither checking of the account balance nor reservation from the account will be done, only the price of the service will be returned in the *Cost-Information* AVP in the *Accounting-Answer* Command.

7.3.11 Requested-Service-Unit AVP

The *Requested-Service-Unit* AVP (AVP Code TBD) is of type Grouped and contains the amount of requested units specified by the service element (e.g. AS/MRFC). The included *Unit-Value* AVP contains the requested *Unit-Value* and the *Unit-Type* AVP defines the type of the unit. If the unit type is money, a *Currency-Code* AVP SHOULD be included.

It has the following ABNF grammar:

```

_____ <Requested-Service-Unit>::=< AVP Header: TBD >
_____ { Unit-Type }
_____ { Unit-Value }
_____ [ Currency-Code ]

```

7.3.12 Service-Parameter-Info AVP

The *Service-Parameter-Info* AVP (AVP Code TBD) is of type Grouped and contains a service specific information used for price calculation or rating. The *Service-Parameter-Type* AVP defines the service parameter type and the *Service-Parameter-Value* AVP contains the parameter value. The actual contents of these AVPs are not within the scope of this document and SHOULD be defined in additional service specific documentation.

It has the following ABNF grammar:

```

_____ <Service-Parameter-Info>::=< AVP Header: TBD >
_____ { Service-Parameter-Type }
_____ { Service-Parameter-Value }

```

7.3.13 Service-Parameter-Type AVP

The *Service-Parameter-Type* AVP is of type Unsigned32 (AVP Code TBD) and defines the type of the service event specific parameter (e.g. it can be end-user location, contents provider Id, tax percentage). The different parameters and their types are service specific and the meanings of these parameters are defined between the service element and the ECF.

7.3.14 Service-Parameter-Value AVP

The *Service-Parameter-Value* AVP is of type UTF8String (AVP Code TBD) and contains the value of the service parameter type.

7.3.15 Unit-Type AVP

The *Unit-Type* AVP is of type Enumerated (AVP Code TBD) and contains the type of the unit. The unit type can be one of the following:

SERVICE_CREDIT_TIME 0

The unit is of type time, given in seconds.

SERVICE_CREDIT_VOLUME 1

The unit is of type volume, given in kB.

SERVICE_CREDIT_EVENT 2

The unit is of type event, given as a number of events.

SERVICE_CREDIT_MONEY 3

The unit is of type money, given as a monetary value, whose currency SHOULD be specified by the *Currency-Code* AVP.

7.3.16 Unit-Value AVP

The *Unit-Value* AVP is of type Float64 (AVP Code TBD) and contains the granted or used Unit-Value. The value can be time in seconds, volume in kB, number of events or monetary amount depending on the given *Unit-Type*.

If the *Unit-Type* AVP is set to time in the *Accounting-Answer* command, the *Unit-Value* AVP specifies the granted time in seconds (measured from the moment when the services becomes active or from the previous *Answer* command) until a new *Accounting-Request* MUST be sent.

If the *Unit-Type* AVP is set to time in the *Accounting-Request* command, the *Unit-Value* AVP specifies the used time since previous report or time requested by the service element (e.g. AS/MRFC).

If the *Unit-Type* AVP is set to volume in the *Accounting-Answer* command, the *Unit-Value* AVP specifies the granted volume in kB (measured from the moment when the services becomes active or from the previous *Answer* command) until a new *Accounting-Request* MUST be sent. If the *Unit-Type* AVP is set to volume in the *Accounting-Request* command, the *Unit-Value* AVP specifies the used volume since previous report or volume requested by service element (e.g. AS/MRFC).

If the *Unit-Type* AVP is set to event in the *Accounting-Answer* command, the *Unit-Value* AVP specifies the granted number of events (measured from the moment when the service becomes active or from the previous *Answer* command) until a new *Accounting-Request* MUST be sent. If the *Unit-Type* AVP is set to event in the *Accounting-Request* command, the *Unit-Value* AVP specifies the used number of events since previous report or number of events requested by the service element (e.g. AS/MRFC).

If the *Unit-Type* AVP is set to money in the *Accounting-Answer* command, the *Unit-Value* AVP specifies the granted monetary amount, which the end user can use until a new *Accounting-Request* MUST be sent. If the *Unit-Type* AVP is set to money in the *Accounting-Request* command, the *Unit-Value* AVP specifies the used monetary amount since previous report or the monetary amount requested by the service element (e.g. AS/MRFC).

7.3.17 Used-Service-Unit AVP

The *Used-Service-Unit* AVP is of type Grouped AVP (AVP Code TBD) and contains the amount of used units since the previous *Accounting-Answer* command. The included *Unit-Type* AVP defines the type of the unit and the *Unit-Value* AVP contains the used amount. If the unit type is money, a *Currency-Code* AVP SHOULD be included.

It has the following ABNF grammar:

<Used-Service-Unit> ::= < AVP Header: TBD >

_____ { Unit-Type }
_____ { Unit-Value }
_____ [Currency-Code]

7.3.18 Result-Code AVP values

This section defines new *Result-Code* AVP (Diameter Base Protocol [3]) values that must be supported by all Diameter implementations that conform to this specification.

The *Accounting-Answer* message includes the *Result-Code* AVP, which may indicate that an error was present in the *Accounting-Request* message. A rejected *Accounting-Request* message should cause the user's session to be terminated.

Errors that fall within the transient failures category are used to inform a peer that the request could not be satisfied at the time it was received, but MAY be able to satisfy the request in the future.

DIAMETER_END_USER_SERVICE_DENIED 40XX

The ECF denies the service request due to service restrictions or limitations related to the end-user, for example the end-user's account could not cover the requested service.

Errors that fall within permanent failure category are used to inform the peer that the request failed, and should not be attempted again.

DIAMETER_END_USER_NOT_FOUND 50XX

The specified end user could not be found in the ECF.

~~[Editor Note: The section "AVP Definitions", originally assigned as Section 7 can be deleted. All AVPs have been already listed. Those related to offline in Section 5 and those related to online in section 6. No need to repeat]~~

Annex A (informative): Offline Charging for an Event – Mobile Origination with one-time event accounting data transfer

~~[Editor: This flow requires further study. Action Item for Nokia]~~

Figure A.1 shows the transaction that are required between CSCF and CCF in case of one-time event accounting data transfer, there is no need to maintain accounting session state for the generating node and the CCF. The flow does not show the optional I-CSCF(THIG). Only the steps for the S-CSCF are explained below to avoid redundancy. The mechanism illustrated in Figure A.1 is of generic nature, thus it may be applied for other scenarios as well. An implementation may use this mechanism if a non-volatile memory for the safe storage of accounting records is available in the NE.

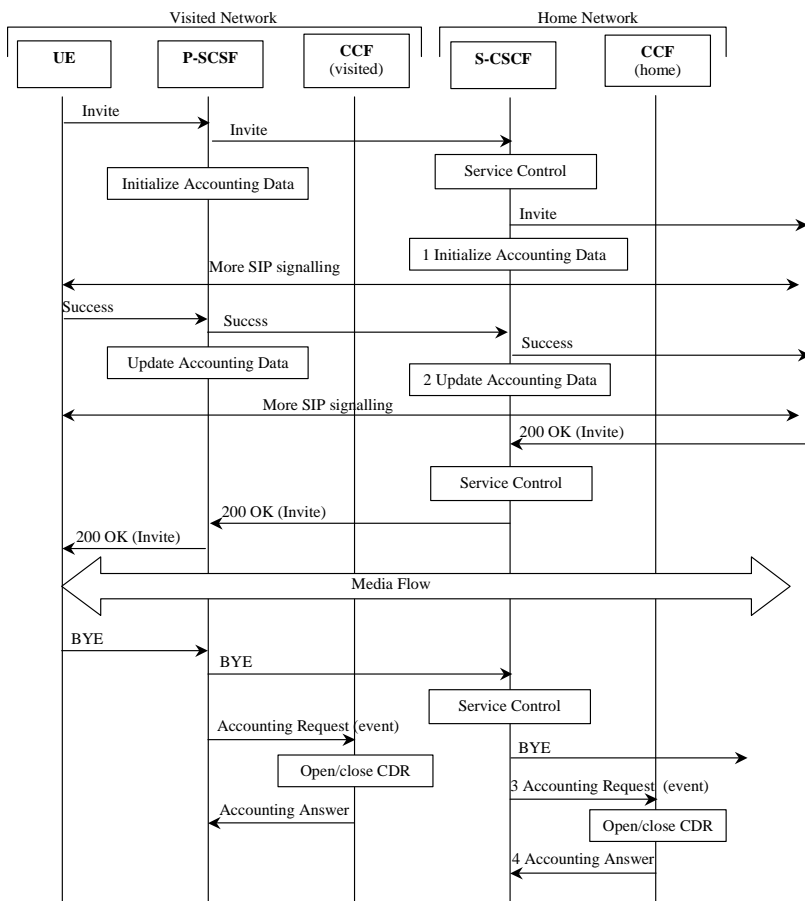


Figure A.1: Message Sequence Chart for Session Establishment (Mobile Origination) with one-time event accounting data transfer

1. The S-CSCF stores the initial accounting information gathered upon the INVITE request.
2. The S-CSCF updates the accounting information stored on Step 1 with information gathered upon the Success request (e.g. final SDP, access part of the correlation vector).
3. The S-CSCF sends an Accounting Request message with Accounting-Record-type set to EVENT-RECORD upon the IMS session is released. This ACR contains the full set of accounting data gathered for the IMS session.
4. The CCF acknowledges the data and creates an S-CSCF CDR.

Annex B (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
Mar 2002	S_15	SP-020033	--	--	Submitted to TSG SA #15 for Information	1.0.0	
Jun 2002	S_16	SP-020327	--	--	Submitted to TSG SA #16 for the 2 nd time for Information	1.5.0	