



A MITEL
PRODUCT
GUIDE

MiVoice MX-ONE

Call Information Logging, Quality Logging

Release 7.7

3/155 19-ANF 901 14 Uen U

May 2024

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by **Mitel Networks Corporation (MITEL[®])**. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos and graphics (collectively "Trademarks") appearing on Mitel's Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC), its affiliates, parents, or subsidiaries (collectively "Mitel") or others. Use of the Trademarks is prohibited without the express consent from Mitel. Please contact our legal department at legal@mitel.com for additional information. For a list of the worldwide Mitel Networks Corporation registered trademarks, please refer to the website: <http://www.mitel.com/trademarks>.

[®], [™] Trademark of Mitel Networks Corporation

© Copyright 2024, Mitel Networks Corporation

All rights reserved

Contents

1 SCOPE.....	1
2 GLOSSARY AND ACRONYMS.....	2
3 GENERAL.....	3
4 OUTPUTS.....	4
4.1 OUTPUT TYPES.....	6
4.2 OUTPUT SUBTYPES.....	7
5 CALL LOGGING DATA PATH.....	8
6 CIL AND QOS RECORD TYPES.....	9
6.1 CALL RECORD.....	9
6.2 MOBILITY RECORD.....	9
6.3 HEARTBEAT RECORD.....	9
7 FIXED OUTPUT FORMATS.....	10
7.1 SQL DATA SCHEMA FOR POSTGRESQL.....	10
7.2 XML OUTPUT FORMAT.....	25
7.3 COMMA-SEPARATED OUTPUT FORMAT.....	30
8 GENERAL FORMAT.....	32
8.1 USING A SCRIPT AS BOTH FORMATTER AND FILTER.....	32
8.1.1 EXAMPLES.....	42
9 PREDEFINED FORMATS.....	46
9.1 USING A SCRIPT AS FILTER.....	46
9.1.1 EXAMPLE.....	55
9.2 HEADER INFORMATION WITH PREDEFINED FORMATS (FP15, MDFP15, ASB501, ASBU MDFP15).....	55
9.3 FP15 FORMAT.....	56
9.4 MDFP15 FORMAT.....	60
9.5 ASB 501 (MD110 STANDARD) FORMAT.....	66

9.6 ASBU MDFP15 FORMAT.....	74
9.7 HEARTBEAT FORMAT.....	79
9.8 DEMO1 FORMAT.....	82
9.9 DEMO2 FORMAT.....	82
9.10 MOBILITY FORMAT.....	83

10 CALL LOGGING RECORD FIELDS..... 84

10.1 CIL DATA FIELDS.....	84
10.1.1 ACCOUNTCODE.....	85
10.1.2 ACCOUNTCODEVALID.....	85
10.1.3 CALLCASEDATA.....	85
10.1.4 CALLINGNUMBER.....	85
10.1.5 CALLINGNUMBERTYPE.....	86
10.1.6 CALLINGNUMBERVALID.....	86
10.1.7 CALLID.....	87
10.1.8 CALLIDASS1.....	87
10.1.9 CALLIDASS2.....	87
10.1.10 CHARGEDNUMBER.....	87
10.1.11 CILCODE.....	88
10.1.12 CILCODEVALID.....	88
10.1.13 CONDITIONCODE.....	88
10.1.14 CONDITIONCODESEG.....	89
10.1.15 CONDITIONCODEPOS.....	90
10.1.16 CONDITIONCODE1CHARACTER, CONDITIONCODE2CHARACTER, CONDITIONCODE3CHARACTER.....	92
10.1.17 CONDITIONCODEUSERDEFINED.....	106
10.1.18 CONNECTEDNUMBER.....	107
10.1.19 CSTCORRDATA.....	107
10.1.20 CURRENTTIME.....	107
10.1.21 DIALEDNUMBER.....	107
10.1.22 DURATION.....	108
10.1.23 ENCRYPTEDMEDIA.....	109
10.1.24 INCTRNKID.....	110
10.1.25 INCTRNKID9.....	110
10.1.26 INCTRNKIDVALID.....	111
10.1.27 OGTRNKID.....	112
10.1.28 OGTRNKID9.....	112
10.1.29 OGTRNKVALID.....	112
10.1.30 OPQUETIME.....	113
10.1.31 OPQUETIMEVALID.....	113
10.1.32 OSACODE.....	113
10.1.33 OSACODEVALID.....	113
10.1.34 QUEUEUETIMECOUNTER.....	114
10.1.35 RINGTIMECOUNTER.....	114
10.1.36 SEQLIM.....	115
10.1.37 SEQNUMBER.....	115
10.1.38 STARTTIME.....	116
10.1.39 STOPTIME.....	116
10.1.40 TAXPULSES.....	116
10.1.41 TAXPULSESVALID.....	116
10.1.42 TIMECOUNTERSVALID.....	116
10.1.43 TNSCODE.....	117
10.1.44 TNSCODEVALID.....	117
10.1.45 TRUNKQUETIME.....	117

10.2 QOS FIELDS.....	117
10.2.1 QOSENDPOINTVALID_X.....	118
10.2.2 RTPADDRESS_X.....	118
10.2.3 ENDPOINTTYPE_X, ENDPOINTTYPENAME_X, ENDPOINTTYPE- DESC_X.....	118
10.2.4 EXTENSION_X.....	121
10.2.5 QOSDATAVALID_X.....	121
10.2.6 ESTIMATEDTHROUGHPUT_X.....	121
10.2.7 FRACTIONLOSTRATE_X.....	121
10.2.8 CODECNAME1_X, CODECNAME2_X, CODECNAME3_X.....	122
10.2.9 CODECTYPE1_X, CODECTYPE2_X, CODECTYPE3_X.....	122
10.2.10 CUMULATIVENUMBEROFPACKETSLOST_X.....	123
10.2.11 MEANESTIMATEDEND2ENDDELAY_X.....	123
10.2.12 MEANJITTER_X.....	123
10.2.13 PACKETLOSTRATE_X.....	123
10.2.14 SIMPLER_VALUE_X.....	123
10.2.15 WORSESTIMATEDEND2ENDDELAY_X.....	124
10.2.16 WORSTJITTER_X.....	124
10.3 MOBILITY LOGGING FIELDS.....	124
10.3.1 CALLINGNUMBER.....	124
10.3.2 EVENTLEVEL.....	124
10.3.3 EQU.....	125
10.3.4 EQU LIM.....	125
10.3.5 EQU MAGAZINE.....	125
10.3.6 EQU BOARD POSITION.....	125
10.3.7 EQU BOARD INDIVIDUAL.....	125
10.3.8 EQU VALID.....	125
10.3.9 ISMOBILE LOGGING.....	125
10.3.10 MOBILITY EVENT.....	126
10.3.11 SEQLIM.....	127
10.3.12 STOPTIME.....	127

11 LOGGING OF UNCOMPLETED CALLS.....128

11.1 LOGGING OF CALLS TO VACANT NUMBER.....	128
11.2 LOGGING OF CALLS TO BUSY PARTY.....	128

12 INTERACTION WITH OTHER FEATURES.....129

12.1 ACCOUNT CODE.....	130
12.1.1 ACCOUNT CODE - DNIS.....	130
12.1.2 ACCOUNT CODE - OPERATOR EXTENDING.....	130
12.1.3 ACCOUNT CODE - CONFERENCE.....	130
12.1.4 ACCOUNT CODE - CALLBACK.....	130
12.1.5 ACCOUNT CODE - LAST NUMBER REDIAL.....	130
12.2 ALTERNATIVE ROUTING.....	130
12.3 AUTHORIZATION CODE.....	131
12.4 ADDITIONAL NUMBER.....	131
12.5 CALLBACK.....	131
12.6 CALL DEFLECTION, SINGLE-STEP TRANSFER.....	131
12.7 CALL METERING.....	131
12.7.1 CALL METERING - TRANSFER OR EXTEND.....	131
12.7.2 CALL METERING - CONFERENCE.....	132
12.7.3 CALL METERING - CHARGED PARTY NUMBER CHANGE.....	132
12.8 CALL WAITING.....	132
12.9 COLLECT CALL.....	132

12.9.1 COLLECT CALL - TRANSFER OR EXTEND.....	132
12.9.2 COLLECT CALL - CONFERENCE.....	133
12.9.3 COLLECT CALL - CALL PICKUP.....	133
12.10 CONFERENCE.....	133
12.10.1 CONFERENCE - CALL WAITING.....	133
12.10.2 CONFERENCE - INTRUSION.....	133
12.11 COORDINATED DIALING PLAN CALLS.....	134
12.12 DIRECT CALL PICKUP.....	134
12.13 DIVERSION.....	134
12.14 DTS.....	134
12.15 DYNAMIC ROUTE ALLOCATION.....	135
12.16 EMERGENCY EXTENSION CALL.....	135
12.17 EXTERNAL FOLLOW-ME, ECF.....	135
12.17.1 ECF - TRANSFER.....	135
12.18 FOLLOW-ME.....	144
12.19 GROUP CALL PICKUP.....	144
12.20 GROUP HUNTING OR CASCADE RING GROUP.....	145
12.21 INTERNAL TRANSFER FROM VOICE TO DATA CALL.....	145
12.22 INTRUSION.....	145
12.23 ISDN CALL BY CALL.....	146
12.24 LEAST COST ROUTING.....	146
12.25 MALICIOUS CALL TRACE.....	146
12.26 MOBILITY EVENTS.....	146
12.26.1 LOCATION REGISTRATION.....	146
12.26.2 DETACH.....	147
12.26.3 HANDOVER.....	147
12.26.4 CALL TO OR FROM CXN.....	147
12.26.5 ABNORMAL CALL RELEASE.....	147
12.27 NETWORK CALLS.....	147
12.28 NETWORK DIVERSION.....	147
12.29 NETWORK REROUTING.....	148
12.30 NETWORK TRANSFER.....	148
12.31 OPERATOR.....	148
12.31.1 OPERATOR - EXTENDING.....	148
12.32 RECALL AS A RESULT OF EXTENDING.....	153
12.33 EXTENDING OF EXTERNAL LINE TO EXTERNAL LINE TRAFFIC.....	155
12.34 SERIAL CALL.....	159
12.35 OPERATOR - INTRUSION.....	159
12.36 OPERATOR QUEUE TIME.....	161
12.37 LONG DURATION CALLS.....	161
12.38 REPEATED INDIVIDUAL DISTRIBUTION (RID).....	161
12.39 ROUTE OPTIMIZATION.....	163
12.40 TRANSFER.....	164

13 ALARM GENERATION.....173

13.1 SETUP OF CALL INFORMATION OUTPUT IS INVALID.....	173
13.2 FAILED WHILE WRITING TO CALL INFORMATION OUTPUT.....	173
13.3 FAILED WHILE CONNECTING CALL INFORMATION OUTPUT.....	173
13.4 CALL INFORMATION OUTPUT QUEUE ALMOST FULL.....	174
13.5 SPEECH QUALITY VALUE AT YELLOW LEVEL.....	174
13.6 SPEECH QUALITY VALUE AT RED LEVEL.....	174

14 ADMINISTRATION.....175

14.1 CALLINFO_STATUS_SET.....	175
14.2 CALLINFO_OUTPUT_SET.....	175
14.3 CALLINFO_STATUS_PRINT.....	176
14.4 CALLINFO_MASK_SET.....	176
14.5 CALLINFO_MASK_PRINT.....	176
14.6 CALLINFO_OUTPUT_CHANGE.....	176
14.7 CALLINFO_FORMAT_PRINT.....	176
14.8 CALLINFO_OUTPUT_INFO.....	176
14.9 CALLINFO_CONDCODE_SET.....	177
14.10 CALLINFO_CONDCODE_PRINT.....	177
14.11 CALLINFO_SQL_TO_FILE.....	177
14.12 CALLINFO_FILE_TO_FILE.....	177
14.13 CALLINFO_FILE_TO_SQL.....	177
14.14 CALLINFO_TCP_PRINT.....	177
14.15 CALLINFO_LIMIT_SET.....	177
14.16 CALLINFO_LIMIT_PRINT.....	178

15 CONFIGURATION..... 179

16 CAPACITY AND LIMITATIONS..... 180

17 PERFORMANCE CHARACTERISTICS..... 181

17.1 LIM ISOLATION.....	181
17.1.1 CIL OR QOS OUTPUT.....	181
17.2 DATA OR PROGRAM RELOAD.....	181
17.3 RESTART.....	181
17.4 OUTPUT DEVICE FAILURE.....	182
17.4.1 CIL AND QOS OUTPUT.....	182
17.4.2 BACK UP CIL AND QOS OUTPUT.....	182

This document provides information on how recorded system data is formatted. It concerns the areas Call Information Logging (CIL) and Quality of Service (QoS) in the MX-ONE.

CIL and QoS enable a customer to analyze and decide how to charge for different kind of calls in order to control the telephone costs.

QoS logging is used to supervise the quality of IP telephone calls and to store data for later analysis to improve the network.

GLOSSARY AND ACRONYMS

2

For a complete list of abbreviations and glossary, see the description for *ACRONYMS, ABBREVIATIONS AND GLOSSARY*.

The call logging function provides collection and storage of data generated in the tele-phony application.

The data can contain logging information of calls, such as internal calls, and incoming and outgoing external calls. Also, calls abandoned during queuing or ringing will be logged. Calls to busy parties and calls to vacant numbers will also be logged.

Calls that are started and completed while call logging is activated will have their data written and stored according to the definition made for each different output.

Output recordings are made on a per call basis. All output types write to buffers, so a small delay might occur.

Each call logging record consists of many elements that describe the call or event.

This document contains descriptions of each data element, and describes the different output formats that can be used to write those data elements.

This chapter contains the following sections:

- [OUTPUT TYPES](#)
- [OUTPUT SUBTYPES](#)

Each data output is assigned a type and subtype that will define the device or endpoint. A maximum of ten output event loggings per LIM may be assigned. All output devices are active and will record data independently. If recording cannot be made on the active output unit, it will continue on the other output units.

Each output can be assigned to an external device, file, or TCP/IP device by I/O command.

All data outputs work independently and can all be active at the same time, writing the same information on several output devices, or using different formatting strings to write selected information on different outputs as desired.

A faulty output unit will issue an alarm so that the fault can be corrected. The data to the faulty unit can be re-created by command, if another output unit has stored the same criteria locally, in a proper format.

The formatting and storing is determined by selecting a type and subtype. The output type generally describes the transport mechanism and the target where the data is written. The subtype describes the formatting rules that apply.

A formatting string may be required by the subtype to produce the output in the desired way.

Depending on the subtype and the format string, the call records are analyzed and screened according to the output criteria. Only call records meeting the defined output criteria will be stored.

When using predefined format subtypes, each output unit is assigned a script, that determines the time format and can be used as a filter to control what to write or discard. With the subtype "general", the "-format" parameter is used both for filtering and formatting.

Table 1: Output Types and Subtypes

sql	postgresql	Stores all (no selection possible) Stores
	none	nothing

OUTPUTS

file	none	Stores nothing
	commaSeparated	Stores all (no selection possible)
	xml	Stores all (no selection possible)
	general fp15	Scripting for formatting and selection
	mdfp15	Scripting for selection
	asb501	Scripting for selection
	asbumdfp15	Scripting for selection
	demo1	Scripting for selection
	demo2	Dect status logging
	mobility	
tcp	commaSeparated	Stores all (no selection possible)
	xml	Stores all (no selection possible)
	general fp15	Scripting for formatting and selection
	mdfp15	Scripting for selection
	asb501	Scripting for selection
	asbumdfp15	Scripting for selection
	demo1	Scripting for selection
	demo2	Dect status logging
	mobility	

v24	commaSeparated	Stores all (no selection possible)
	xml	Stores all (no selection possible)
	general fp15	Scripting for formatting and selection
	mdfp15	Scripting for selection
	asb501	Scripting for selection
	asbumdfp15	Scripting for selection
	demo1	Scripting for selection
	demo2	Dect status logging
	mobility	

4.1 OUTPUT TYPES

The SQL type is used when connecting to an SQL database. It can be an internal or external database.

The file type is stored in the file system, locally or over NFS. The data will be stored as daily files during a week on the hard disk before it is overwritten. Each file should have a unique file name. In addition to the given file name the system will add a day ordinal to the name which reflects the day it was produced, and a file type extension. (example: callData.1.xml callData.2.xml 1= monday, 2= tuesday...).

The tcp type can be used to connect to an external computer with an application that will accept a TCP/IP stream of data in the format written.

The V.24 type can be used for writing to a serial port, a parallel printer port, or to any type of streaming device in the local computer.



Note:

An output using any of the subtypes fp15, mdfp15, asb501, and asbumdfp15 will write a header when opening the output. The end of line option should be set to "crlf000" to be compatible with the ASB 501 04 file format for subtypes fp15, mdfp15, asb501, and asbumdfp15.

For more information on the header fields, [See "Header Information with Predefined Formats \(FP15, MDFP15, ASB501, ASBU MDFP15\)"](#).

4.2 OUTPUT SUBTYPES

The postgresSQL subtype always writes all valid data to the assigned output, and it is assumed that the post-processing will take care of all filtering and removal of unwanted data.

The XML and Comma-separated subtypes also writes all data but in a human readable form that is easy to handle by computers.

The subtypes fp15, mdfp15, asb501, and asbumdfp15 each have a predefined format with fixed fields and preassigned widths defined. If the width of the field is defined larger than the actual width, the unused position is padded with spaces. On the other hand, if the width of the field is defined smaller than the actual width, a truncation will occur. The data elements output by the predefined output formats are described in detail later in this document, [See "FP15 Format"](#), [See "MDFP15 Format"](#), [See "ASB 501 \(MD110 Standard\) Format"](#), and [See "ASBU MDFP15 Format"](#).

The subtype "general" will allow the customer to tailor a script to output virtually any printout format. (Subtypes fp15, mdfp15, asb501, asbumdfp15, demo1, and demo2 are designed this way internally.)

The demo1 and demo2 subtypes are sample formats that can be used, but are mainly made available to demonstrate the potential of the script language, and the ease of use that can be achieved by using a verbose printout.

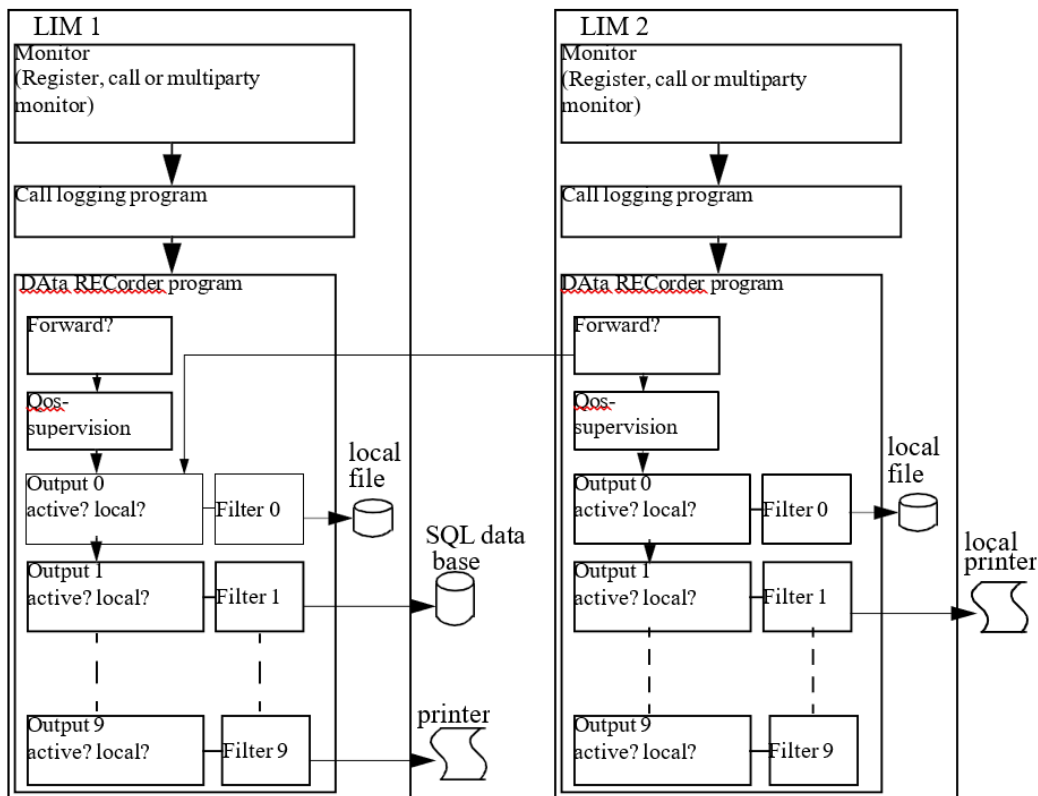
The mobility subtype is used as a tool when tracking DECT portable phones within the system to detect usage and aid in faultfinding.

CALL LOGGING DATA PATH

5

The data from the MX-ONE Service Node passes from the monitors to the call logging program that collects the data. At the end of the call, the data is sent for printout to the data recording program. In the data recording program data will be duplicated and forwarded to other LIMs, if necessary. A test is then made on the TCP/IP data and a QoS supervision is performed. Then all 10 outputs are tested in turn. An active output will write and record the data after filtering and formatting according to the requested type and subtype, as set by command.

The following picture depicts local storage to files, printing central data on a printer in LIM 1, and local data on a printer in LIM2. Central logging is done to an offsite SQL database.



This chapter contains the following sections:

- [CALL RECORD](#)
- [MOBILITY RECORD](#)
- [HEARTBEAT RECORD](#)

The following are CIL and QoS records:

- Call record
- Heartbeat record

6.1 CALL RECORD

The CIL software is capable of generating call records in all the different call record formats, specified by type and subtypes.

6.2 MOBILITY RECORD

The CIL software is capable of generating mobility records in all types and subtypes, however subtypes fp15, mdfp15, asb501, and asbumdfp15 will not provide all data possible and should be avoided.

6.3 HEARTBEAT RECORD

This record type is initiated by command and it is used to aid monitoring the data path for an output.

The record is generated periodically to the ordinary CL output file with a 15 minutes time interval. It is only possible to initiate this facility for the "tcp" and "V24" types.

This chapter contains the following sections:

- [SQL DATA SCHEMA FOR POSTGRESQL](#)
- [XML OUTPUT FORMAT](#)
- [COMMA-SEPARATED OUTPUT FORMAT](#)

There are three fixed formats, the subtypes "postgresql", "xml", and "commaSepa- rated". These formats will write all data to the output and it is not possible to filter or change the output criteria in any way.

7.1 SQL DATA SCHEMA FOR POSTGRESQL

The SQL data schema for postgresql is created when connecting to an SQL database, if no prior database is found. The database is treated as a "write-only" connection from the CIL and QoS application side. Any maintenance to the database (like deletion, backup, and so on) is done from the application or person that is using or reading the data.

```
--  
  
-- Create tables for DAREC i.e. call logging and quality  
  
-- measurements. These tables are automatically created by DAREC  
  
-- This is a printout of the internal definitions  
  
--  
  
-- Tables are created in a way that should speed up normal  
-- searches. As many tables as possible have constant width  
-- columns for faster searching.  
  
-- Frequently searched tables are kept small.  
  
--  
  
-- Applications that read this data might want to create their  
-- own views as a level of indirection. The table definition  
-- might change between versions with no prior notice.  
  
-- Names starting with MD_ are reserved for the DAREC output.
```

```

--

-- Create table for logging host. hostId column from this

-- table ties it together with MD_phoneCall.

--

-- define sequence MD_loggingHostSeq

create sequence MD_loggingHostSeq increment 1 minvalue 1 cycle;

--

-- define table MD_loggingHost create table MD_loggingHost (hostId smallint not null unique primary key,
limNumber smallint not null, hostName varchar(255) not null, exchangeId varchar(255) not null);

--

-- Create table for basic phone call data. id column from this

-- table ties it together with the other tables.

-- Tied to MD_loggingHost with hostId.

-- All dates and times logged are UTC.

--

-- define sequence MD_phoneCallIdSeq

create sequence MD_phoneCallIdSeq increment 1 minvalue 1 cycle;

--

-- define table MD_phoneCall

create table MD_phoneCall ( id bigint not null unique primary key,

hostId smallint not null references MD_loggingHost (hostId)

on delete cascade,

startDate date not null,

startTime time not null,

stopDate date not null,

stopTime time not null,

```

```

duration int not null,

dialedNumber char(20) not null,

connectedNumber char(20) not null,

conditionCode smallint not null,

callCaseData int not null,

chargedNumber char(20) not null,

seqNumber int not null,

seqLim smallint not null,

callId char(24) not null,

callIdAss1 char(24) not null,

callIdAss2 char(24) not null) ;

--

-- Create table for optional calling number [A-number] of phone call

--

-- define table MD_phoneCallCallingNumber

create table MD_phoneCallCallingNumber ( id bigint not null unique primary key references MD_phoneCall
(id) on delete cascade,

callingNumber char(20) not null,

callingNumberType smallint not null ) ;

--

-- Create tables for charge data. Tied to phone call data by

-- id column

--

-- define table MD_chargeData

create table MD_chargeData ( id bigint not null unique primary key references MD_phoneCall (id) on delete
cascade,
```

```

accessCode1 char(6) not null,

accessCode2 char(6) not null,

cstCorrData char(20) not null,

eventLevel smallint not null,

isMobileLogging bool not null);

--

-- define table MD_chargeAccount

create table MD_chargeAccount ( id bigint not null unique primary key references MD_phoneCall (id) on
delete cascade,

accountCode char(10) not null );

--

-- define table MD_chargePulses

create table MD_chargePulses ( id bigint not null unique primary key references MD_phoneCall (id) on
delete cascade,

taxPulses smallint not null );

--

-- define table MD_chargeCil

create table MD_chargeCil ( id bigint not null unique primary key references MD_phoneCall (id) on delete
cascade,

cilCode char(20) not null );

--

-- define table MD_chargeTns

create table MD_chargeTns ( id bigint not null unique primary key references MD_phoneCall (id) on delete
cascade,

tnscode char(4) not null );

--

-- define table MD_chargeOsa

```

```

create table MD_chargeOsa ( id bigint not null unique primary key references MD_phoneCall (id) on delete
cascade,

osacode char(2) not null );

--

-- define table MD_chargeOgTrunk

create table MD_chargeOgTrunk ( id bigint not null unique primary key references MD_phoneCall (id) on
delete cascade,

ogTrnkId char(5) not null, trnkQueTim smallint not null );

--

-- define table MD_chargeIncTrunk

create table MD_chargeIncTrunk ( id bigint not null unique primary key references MD_phoneCall (id) on
delete cascade,

incTrnkId char(10) not null );

--

-- define table MD_chargeOpQueue

create table MD_chargeOpQueue ( id bigint not null unique primary key references MD_phoneCall (id) on
delete cascade,

opQueTim smallint not null );

--

-- define table MD_chargeExtTime

create table MD_chargeExtTime ( id bigint not null unique primary key references MD_phoneCall (id) on
delete cascade,

ringTimeCounter smallint not null, queueTimeCounter smallint not null );

--

-- define table MD_chargeEqu

create table MD_chargeEqu ( id bigint not null unique primary key references MD_phoneCall (id) on delete
cascade,

lim smallint not null, magazine smallint not null, board smallint not null, individual smallint not null );

--

```

```

-- Create table qualityData with IP quality for phone call

-- Tied to phone call data by id column

-- Notice that there might be several quality data records for

-- each phone call - one in each RTP/IP endpoint.

--

-- define table MD_qualityData

create table MD_qualityData ( id bigint not null references MD_phoneCall (id) on delete cascade,

endpoint smallint not null,

worstEstimatedEndToEndDelay int not null,

meanEstimatedEndToEndDelay int not null,

cumNumPacketsLost int not null,

packetLostRate smallint not null,

worstJitter int not null,

estimatedThroughput int not null,

fractionLostRate smallint not null,

meanJitter int not null,

simpleRValue smallint not null,

primary key (id,

endpoint) );

-- simpleRValue range 0 - 10000 (= 0.00% - 100.00%)

--

-- The endpoint in MD_qualityData can have any number of codecs

-- [according to ITU-T H.323 07/2003: 6.2.8.2 "Logical channel

-- signalling" and 6.2.8.6. "Multiplexed stream..."]

-- Most commonly there is 1 [speech] or 2 [speech + video] codecs

```

```

--

-- define table MD_qualityDataCodec

create table MD_qualityDataCodec ( id bigint not null references MD_phoneCall (id) on delete cascade,

endpoint smallint not null,

typeOfCodec char(1) not null,

codec smallint not null );

-- typeOfCodec can have the following values:

-- 'S' = speach

-- 'V' = video

-- 'D' = data

-- 'O' = other

-- 'I' = invalid

--

-- Create table for endpoint type.

-- Names and descriptions of endpoint types.

-- endpointType column from this

-- table ties it together with MD_qualityAddress.

--

-- define table MD_endpointType

create table MD_endpointType (endpointType smallint not null unique primary key,

name varchar(25) not null,

description varchar(255) not null) ;

--

-- Create table MD_qualityAddress with addresses of IP quality for

-- phone call. Tied to phone call data by id column.

```

```

-- Tied to qualityData with id and endpoint column.

--

-- define table MD_qualityAddress

create table MD_qualityAddress ( id bigint not null references MD_phoneCall (id) on delete cascade,

endpoint smallint not null,

endpointType smallint not null

references MD_endpointType, (endpointType),

rtpAddress varchar(54) not null,

extension varchar(255) not null, primary key (id,

endpoint) ) ;

--

-- Data definitions are ready.

-- All tables indexes and sequences defined.

--

-- Create a view that holds the aggregated information about a

-- phone call. Notice that columns in the view can be null if data

-- is not available for a specific call

--

-- define view MD_callView

create view MD_callView as select * from (select * from MD_loggingHost INNER JOIN MD_phoneCall

using (hostId)

) as MD_phoneCallOnHost LEFT OUTER JOIN MD_phoneCallCallingNumber using (id) LEFT OUTER

JOIN MD_chargeData using (id) LEFT OUTER JOIN MD_chargeAccount using (id)

LEFT OUTER JOIN MD_chargePulses using (id)

LEFT OUTER JOIN MD_chargeCil using (id)

LEFT OUTER JOIN MD_chargeTns using (id)

```



```

LEFT OUTER JOIN MD_chargeOsa using (id)

LEFT OUTER JOIN MD_chargeOgTrunk using (id)

LEFT OUTER JOIN MD_chargeIncTrunk using (id)

LEFT OUTER JOIN MD_chargeOpQueue using (id)

LEFT OUTER JOIN MD_chargeExtTime using (id)

LEFT OUTER JOIN MD_chargeEqu using (id);

--

-- define view MD_qosView

create view MD_qosView as select * from MD_callView LEFT OUTER JOIN MD_qualityAddress using (id)
LEFT OUTER JOIN MD_qualityData using (id,endpoint) LEFT OUTER JOIN MD_qualityDataCodec using
(id,

endpoint) ;

--

-- Now create the view for OM.

--

-- define view MD_OM_QosView

create view MD_OM_QosView AS select id, startDate,

startTime,

duration,

callingNumber,

dialedNumber,

connectedNumber,

callId,

endpoint,

endpointType,

MD_endpointType.name AS endpointTypeName,

```

extension,

cumNumPacketsLost,

estimatedThroughput,

meanJitter,

worstJitter,

meanEstimatedEndToEndDelay,

worstEstimatedEndToEndDelay,

packetLostRate,

simpleRValue FROM MD_callView INNER JOIN MD_qualityAddress using (id) INNER JOIN
MD_qualityData using (id,

endpoint) INNER JOIN MD_endpointType using (endpointType);

--

-- All tables and views defined.

-- Now fill table MD_endpointType with data on endpoint types.

--

insert into MD_endpointType (endpointType,

name,

description) values (0,

'a_ep' ,

'A party terminal end point') ;

insert into MD_endpointType (endpointType,

name,

description) values (1, 'a_eemg' ,

'A party. media gateway.') ;

insert into MD_endpointType (endpointType, name,

description) values (2, 'a_gw' ,

```

'A party. other gateway' );

insert into MD_endpointType (endpointType, name,
description) values (3, 'a_multiLim' ,
'A party. Multi LIM' );

insert into MD_endpointType (endpointType, name,
description) values (4, 'a_operator' ,
'A party. operator' );

insert into MD_endpointType (endpointType, name,
description) values (5, 'a_voiceMail' ,
'A party. voice mail' );

insert into MD_endpointType (endpointType, name,
description) values (6, 'a_voiceMenu' ,
'A party. voice menu. ETSI*/ Stored Voice Services' ); insert into MD_endpointType (endpointType,
name,
description) values (7, 'a_vaa' ,
'A party. voice activated attendant. voice dialing' ); insert into MD_endpointType (endpointType, name,
description) values (8, 'a_hold' ,
'A party. music on hold or other hold service' ); insert into MD_endpointType (endpointType, name,
description) values (9, 'a_remote' ,
'A party. remote party' );

insert into MD_endpointType (endpointType, name,
description) values (10, 'a_other' ,
'A party. other' );

insert into MD_endpointType (endpointType, name,
description) values (11, 'a_spare_b' ,

```

```
'A party. spare' );  
  
insert into MD_endpointType (endpointType, name,  
description) values (12, 'a_spare_c' ,  
'A party. spare' );  
  
insert into MD_endpointType (endpointType, name,  
description) values (13, 'a_spare_d' ,  
'A party. spare' );  
  
insert into MD_endpointType (endpointType,  
name,  
description) values (14, 'a_spare_e' ,  
'A party. spare' );  
  
insert into MD_endpointType (endpointType, name,  
description) values (15, 'a_spare_f' ,  
'A party. spare' );  
  
insert into MD_endpointType (endpointType, name,  
description) values (16, 'b_ep' ,  
'B party terminal end point' );  
  
insert into MD_endpointType (endpointType, name,  
description) values (17, 'b_eemg' ,  
'B party. media gateway.' );  
  
insert into MD_endpointType (endpointType, name,  
description) values (18, 'b_gw' ,  
'B party. other gateway' );  
  
insert into MD_endpointType (endpointType, name,  
description) values (19, 'b_multiLim' ,
```

```

'B party. Multi LIM' ) ;

insert into MD_endpointType (endpointType, name,
description) values (20, 'b_operator' ,
'B party. operator' ) ;

insert into MD_endpointType (endpointType, name,
description) values (21, 'b_voiceMail' ,
'B party. voice mail' ) ;

insert into MD_endpointType (endpointType, name,
description) values (22, 'b_voiceMenu' ,
'B party. voice menu. ETSI*/ Stored Voice Services' ) ; insert into MD_endpointType (endpointType,
name,
description) values (23, 'b_vaa' ,
'B party. voice activated attendant. voice dialing' ) ; insert into MD_endpointType (endpointType, name,
description) values (24, 'b_hold' ,
'B party. music on hold or other hold service' ) ; insert into MD_endpointType (endpointType, name,
description) values (25,
'b_remote' ,
'B party. remote party' ) ;

insert into MD_endpointType (endpointType, name,
description) values (26, 'b_other' ,
'B party. other' ) ;

insert into MD_endpointType (endpointType, name,
description) values (27, 'b_spare_b' ,
'B party. spare' ) ;

insert into MD_endpointType (endpointType, name,

```

```

description) values (28, 'b_spare_c' ,
'B party. spare' ) ;

insert into MD_endpointType (endpointType, name,
description) values (29, 'b_spare_d' ,
'B party. spare' ) ;

insert into MD_endpointType (endpointType, name,
description) values (30, 'b_spare_e' ,
'B party. spare' ) ;

insert into MD_endpointType (endpointType, name,
description) values (31, 'b_spare_f' ,
'B party. spare' ) ;

insert into MD_endpointType (endpointType, name,
description) values (32, 'o_ep' ,
'Other party not A or B terminal end point' ) ; insert into MD_endpointType (endpointType, name,
description) values (33, 'o_eemg' ,
'Other party not A or B . media gateway.' ) ; insert into MD_endpointType (endpointType, name,
description) values (34, 'o_gw' ,
'Other party not A or B . other gateway' ) ; insert into MD_endpointType (endpointType, name,
description) values (35, 'o_multiLim' ,
'Other party not A or B . Multi LIM' ) ;

insert into MD_endpointType (endpointType, name,
description) values (36, 'o_operator' ,
'Other party not A or B . operator' ) ;

insert into MD_endpointType (endpointType, name,
description) values (37, 'o_voiceMail' ,

```

```

'Other party not A or B . voice mail' ) ;

insert into MD_endpointType (endpointType, name,
description) values (38, 'o_voiceMenu' ,
'Other party not A or B . voice menu' ) ; insert into MD_endpointType (endpointType, name,
description) values (39, 'o_vaa' ,
'Other party not A or B . voice activated attendant' ) ; insert into MD_endpointType (endpointType,
name,
description) values (40, 'o_hold' ,
'Other party not A or B . music on hold or other hold service' ) ; insert into MD_endpointType
(endpointType,
name,
description) values (41, 'o_remote' ,
'Other party not A or B . remote party' ) ; insert into MD_endpointType (endpointType, name,
description) values (42, 'o_other' ,
'Other party not A or B . other' ) ;

insert into MD_endpointType (endpointType, name,
description) values (43, 'o_spare_b' ,
'Other party. spare' ) ;

insert into MD_endpointType (endpointType, name,
description) values (44, 'o_spare_c' ,
'Other party. spare' ) ;

insert into MD_endpointType (endpointType, name,
description) values (45, 'o_spare_d' ,
'Other party. spare' ) ;

insert into MD_endpointType (endpointType, name,
description) values (46, 'o_spare_e' ,

```

```
'Other party. spare' );

insert into MD_endpointType (endpointType, name,
description) values (47, 'o_spare_f' ,
'Other party. spare' );

insert into MD_endpointType (endpointType, name,
description) values (48, 'otherKind' ,
'Other kind of endpoint' );

--

-- Everything created.

--
```

7.2 XML OUTPUT FORMAT

The XML printout contains one record per call. For the field names, [See "CIL Data Fields."](#)



Note:

Data fields not carrying valid data are excluded from the output in this format.

Comments according to XML format might be embedded in the file.

```
<?xml version="1.0" encoding="iso8859-1"?>

<DAREC_data>

<Record>

<startTime><year>2004</year>

<month>10</month>

<day>6</day>

<hour>15</hour>

<minute>7</minute>
```



```
<second>38</second>

<micro_second>0</micro_second>

<weekday>3</weekday>

<isdst>no</isdst>

</startTime>

<stopTime> <year>2004</year>

<month>10</month>

<day>6</day>

<hour>15</hour>

<minute>7</minute>

<second>39</second>

<micro_second>0</micro_second>

<weekday>3</weekday>

<isdst>no</isdst>

</stopTime>

<duration>1</duration>

<callingNumber>11</callingNumber>

<callingNumberType>1</callingNumberType>

<accessCode1>1</accessCode1>

<accessCode2>1</accessCode2>

<dialedNumber>101</dialedNumber>

<connectedNumber>101</connectedNumber>

<conditionCode>8</conditionCode>

<callCaseData>0</callCaseData>

<chargedNumber>101</chargedNumber>
```

```

<accountCode>11</accountCode> <taxPulses>1</taxPulses>

<cilCode>1</cilCode>

<tnsCode>1</tnsCode>

<osaCode>1</osaCode>

<ogtrnkid>1</ogtrnkid>

<trunkQueTime>1</trunkQueTime>

<opQueTime>1</opQueTime>

<cstCorrData>1</cstCorrData>

<ringTimeCounter>4</ringTimeCounter>

<queueTimeCounter>4</queueTimeCounter>

<equ>

<lim>2</lim>

<magazine>1</magazine>

<board>1</board>

<individual>1</individual>

</equ>

<eventLevel>1</eventLevel>

<isMobileLogging>0</isMobileLogging>

<inctrnkid>1</inctrnkid>

<seqNumber>1</seqNumber>

<seqLim>1</seqLim>

<callId> 1</callId>

<callIdAss1> 0</callIdAss1>

<callIdAss2> 0</callIdAss2>

<encryptedMedia> no </encryptedMedia>

```

```

<qualityData>

<endpoint>

<endpointType>0</endpointType>

<endpointTypeName>a_ep</endpointTypeName>

<rtpAddress>121.131.141.151:551</rtpAddress>

<extension>foo@1</extension>

<qos>

<worstEstimatedEnd2EndDelay>11

</worstEstimatedEnd2EndDelay>

<meanEstimatedEnd2EndDelay>11

</meanEstimatedEnd2EndDelay>

<cumulativeNumberOfPacketsLost>1

</cumulativeNumberOfPacketsLost>

<packetLostRate>1</packetLostRate>

<worstJitter>1</worstJitter>

<estimatedThroughput>1001

</estimatedThroughput>

<fractionLostRate>1</fractionLostRate>

<meanJitter>1</meanJitter>

<codec>

<codec_ind>

<type>S</type>

<codecName>3</codecName>

</codec_ind>

</codec>

```

```

<simpleR_value>500</simpleR_value>

</qos>

</endpoint>

<endpoint>

<endpointType>16</endpointType>

<endpointTypeName>b_ep</endpointTypeName>

<rtpAddress>122.132.142.152:552</rtpAddress>

<extension>foo@2</extension>

<qos>

<worstEstimatedEnd2EndDelay>102

</worstEstimatedEnd2EndDelay>

<meanEstimatedEnd2EndDelay>12

</meanEstimatedEnd2EndDelay>

<cumulativeNumberOfPacketsLost>52

</cumulativeNumberOfPacketsLost>

<packetLostRate>6</packetLostRate>

<worstJitter>312</worstJitter>

<estimatedThroughput>1002

</estimatedThroughput>

<fractionLostRate>13</fractionLostRate>

<meanJitter>32</meanJitter>

<codec>

<codec_ind>

<type>V</type>

<codecName>2</codecName>

```

```

</codec_ind>

<codec_ind>

<type>V</type>

<codecName>2</codecName>

</codec_ind> </codec> <simpleR_value>5002</simpleR_value>

</qos>

</endpoint>

</qualityData>

</Record>

</DAREC_data>

```

7.3 COMMA-SEPARATED OUTPUT FORMAT

The output always starts with a header that explains the format. In this example the backslash (\) is used to indicate that no new line character is used in the real data stream.

Call and quality data stored by DAREQ in comma separated format

First 8 lines are comments, after that every line is comma separated data # "&" in data is stored as "&";" and "," as ","

The number of endpoints are open. If "QoS endpoint block x valid"

is false (=0) no (more) endpoint(s) will follow.

The following line is comma separated column heading

start time UTC, stop time UTC, start time local, stop time local, duration, calling number, calling number type, calling number valid, access code 1, access code 2, dialed number, connected number, condition code, call case data, charged number, account code, account code valid, tax pulses, tax pulses valid, cil code, cil code valid, tns code, tns code valid, osa code, osa code valid, og trnk id, trunk queue time, og trnk id and trunk queue time valid, op queue time, op queue time valid, cstCorrData, ring time counter, queue time counter, time counters valid, equ, equ valid, event level, is mobile logging, inc trnk id, inc trnk id valid, seq num, seq lim, callId, CallIdAss1, CallIdAss2, QoS endpoint block 1 valid, endpoint type 1, rtp addr 1, extension 1, QoS data block 1 valid, worst estimated end to end delay 1, mean estimated end to end delay 1, cum num of packets lost 1, packet lost rate 1, worst jitter 1, estimated throughput 1, fraction lost rate 1, mean jitter 1, codec 1-1 type, codec 1-1, codec 1-2 type, codec 1-2, codec 1-3 type, codec 1-3, simple R value 1, QoS endpoint block 2 valid, endpoint type 2, rtp addr 2, extension 2, QoS data block 2 valid, worst estimated end to end delay 2, mean estimated end to end delay 2, cum num of packets lost 2, packet lost rate 2, worst jitter 2, estimated throughput 2, fraction lost rate 2, mean jitter 2, codec 2-1 type, codec 2-1, codec 2-2 type, codec 2-2, codec 2-3 type, codec 2-3, simple R value 2, QoS endpoint

block 3 valid, endpoint type 3, rtp addr 3, extension 3, QoS data block 3 valid, worst estimated end to end delay 3, mean estimated end to end dealy 3, cum num of packets lost 3, packet lost rate 3, worst jitter 3, estimated throughput 3, fraction lost rate 3, mean jitter 3 ,codec 3-1 type, codec 3-1, codec 3-2 type, codec 3-2, codec 3-3 type, codec 3-3, simple R value 3, QoS endpoint block 4 valid, endpoint type 4, rtp addr 4, extension 4, QoS data block 4 valid, worst estimated end to end delay 4, mean esti- mated end to end dealy 4, cum num of packets lost 4, packet lost rate 4, worst jitter 4, estimated throughput 4, fraction lost rate 4, mean jitter 4 ,codec 4-1 type, codec 4-1, codec 4-2 type, codec 4-2, codec 4-3 type, codec 4-3, simple R value 4, QoS endpoint block 5 valid, endpoint type 5, rtp addr 5, extension 5, QoS data block 5 valid, worst estimated end to end delay 5, mean estimated end to end dealy 5, cum num of packets lost 5, packet lost rate 5, worst jitter 5, estimated throughput 5, fraction lost rate 5, mean jitter 5 ,codec 5-1 type, codec 5-1, codec 5-2 type, codec 5-2, codec 5-3 type, codec 5-3, simple R value 5 QoS endpoint block 6 valid, endpoint type 6, rtp addr 6, extension 6, QoS data block 6 valid, worst estimated end to end delay 6, mean estimated end to end dealy 6, cum num of packets lost 6, packet lost rate 6, worst jitter 6, estimated throughput 6, fraction lost rate 6, mean jitter 6 ,codec 6-1 type, codec 6-1, codec 6-2 type, codec 6-2, codec 6-3 type, codec 6-3, simple R value 6,

#

2004-12-16 12:32:12, 2004-12-16 12:32:13, 2004-12-16 13:32:12 (CET), 2004-12-16 13:32:13 (CET), 0d00:00:01, 11, 1, 1, 21, 1, 101, 1001, 131, 131, 1236, 21, 1, 2, 1, 31, 1, 41, 1, 1, 1, 61, 3, 1, 21, 1, 81, 4, 5, 1, 2-1-1-1, 1, 111, 1, 121, 1, 1, 1, 0, 1, 2, 1, 0, 121.131.141.151:551, foo@1, 1, 101, 11, 51, 5, 311, 1001, 12, 31, S, 2, l, , l, , 5001, 1, 16, 122.132.42.152:552, foo@2, 1, 102, 12, 52, 6, 312, 1002, 13, 32, S, 2, l, , l, , 5002, \ 0

This chapter contains the following sections:

- [USING A SCRIPT AS BOTH FORMATTER AND FILTER](#)

The subtype "general" uses the -format parameter to filter and format the data.

8.1 USING A SCRIPT AS BOTH FORMATTER AND FILTER

The format string for the general formatter consists of one or several format specifications separated by semicolons (;). The semicolon may only be used for this kind of separation in the format string.

Each format specification consists of an optional condition followed by a colon ":" and then followed by the format information. The colon needs to be present only if the condition is present. The colon may only be used in the format string to separate the condition from the format information.

The record is output using the format information only if the condition is true. A format specification without a condition is the same as a condition that is always true. The condition is built up from one or several subconditions, each enclosed in brackets. A subcondition has the form:

[fieldName operator value] or [fieldName operator fieldName] where the operator is one of

==

!=

<=

>=

<

>

begins

ends

contains

like

For <=, >=, < and > a plain ASCII comparison up to the length of the shortest string is done. 'begins', 'ends', 'contains', and 'like' are well defined for fields with string values. 'begins' checks if the beginning

matches. 'ends' checks if the end matches. 'contains' checks if the right hand side is anywhere in the left hand side value.

'like' matches the complete value against a string that can include multiple wild characters: '?' for any single character, or '%' for zero or more characters. ('like' uses much more CPU power than the other operators.)

It is possible to add '!' in front of any operator.

A condition built up of subconditions is true only if all the subconditions are true. The following special conditions are also accepted:

[head] True only the first time. Used to write a heading

[foot] True only after all data is written. Used to write a footer.

The format information is written as it is, with the exception that any text enclosed in curly braces "{}" is used to fetch data from the variable or function specified.

The curly "{" entrance and "}" exit braces may occur in the format string only for this purpose.

The following functions exist:

{exit} stop execution of this call record parsing.

{semicolon} output a semicolon ";"

{colon} output a colon ":"

{leftBrace} output a left brace "{"

{rightBrace} output a right brace "}"

{newline} output a newline (linefeed)

{return} output a carriage return

{ascii code} output the ASCII character with the decimal value code

{fieldName justification minWidth maxWidth} where:

justification is "L" for left or "R" for right or

"OR" for right with zero-fill to the left

minWidth is the minimum width as decimal number maxWidth is the maximum width as decimal number

{fieldName subFieldName justification minWidth maxWidth} where fieldName must refer to a time value:

subFieldName is "day" for day of month (1-31) is "month" for month of year (1-12)

is "year2" for year (00-99)

is "year" for year (1970-2038) is "hour" for hour (0-24)

is "minute" for minute (0-59) is "second" for second (0-59)

is "duration" for duration formatted is "raw" for raw formatted

is "iso" for ISO formatted date and time is "isoDate" for ISO formatted date

is "isoTime" for ISO formatted time

is "md110date" for MD110 compatible date is "md110time" for MD110 compatible time

is "md110duration" for MD110 compatible duration

is "md110durationDecMinute" for MD110 compatible duration

subfieldName can also be a string including %, and is then interpreted according to the UNIX strftime(),

see "man strftime" see above for an explanation of the rest of the format.

{fieldName localUtc subFieldName justification minWidth maxWidth} where fieldName must refer to a time value:

localUtc is "utc" for output as UTC (default), is "local" for output as local time,

is "localtime" as synonym for "local".

see above for an explanation of the rest of the format. The field names are:

accessCode1

accessCode2

accountCode

accountCodeValid

callCaseData

callingNumber

callingNumberType

callingNumberValid

callId

callIdAss1

GENERAL FORMAT

callIdAss2

chargedNumber

cilCode

cilCodeValid

codecName1_1

codecName1_2

codecName1_3

codecName1_4

codecName1_5

codecName1_6

codecName2_1

codecName2_2

codecName2_3

codecName2_4

codecName2_5

codecName2_6

codecName3_1

codecName3_2

codecName3_3

codecName3_4

codecName3_5

codecName3_6

codecType1_1

codecType1_2

codecType1_3

codecType1_4
codecType1_5
codecType1_6
codecType2_1
codecType2_2
codecType2_3
codecType2_4
codecType2_5
codecType2_6
codecType3_1
codecType3_2
codecType3_3
codecType3_4
codecType3_5
codecType3_6
conditionCode1Character
conditionCode2Character
conditionCode3Character
conditionCodeUserDefined
conditionCode
conditionCodeSeg
conditionCodePos
connectedNumber
cstCorrData
cumulativeNumberOfPacketsLost_1

cumulativeNumberOfPacketsLost_2
cumulativeNumberOfPacketsLost_3
cumulativeNumberOfPacketsLost_4
cumulativeNumberOfPacketsLost_5
cumulativeNumberOfPacketsLost_6 currentTime
dialedNumber
duration
endpointType_1
endpointType_2
endpointType_3
endpointType_4
endpointType_5
endpointType_6
endpointTypeName_1
endpointTypeName_2
endpointTypeName_3
endpointTypeName_4
endpointTypeName_5
endpointTypeName_6
endpointTypeDesc_1
endpointTypeDesc_2
endpointTypeDesc_3
endpointTypeDesc_4
endpointTypeDesc_5
endpointTypeDesc_6

equValid
equ
equNoMgw
equLim
equMgw
equMgwNoLim
equMagazine
equBoardPosition
equBoardIndividual
eventLevel
estimatedThroughput_1
estimatedThroughput_2
estimatedThroughput_3
estimatedThroughput_4
estimatedThroughput_5
estimatedThroughput_6
extension_1
extension_2
extension_3
extension_4
extension_5
extension_6
fractionLostRate_1
fractionLostRate_2
fractionLostRate_3

GENERAL FORMAT

fractionLostRate_4

fractionLostRate_5

fractionLostRate_6

freeOfChargeCallinformation

incTrnkId

incTrnkId9

incTrnkIdValid

isMobileLogging

meanEstimatedEnd2EndDelay_1

meanEstimatedEnd2EndDelay_2

meanEstimatedEnd2EndDelay_3

meanEstimatedEnd2EndDelay_4

meanEstimatedEnd2EndDelay_5

meanEstimatedEnd2EndDelay_6

meanJitter_1

meanJitter_2

meanJitter_3

meanJitter_4

meanJitter_5

meanJitter_6

mobilityEvent

ogTrnkId

ogTrnkId9

ogTrnkValid

opQueTime

opQueTimeValid

osaCode

osaCodeValid

packetLostRate_1

packetLostRate_2

packetLostRate_3

packetLostRate_4

packetLostRate_5

packetLostRate_6

qosDataValid_1

qosDataValid_2

qosDataValid_3

qosDataValid_4

qosDataValid_5

qosDataValid_6

qosEndPointValid_1

qosEndPointValid_2

qosEndPointValid_3

qosEndPointValid_4

qosEndPointValid_5

qosEndPointValid_6

queueTimeCounter

ringTimeCounter

rtpAdress_1

rtpAdress_2

GENERAL FORMAT

rtpAdress_3

rtpAdress_4

rtpAdress_5

rtpAdress_6

seqNumber

seqLim

simpleR_value_1

simpleR_value_2

simpleR_value_3

simpleR_value_4

simpleR_value_5

simpleR_value_6

startTime

stopTime

taxPulses

taxPulsesValid

timeCountersValid

tnsCode

tnsCodeValid

trunkQueTime

worstEstimatedEnd2EndDelay_1

worstEstimatedEnd2EndDelay_2

worstEstimatedEnd2EndDelay_3

worstEstimatedEnd2EndDelay_4

worstEstimatedEnd2EndDelay_5

worstEstimatedEnd2EndDelay_6

worstJitter_1

worstJitter_2

worstJitter_3

worstJitter_4

worstJitter_5

worstJitter_6

The field names are not case sensitive, [See “CIL Data Fields”](#). .

8.1.1 EXAMPLES

Format **demo1** can be written as:

```
[head]:Call list excluding emergency calls{newline};\
[isMobileLogging != 1][dialedNumber != 112]:\
{startTime L 19 19} {duration R 15 15} from \
{callingNumber L 20 20} to {connectedNumber L 0 20}{newline};\ [foot]:End of list.{newline}
```

Format **demo2** can be written as:

```
[head]:Call list demo 2{newline};\
[isMobileLogging != 1][callingNumberValid == 1]:From \
{callingNumber L 0 20};\
[isMobileLogging != 1]: dialed {dialedNumber L 0 20 }; \
[isMobileLogging != 1][connectedNumber != ""]: \
to {connectedNumber L 0 20}; \
[isMobileLogging != 1][connectedNumber == ""]: not connected; \
[isMobileLogging != 1]: duration {duration R 0 15}; \
[isMobileLogging != 1][chargedNumber != ""]: \
charged {chargedNumber L 0 20}; \
```

```
[isMobileLogging != 1]:{newline}From {startTime L 0 19} to \
{stopTime L 0 19}{newline}{newline};\ [foot]:End of list.{newline}
```

Format **mobility** can be written as:

```
[head]:Mobility logging{newline}; \
[isMobileLogging==0] : {startTime L 19 19} {duration R 15 15} \
from {callingNumber L 20 20} to {connectedNumber L 0 20} \
{newline}; \
[isMobileLogging==1] : {startTime L 19 19} \
{callingNumber L 5 5} {mobilityEvent L 3 3} {equLim R 3 3} \
{equMgwNoLim L 1 1} - {equMagazine L 1 1} \
- {equBoardPosition L 2 2} {equBoardIndividual R 3 3} \
{mobilityEvent L 4 4} {mobilityText L 20 60} {newline};
```

Format **FP15** can be written as:

```
[head]: {currentTime month 0R 2 2} {currentTime day 0R 2 2}\
{newline};\
[isMobileLogging != 1]: {stoptime md110time L 4 4} \
{duration md110durationDecMinute L 5 5}\
{conditioncode1character L 2 2}{accesscode1 R 3 3}\
{accesscode2 R 3 3} {dialednumber R 15 15} \
{callingnumber L 4 4} ;\
[isMobileLogging != 1][accountcodevalid != 1]:{cilCode R 15 15}\
{newline};\
[isMobileLogging != 1][accountcodevalid == 1]:{accountcode R 15 15}\
{newline};
```

Format **MDFP15** can be written as:

```
[head]: {currentTime month 0R 2 2} {currentTime day 0R 2 2} {newline};\

[isMobileLogging != 1]: {stoptime md110time L 4 4} {duration md110duration L 5 5} \

{conditioncode2character L 2 2} \

{accesscode1 R 3 3}{accesscode2 R 3 3} ;\

[isMobileLogging != 1] [opquetimevalid != 1]:{dialednumber R 20 20} ;\

[isMobileLogging != 1] [opquetimevalid == 1]:{opQueTime R 2 2} ; \

[isMobileLogging != 1]:{callingnumber L 10 10} {accountcode R 15 15} \

{cilcode R 6 6} {tnscore R 3 3} {osacode R 2 2} ;\

[isMobileLogging != 1] [ogTrnkValid == 1]:{ogtrnkid9 R 9 9}{newline};\

[isMobileLogging != 1] [ogTrnkValid != 1]:{connectedNumber R 9 9}\

{newline};
```

Format **ASB501** can be written as:

```
[head]: {currentTime month 0R 2 2} {currentTime day 0R 2 2}"

{newline};" [isMobileLogging != 1]\

: {stoptime md110date L 4 4}{stoptime md110time L 4 4} \

{duration md110duration L 5 5} {taxpulses 0R 4 4} \

{conditioncode2character L 2 2} \

{accesscode1 R 5 5}{accesscode2 R 5 5} ;\

[isMobileLogging != 1][opquetimevalid != 1]:{dialednumber R 20 20} ;\

[isMobileLogging != 1][opquetimevalid == 1]:{opQueTime R 2 2} ; \

[isMobileLogging != 1][chargedNumber != "]:{chargednumber L 10 10} ;\

[isMobileLogging != 1][chargedNumber == "]:{callingnumber L 10 10} ;\

[isMobileLogging != 1]:{accountcode R 15 15} \

{cilcode R 6 6} {tnscore R 3 3} ;\

[isMobileLogging != 1][osaCodeValid == 1]:{osacode R 2 2} ;\
```

[isMobileLogging != 1][osaCodeValid != 1]:;\

[isMobileLogging != 1]:{trunkquetime 0R 2 2};\

[[isMobileLogging != 1][ogTrnkValid == 1]:{ogTrnkId9 R 9 9} \

{newline}{exit};\

[isMobileLogging != 1][ogTrnkValid != 1]:{connectedNumber R 9 9} \

{newline};

Format **ASBUMDFP15** can be written as:

[head]: {currentTime month 0R 2 2} {currentTime day 0R 2 2} \

{newline};\ [isMobileLogging != 1]\

: {stoptime md110time L 4 4} {duration md110durationDecMinute L 4 4} \

{conditioncode1character L 1 1} \

{accesscode1 R 3 3}{accesscode2 R 3 3} {dialednumber R 20 20} \

{callingnumber L 5 5} {accountcode R 15 15} \

{cilcode R 6 6} ;\

[isMobileLogging != 1][ogTrnkValid == 1]:{ogTrnkId9 R 9 9} \

{newline}{exit};\

[isMobileLogging != 1][ogTrnkValid != 1]:{connectedNumber R 9 9} \

{newline};

This chapter contains the following sections:

- [USING A SCRIPT AS FILTER](#)
- [HEADER INFORMATION WITH PREDEFINED FORMATS \(FP15, MDFP15, ASB501, ASBU MDFP15\)](#)
- [FP15 FORMAT](#)
- [MDFP15 FORMAT](#)
- [ASB 501 \(MD110 STANDARD\) FORMAT](#)
- [ASBU MDFP15 FORMAT](#)
- [HEARTBEAT FORMAT](#)
- [DEMO1 FORMAT](#)
- [DEMO2 FORMAT](#)
- [MOBILITY FORMAT](#)

The CIL and QsS call logging function provides the ability to screen which call records are to be output. The output criteria is defined by I/O commands.

The call logging will output only those call records that meet the output criteria.

Subtypes fp15, mdfp15, asb501, and asbumdfp15 uses the -format parameter as input to set the time output format and to filter the data prior to recording.

The format parameter string must start with one of the words "local", "localtime", or "utc", to specify whether times and dates are printed as localtime or UTC.

The callinfo_output_set commands -format parameter defines the rules of what to collect and discard. A script language can test any field in the data. One script may be defined per output.

9.1 USING A SCRIPT AS FILTER

The format string must start with one of the words "local", "localtime", or "utc", to specify whether times and dates are printed as localtime or UTC. The default action is to output all records except for mobile logging.

By specifying a format string condition the recorded data can be filtered before it is written to the output unit.

The condition is built up from one or several subconditions, each enclosed in brackets. A subcondition has the form: [fieldName operator value] or [fieldName operator field- Name] where the operator is one of

==

!=

<=

>=

<

>

begins

ends

contains

like

For <=, >=, < and > a plain ASCII comparison up to the length of the shortest string is done. 'begins', 'ends', 'contains', and 'like' are well defined for fields with string values. 'begins' checks if the beginning matches. 'ends' checks if the end matches. 'contains' checks if the right hand side is anywhere in the left hand side value.

'like' matches the complete value against a string that can include multiple wild characters: '?' for any single character, or '%' for zero or more characters. ('like' uses much more CPU power than the other operators.)

It is possible to add '!' in front of any operator.

A condition built up of subconditions is true only if all the subconditions are true. The field names are:

accessCode1

accessCode2

accountCode

accountCodeValid

callCaseData

callId

callIdAss1

callIdAss2

chargedNumber

callingNumber

callingNumberType

callingNumberValid

cilCode

cilCodeValid

codecName1_1

codecName1_2

codecName1_3

codecName1_4

codecName1_5

codecName1_6

codecName2_1

codecName2_2

codecName2_3

codecName2_4

codecName2_5

codecName2_6

codecName3_1

codecName3_2

codecName3_3

codecName3_4

codecName3_5

codecName3_6

codecType1_1

codecType1_2

codecType1_3

codecType1_4

codecType1_5

PREDEFINED FORMATS

codecType1_6

codecType2_1

codecType2_2

codecType2_3

codecType2_4

codecType2_5

codecType2_6

codecType3_1

codecType3_2

codecType3_3

codecType3_4

codecType3_5

codecType3_6

conditionCode1Character

conditionCode2Character

conditionCode3Character

conditionCodeUserDefined

conditionCode

conditionCodeSeg

conditionCodePos

connectedNumber

cstCorrData

cumulativeNumberOfPacketsLost_1

cumulativeNumberOfPacketsLost_2

cumulativeNumberOfPacketsLost_3

cumulativeNumberOfPacketsLost_4

cumulativeNumberOfPacketsLost_5

cumulativeNumberOfPacketsLost_6

currentTime

dialedNumber

duration

encryptedMedia

endpointType_1

endpointType_2

endpointType_3

endpointType_4

endpointType_5

endpointType_6

endpointTypeName_1

endpointTypeName_2

endpointTypeName_3

endpointTypeName_4

endpointTypeName_5

endpointTypeName_6

endpointTypeDesc_1

endpointTypeDesc_2

endpointTypeDesc_3

endpointTypeDesc_4

endpointTypeDesc_5

endpointTypeDesc_6

PREDEFINED FORMATS

equValid
equ
equNoMgw
equLim
equMgw
equMgwNoLim
equMagazine
equBoardPosition
equBoardIndividual
eventLevel
estimatedThroughput_1
estimatedThroughput_2
estimatedThroughput_3
estimatedThroughput_4
estimatedThroughput_5
estimatedThroughput_6
extension_1
extension_2
extension_3
extension_4
extension_5
extension_6
fractionLostRate_1
fractionLostRate_2
fractionLostRate_3

fractionLostRate_4

fractionLostRate_5

fractionLostRate_6

incTrnkId

incTrnkId9

incTrnkIdValid

isMobileLogging

meanEstimatedEnd2EndDelay_1

meanEstimatedEnd2EndDelay_2

meanEstimatedEnd2EndDelay_3

meanEstimatedEnd2EndDelay_4

meanEstimatedEnd2EndDelay_5

meanEstimatedEnd2EndDelay_6

meanJitter_1

meanJitter_2

meanJitter_3

meanJitter_4

meanJitter_5

meanJitter_6

mobilityEvent ogTrnkId

ogTrnkId9

ogTrnkValid

opQueTime

opQueTimeValid

osaCode

osaCodeValid

packetLostRate_1

packetLostRate_2

packetLostRate_3

packetLostRate_4

packetLostRate_5

packetLostRate_6

qosDataValid_1

qosDataValid_2

qosDataValid_3

qosDataValid_4

qosDataValid_5

qosDataValid_6

qosEndPointValid_1

qosEndPointValid_2

qosEndPointValid_3

qosEndPointValid_4

qosEndPointValid_5

qosEndPointValid_6

queueTimeCounter ringTimeCounter

rtpAdress_1

rtpAdress_2

rtpAdress_3

rtpAdress_4

rtpAdress_5

rtpAdress_6

seqNumber seqLim

simpleR_value_1

simpleR_value_2

simpleR_value_3

simpleR_value_4

simpleR_value_5

simpleR_value_6

startTime

stopTime

taxPulses

taxPulsesValid

timeCountersValid

tnsCode tnsCodeValid

trunkQueTime

worstEstimatedEnd2EndDelay_1

worstEstimatedEnd2EndDelay_2

worstEstimatedEnd2EndDelay_3

worstEstimatedEnd2EndDelay_4

worstEstimatedEnd2EndDelay_5

worstEstimatedEnd2EndDelay_6

worstJitter_1

worstJitter_2

worstJitter_3

worstJitter_4

worstJitter_5

worstJitter_6

The field names are not case sensitive, [See “CIL Data Fields”](#).

9.1.1 EXAMPLE

Example script to output all calls except calls to emergency numbers and callback from repair personnel:

```
local [dialedNumber != 00112][dialedNumber != callingNumber]
```

9.2 HEADER INFORMATION WITH PREDEFINED FORMATS (FP15, MDFP15, ASB501, ASBU MDFP15)

00	SPACE
01	MONTH TENS
02	MONTH UNITS
03	SPACE
04	DAY TENS
05	DAY UNITS
06	CARRIAGE RETURN*
07	LINE FEED*NULL*
08	NULL*
09	NULL*
10	NULL*

**Note:**

The end of line sequence is controlled by the *callinfo_output_set* command.

9.3 FP15 FORMAT

00	SPACE
01	SPACE
02	SPACE
03	TIME-HR.TENS
04	TIME-HR.UNITS
05	TIME.MINUTE.TENS
06	TIME.MINUTE.UNITS
07	SPACE
08	DURATION-HOUR
09	DURATION-MINUTE.TENS
10	DURATION-MINUTE.UNITS
11	DURATION-MINUTE TENTH
12	SPACE
13	CONDITION CODE

PREDEFINED FORMATS

14	SPACE
15	ACCESS CODE 1 -1
16	ACCESS CODE 1 -2
17	ACCESS CODE 1 -3
18	ACCESS CODE 2/ISDN CBC
19	ACCESS CODE 2/ISDN CBC
20	ACCESS CODE 2/ISDN CBC
21	SPACE
22	DIALLED NUMBER -1
23	DIALLED NUMBER -2
24	DIALLED NUMBER -3
25	DIALLED NUMBER -4
26	DIALLED NUMBER -5
27	DIALLED NUMBER -6
28	DIALLED NUMBER -7
29	DIALLED NUMBER -8
30	DIALLED NUMBER -9

31	DIALLED NUMBER -10
32	DIALLED NUMBER -11
33	DIALLED NUMBER -12
34	DIALLED NUMBER -13
35	DIALLED NUMBER -14
36	DIALLED NUMBER -15
37	SPACE
38	CHARGED NUMBER -1
39	CHARGED NUMBER -2
40	CHARGED NUMBER -3
41	CHARGED NUMBER -4
42	SPACE
43	ACCOUNT CODE/CILCODE (Authorization code) -1
44	ACCOUNT CODE/CILCODE (Authorization code) -2
45	ACCOUNT CODE/CILCODE (Authorization code) -3
46	ACCOUNT CODE/CILCODE (Authorization code) -4
47	ACCOUNT CODE/CILCODE (Authorization code) -5

48	ACCOUNT CODE/CILCODE (Authorization code) -6
49	ACCOUNT CODE/CILCODE (Authorization code) -7
50	ACCOUNT CODE/CILCODE (Authorization code) -8
51	ACCOUNT CODE/CILCODE (Authorization code) -9
52	ACCOUNT CODE/CILCODE (Authorization code) -10
53	ACCOUNT CODE/CILCODE (Authorization code) -11
54	ACCOUNT CODE/CILCODE (Authorization code) -12
55	ACCOUNT CODE/CILCODE (Authorization code) -13
56	ACCOUNT CODE/CILCODE (Authorization code) -14
57	ACCOUNT CODE/CILCODE (Authorization code) -15
58	CARRIAGE RETURN*
59	LINE FEED*
60	NULL*
61	NULL*
62	NULL*

**Note:**

The end of line sequence is controlled by the *callinfo_output_set* command.

The ACCOUNT CODE/AUTHORIZATION CODE is a shared field, where either the cilCode (authorization code) or the accountCode is stored. If cilCode and accountCode are both entered, the accountCode will be recorded and cilCode will be discarded.

For calls answered by direct call pickup, the answering party will not be recorded, since FP15 does not contain an outgoing external line ID field.

For all the diversion calls, the answering party will not be recorded, since FP15 does not contain an outgoing external line ID field.

9.4 MDFP15 FORMAT

00	SPACE
01	SPACE
02	SPACE
03	TIME-HR.TENS
04	TIME-HR.UNITS
05	TIME.MINUTE.TENS
06	TIME.MINUTE.UNITS
07	SPACE
08	DURATION-HOUR
09	DURATION-MINUTE.TENS
10	DURATION-MINUTE.UNITS
11	DURATION-MINUTE TENTH / SECOND.TENS
12	SECOND.UNITS

13	SPACE
14	CONDITION CODE -1
15	CONDITION CODE -2
16	SPACE
17	ACCESS CODE 1 -1
18	ACCESS CODE 1 -2
19	ACCESS CODE 1 -3
20	ACCESS CODE 2/ISDN CBC -1
21	ACCESS CODE 2/ISDN CBC -2
22	ACCESS CODE 2/ISDN CBC -3
23	SPACE
24	DIALLED NUMBER / OPERATOR QUEUE TIME -1
25	DIALLED NUMBER / OPERATOR QUEUE TIME -2
26	DIALLED NUMBER -3
27	DIALLED NUMBER -4
28	DIALLED NUMBER -5
29	DIALLED NUMBER -6

30	DIALLED NUMBER -7
31	DIALLED NUMBER -8
32	DIALLED NUMBER -9
33	DIALLED NUMBER -10
34	DIALLED NUMBER -11
35	DIALLED NUMBER -12
36	DIALLED NUMBER -13
37	DIALLED NUMBER -14
38	DIALLED NUMBER -15
39	DIALLED NUMBER -16
40	DIALLED NUMBER -17
41	DIALLED NUMBER -18
42	DIALLED NUMBER -19
43	DIALLED NUMBER -20
44	SPACE
45	CHARGED NUMBER -1
46	CHARGED NUMBER -2

47	CHARGED NUMBER -3
48	CHARGED NUMBER -4
49	CHARGED NUMBER -5
50	CHARGED NUMBER -6
51	CHARGED NUMBER -7
52	CHARGED NUMBER -8
53	CHARGED NUMBER -9
54	CHARGED NUMBER -10
55	SPACE
56	ACCOUNT CODE -1
57	ACCOUNT CODE -2
58	ACCOUNT CODE -3
59	ACCOUNT CODE -4
60	ACCOUNT CODE -5
61	ACCOUNT CODE -6
62	ACCOUNT CODE -7
63	ACCOUNT CODE -8

64	ACCOUNT CODE -9
65	ACCOUNT CODE -10
66	ACCOUNT CODE -11
67	ACCOUNT CODE -12
68	ACCOUNT CODE -13
69	ACCOUNT CODE -14
70	ACCOUNT CODE -15
71	SPACE
72	CILCODE (Authorization code) -1
73	CILCODE (Authorization code) -2
74	CILCODE (Authorization code) -3
75	CILCODE (Authorization code) -4
76	CILCODE (Authorization code) -5
77	CILCODE (Authorization code) -6
78	SPACE
79	TRANSIT NETWORK SELECTION CODE -1
80	TRANSIT NETWORK SELECTION CODE -2

81	TRANSIT NETWORK SELECTION CODE -3
82	SPACE
83	OPERATOR SYSTEM ACCESS CODE -1
84	OPERATOR SYSTEM ACCESS CODE -2
85	SPACE
86	EXTERNAL LINE ID -1 / CONNECTED NUMBER -1
87	EXTERNAL LINE ID -2 / CONNECTED NUMBER -2
88	EXTERNAL LINE ID -3 / CONNECTED NUMBER -3
89	EXTERNAL LINE ID -4 / CONNECTED NUMBER -4
90	EXTERNAL LINE ID -5 / CONNECTED NUMBER -5
91	EXTERNAL LINE ID -6 / CONNECTED NUMBER -6
92	EXTERNAL LINE ID -7 / CONNECTED NUMBER -7
93	EXTERNAL LINE ID -8 / CONNECTED NUMBER -8
94	EXTERNAL LINE ID -9 / CONNECTED NUMBER -9
95	CARRIAGE RETURN*

96	LINE FEED*
97	NULL*
98	NULL*
99	NULL*

**Note:**

The end of line sequence is controlled by the *callinfo_output_set* command.

The dialed number field is a shared field that presents either the dialedNumber or the opQueTime information.

For calls answered by direct call pickup, the answering party will be recorded in the outgoing external line ID field.

For all the diversion calls, the dialed number will be recorded in the dialed number field.

9.5 ASB 501 (MD110 STANDARD) FORMAT

00	SPACE
01	SPACE
02	SPACE
03	DATE-MONTH.TENS
04	DATE-MONTH.UNITS
05	DATE-DAY.TENS
06	DATE-DAY.UNITS

PREDEFINED FORMATS

07	TIME-HR.TENS
08	TIME-HR.UNITS
09	TIME-MINUTE.TENS
10	TIME-MINUTE.UNITS
11	SPACE
12	DURATION-HOUR
13	DURATION-MINUTE.TENS
14	DURATION-MINUTE.UNITS
15	DURATION-MINUTE TENTH / SECOND.TENS
16	SECOND.UNITS
17	SPACE
18	TAX PULSE THOUSANDS
19	TAX PULSE HUNDREDS
20	TAX PULSE TENS
21	TAX PULSE UNITS
22	SPACE
23	CONDITION CODE -1

24	CONDITION CODE -2
25	SPACE
26	ACCESS CODE 1 -1
27	ACCESS CODE 1 -2
28	ACCESS CODE 1 -3
29	ACCESS CODE 1 -4
30	ACCESS CODE 1 -5
31	ACCESS CODE 2 -1
32	ACCESS CODE 2 -2
33	ACCESS CODE 2/ISDN CBC -3
34	ACCESS CODE 2/ISDN CBC -4
35	ACCESS CODE 2/ISDN CBC -5
36	SPACE
37	DIALLED NUMBER / OPERATOR QUEUE TIME -1
38	DIALLED NUMBER / OPERATOR QUEUE TIME -2
39	DIALLED NUMBER -3
40	DIALLED NUMBER -4

41	DIALLED NUMBER -5
42	DIALLED NUMBER -6
43	DIALLED NUMBER -7
44	DIALLED NUMBER -8
45	DIALLED NUMBER -9
46	DIALLED NUMBER -10
47	DIALLED NUMBER -11
48	DIALLED NUMBER -12
49	DIALLED NUMBER -13
50	DIALLED NUMBER -14
51	DIALLED NUMBER -15
52	DIALLED NUMBER -16
53	DIALLED NUMBER -17
54	DIALLED NUMBER -18
55	DIALLED NUMBER -19
56	DIALLED NUMBER -20
57	SPACE

58	CHARGED NUMBER -1
59	CHARGED NUMBER -2
60	CHARGED NUMBER -3
61	CHARGED NUMBER -4
62	CHARGED NUMBER -5
63	CHARGED NUMBER -6
64	CHARGED NUMBER -7
65	CHARGED NUMBER -8
66	CHARGED NUMBER -9
67	CHARGED NUMBER -10
68	SPACE
69	ACCOUNT CODE -1
70	ACCOUNT CODE -2
71	ACCOUNT CODE -3
72	ACCOUNT CODE -4
73	ACCOUNT CODE -5
74	ACCOUNT CODE -6

75	ACCOUNT CODE -7
76	ACCOUNT CODE -8
77	ACCOUNT CODE -9
78	ACCOUNT CODE -10
79	ACCOUNT CODE -11
80	ACCOUNT CODE -12
81	ACCOUNT CODE -13
82	ACCOUNT CODE -14
83	ACCOUNT CODE -15
84	SPACE
85	CILCODE (Authorization code) -1
86	CILCODE (Authorization code) -2
87	CILCODE (Authorization code) -3
88	CILCODE (Authorization code) -4
89	CILCODE (Authorization code) -5
90	CILCODE (Authorization code) -6
91	SPACE

92	TRANSIT NETWORK SELECTION CODE -1
93	TRANSIT NETWORK SELECTION CODE -2
94	TRANSIT NETWORK SELECTION CODE -3
95	SPACE
96	OPERATOR SYSTEM ACCESS CODE -1
97	OPERATOR SYSTEM ACCESS CODE -2
98	SPACE
99	TIME IN QUEUE-MINUTE TENS
100	TIME IN QUEUE-MINUTE UNITS
101	SPACE
102	EXTERNAL LINE ID -1 / CONNECTED NUMBER -1
103	EXTERNAL LINE ID -2 / CONNECTED NUMBER -2
104	EXTERNAL LINE ID -3 / CONNECTED NUMBER -3
105	EXTERNAL LINE ID -4 / CONNECTED NUMBER -4
106	EXTERNAL LINE ID -5 / CONNECTED NUMBER -5

107	EXTERNAL LINE ID -6 / CONNECTED NUMBER -6
108	EXTERNAL LINE ID -7 / CONNECTED NUMBER -7
109	EXTERNAL LINE ID -8 / CONNECTED NUMBER -8
110	EXTERNAL LINE ID -9 / CONNECTED NUMBER -9
111	CARRIAGE RETURN*
112	LINE FEED*
113	NULL*
114	NULL*
115	NULL*

**Note:**

The end of line sequence is controlled by the *callinfo_output_set* command.

The dialed number field is a shared field that presents either the dialedNumber or the opQueTime information.

The external line id/ connected number field is a shared field between ogTrnkId and ConnectedNumber. It is used if the answering party differs from the dialed party, in the case of call diversion, call to a PBX group number, or call to a common bell number. If the call was an abandoned call, and the ringing or queuing party number differs from the dialed party number, the ringing or queuing party number is recorded in this field. When this field is used for recording the answering party, any unused positions are filled with spaces.

For calls answered by direct call pickup, the answering party will be recorded in the outgoing external line ID field.

For all the diversion calls, the dialed number will be recorded in the dialed number field.

9.6 ASBU MDFP15 FORMAT

00	SPACE
01	SPACE
02	SPACE
03	TIME-HR.TENS
04	TIME-HR.UNITS
05	TIME-MINUTE.TENS
06	TIME-MINUTE.UNITS
07	SPACE
08	DURATION-HOUR
09	DURATION-MINUTE.TENS
10	DURATION-MINUTE.UNITS
11	DURATION-MINUTE TENTH / SECOND.TENS
12	SPACE
13	CONDITION CODE
14	SPACE
15	ACCESS CODE 1 -2

16	ACCESS CODE 1 -1
17	ACCESS CODE 1 -3
18	ACCESS CODE 2 -1
19	ACCESS CODE 2 -2
20	ACCESS CODE 2 -3
21	SPACE
22	DIALLED NUMBER -1
23	DIALLED NUMBER -2
24	DIALLED NUMBER -3
25	DIALLED NUMBER -4
26	DIALLED NUMBER -5
27	DIALLED NUMBER -6
28	DIALLED NUMBER -7
29	DIALLED NUMBER -8
30	DIALLED NUMBER -9
31	DIALLED NUMBER -10
32	DIALLED NUMBER -11

33	DIALLED NUMBER -12
34	DIALLED NUMBER -13
35	DIALLED NUMBER -14
36	DIALLED NUMBER -15
37	DIALLED NUMBER -16
38	DIALLED NUMBER -17
39	DIALLED NUMBER -18
40	DIALLED NUMBER -19
41	DIALLED NUMBER -20
42	SPACE
43	CHARGED NUMBER -1
44	CHARGED NUMBER -2
45	CHARGED NUMBER -3
46	CHARGED NUMBER -4
47	CHARGED NUMBER -5
48	SPACE
49	ACCOUNT CODE -1

50	ACCOUNT CODE -2
51	ACCOUNT CODE -3
52	ACCOUNT CODE -4
53	ACCOUNT CODE -5
54	ACCOUNT CODE -6
55	ACCOUNT CODE -7
56	ACCOUNT CODE -8
57	ACCOUNT CODE -9
58	ACCOUNT CODE -10
59	ACCOUNT CODE -11
60	ACCOUNT CODE -12
61	ACCOUNT CODE -13
62	ACCOUNT CODE -14
63	ACCOUNT CODE -15
64	SPACE
65	CILCODE (Authorization code) -1
66	CILCODE (Authorization code) -2

67	CILCODE (Authorization code) -3
68	CILCODE (Authorization code) -4
69	CILCODE (Authorization code) -5
70	CILCODE (Authorization code) -6
71	SPACE
72	EXTERNAL LINE ID -1 / CONNECTED NUMBER -1
73	EXTERNAL LINE ID -2 / CONNECTED NUMBER -2
74	EXTERNAL LINE ID -3 / CONNECTED NUMBER -3
75	EXTERNAL LINE ID -4 / CONNECTED NUMBER -4
76	EXTERNAL LINE ID -5 / CONNECTED NUMBER -5
77	EXTERNAL LINE ID -6 / CONNECTED NUMBER -6
78	EXTERNAL LINE ID -7 / CONNECTED NUMBER -7
79	EXTERNAL LINE ID -8 / CONNECTED NUMBER -8
80	EXTERNAL LINE ID -9 / CONNECTED NUMBER -9
81	CARRIAGE RETURN*

82	LINE FEED*
83	NULL*
84	NULL*
85	NULL*

**Note:**

The end of line sequence is controlled by the *callinfo_output_set* command.

For calls answered by direct call pickup, the answering party will be recorded in the outgoing external line ID field.

For all the diversion calls, the dialed number will be recorded in dialed number field.

9.7 HEARTBEAT FORMAT

00	SPACE
01	YEAR TENS
02	YEAR UNITS
03	MONTH TENS
04	MONTH UNITS
05	DAY TENS
06	DAY UNITS
07	SPACE

08	HOUR TENS
09	HOUR UNITS
10	COLON
11	MINUTE TENS
12	MINUTE UNITS
13	SPACE
14	H
15	E
16	A
17	R
18	T
19	B
20	E
21	A
22	T
23	SPACE
24	F

25	R
26	O
27	M
28	SPACE
29	EXCHANGE ID(0)
30	EXCHANGE ID(1)
31	EXCHANGE ID(2)
32	EXCHANGE ID(3)
33	EXCHANGE ID(4)
34	EXCHANGE ID(5)
35	EXCHANGE ID(6)
36	EXCHANGE ID(7)
37	EXCHANGE ID(8)
38	EXCHANGE ID(9)
39	EXCHANGE ID(10)
40	EXCHANGE ID(11)
41	EXCHANGE ID(12)

42	EXCHANGE ID(13)
43	EXCHANGE ID(14)
44	CARRIAGE RETURN*
45	LINE FEED*
46	NULL*
47	NULL*
48	NULL*

**Note:**

The end of line sequence is controlled by the *callinfo_output_set* command.

9.8 DEMO1 FORMAT

See [General Format](#) example for demo1.

Call list excluding emergency calls

2004-10-07 08:23:01 0d00:00:01 from 1234 to 2345

2004-10-07 08:30:22 0d00:00:05 from 2345 to 1234

2004-10-07 08:33:01 0d00:01:23 from 1235 to 2345

End of list.

9.9 DEMO2 FORMAT

See [General Format](#) example for demo2.

Call list demo 2

From 1234 dialed 2345 to 2346 duration 36 charged 1234

From 2004-10-07 08:23:01 to 2004-10-07 08:23:37

From 2344 dialed 3455 to 3455 duration 11 charged 2344

From 2004-10-07 08:33:44 to 2004-10-07 08:33:55

End of list.

9.10 MOBILITY FORMAT

See [General Format](#) example for demo2. Mobility logging

2013-10-30 11:45:47 303 LRS 123A-1-50 1 Location Registration, Successful

This chapter contains the following sections:

- [CIL DATA FIELDS](#)
- [QOS FIELDS](#)
- [MOBILITY LOGGING FIELDS](#)

The call logging records can contain traffic log data or mobility log data.

Traffic log data, generated by calls made in the system, may contain a number of different data fields generated during a call, regarding A-party, B-party numbers, dialed digits, and so on.

Mobility log data contains data generated by the mobility functionality (for example by CXN or DECT) regarding events like logon, handover, location updates, and so on.

The field "isMobileLogging" determines if the call record contains call data or mobility data. If "isMobileLogging" is false (0) then the data record is generated by a call and will contain CIL data fields. If the field "isMobileLogging" is true, then it will contain Mobile Logging data.

Some of the fields described are really different representation of the same data, and can be used in the different aspects in the script language. In the predefined formats only one of the representations might be present in the output.

10.1 CIL DATA FIELDS

ACCESSCODE1 AND ACCESSCODE2

Table 2: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
3	5	5	3	0-5

This entry is blank for all internal and incoming calls unless the calls (outgoing or incoming) are made through an ISDN Call by Call service.

The field is right adjusted with blanks filling the unused character positions. Two different cases can be distinguished as follows:

- **The call is placed without LCR** . The access code 1 entry contains the access code dialed by the user. The access code 2 entry contains the destination code of the external line group actually used for the call if it is different from the access code dialed by the users.

- **The call is placed with LCR** . The access code 1 contains the LCR access code. The access code 2 contains either the dialed number or the fictitious destination code for the selected route, which depends on an MDP value per exchange.

In a Coordinated Dialing Plan (CDP) environment the destination code portion of the network number as dialed is placed in the Access Code 1 field. Note that the destination code in the CDP environment will be equal to the first one to three digits of the dialed number. For example, if digits 50 are the destination code digits in a CDP environment, and directory number 5037 is dialed, the access code 1 field would contain 50 and the dialed number field would contain 5037.

10.1.1 ACCOUNTCODE

Table 3: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
15 (shared)	15	15	15	0-15

This field is left adjusted with unused positions filled with blanks. When no data is recorded, the field will contain blanks. The system stores a maximum of 10 digits.

10.1.2 ACCOUNTCODEVALID

The field indicates if the accountCode contains data or not. True = 1, false = 0.

10.1.3 CALLCASEDATA

This field will contain a bit mapped integer with call conditions used in this call.

0 = No call conditions

1 = Free of charge call

10.1.4 CALLINGNUMBER

Table 4: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
4	10	10	5	1-20

This number is the directory number of the originator if the origination was from an extension or an operator console. If the originator's number is signalled through the network, then this number will be recorded as the calling number.

This field is left adjusted. Trailing blanks are used if the calling number is less than the maximum number of digits. If the field is defined as being less than 6 digits wide, some information will be lost, with the right most digits of the field being truncated.

When a change in the calling party number is received from the private network or from the VPN and own node has a public outgoing external line connected, as in the case of a transfer occurring on another node over the network, the existing call record is output and a new call record with the updated calling party number is created.

For the incoming external line calls where no calling party number is available, this field carries the external line ID that indicates which external line carried the incoming call. If the private network number and public network number are also available in addition to the external line ID, the priority to record one of the calling party numbers in this field is as following:

- Private network number (highest priority)
- Public network number
- External line ID

When general formats are used, for incoming calls where no calling party number is available the calling number field is left blank. The incoming trunk ID field (if used) will contain the incoming trunk ID.

If the call originates from an ADN number, then the calling number field will contain the ODN number on which the ADN is initiated.

10.1.5 CALLINGNUMBERTYPE

Table 5: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-1

Type of A-party number:

0 = Directory number or private network number

1 = Trunk ID

2 = Public network number

10.1.6 CALLINGNUMBERVALID

Field indicating if callingNumber contains data or not. True = 1, false = 0.

10.1.7 CALLID

Table 6: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-24

String.Unique call reference identifier. Will identify a call within a network.

10.1.8 CALLIDASS1

Table 7: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-24

String.Associated call reference identifier 1. Will link call records together,when services are used (Call transfer etc.).

10.1.9 CALLIDASS2

Table 8: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-24

String.Associated call reference identifier 2. Will link call records together,when services are used (Call transfer etc.).

10.1.10 CHARGEDNUMBER

Table 9: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-4	-10	-10	-5	0-20

The charged number is the number of the party which is charged for the call. In most cases the calling number and the charged number are the same but they can be different in certain cases such as ECF, RID pub, and so on.

For those incoming external public calls, if Collect Call option is available then the dialed and answering party is the charged party.

10.1.11 CILCODE

Table 10: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
15 (shared)	6	6	6	0-20

This field contains the fictitious CIL number that is associated with the Authorization code. This field is left adjusted with unused positions filled with blanks. When no data is recorded, the field will contain blanks.

The system stores a maximum of 20 digits. When an output format supports a lower number of digits, the **least significant** digits will be truncated.

10.1.12 CILCODEVALID

Field indicating if cilCode is containing data or not. True = 1, false = 0.

10.1.13 CONDITIONCODE

Table 11: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-3

Condition codes are assigned to a call record based on the type of call and priority. Some calls will perform more than one service. In this case the condition code with the highest priority is presented. When two condition codes have the same priority, the last condition applicable to the traffic case is used.

Priority to call type:

- 1 Extremely long call duration
- 2 Malicious call tracing call (MCT)
- 2 Malicious call tracing call (MCT)
- 3 External follow me call (ECF), Mobile extension via SIP trunk (same code as ECF), Call terminated due to route optimization, Abandoned incoming call, Abandoned outgoing call, Call to busy party,

- Call to vacant number, RID to a public Line, Conference call, Deflected to a public line, Rerouted calls, Transfer call.
- 4 Least cost routing call (LCR)
 - 5 Call handled by PBX operator
 - 6 Group call pickup, Group hunt calls, Call that has been connected with alternative route selection, Recall to route, Incoming call or tandem call, Internal call, Outgoing call, Intrusion, Call established due to route optimization, Answering party differs from the called party.

This is an integer, where the value range is 0-255.

For an explanation of how call types are used in the three most significant bits of this integer, [See “The Segment Part of the Condition Code \(Type of Termination\)”](#)

For explanation of how call types are used in the four least significant bits of this integer [See “The Position Part of the Condition Code \(Origin of Call Data\)”](#).

The condition code is also used to create printable data for different formats, [See “Condition Codes for Ordinary Calls”](#).

[See “Condition Codes for Long-Duration Calls”](#).

[See “Condition Codes for: The Answering Party is Different from the Dialed Party”](#).

[See “Condition Codes for Group Hunting Calls”](#).

[See “Condition Codes for Data Calls”](#).

[See “Condition Codes for Group Call Pick-Up Calls”](#).

[See “Condition Codes for Direct Diversion Calls”](#).

[See “Condition Codes for Diversion on No-Answer Calls”](#).

10.1.14 CONDITIONCODESEG

Table 12: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-1

Table 13: The Segment Part of the Condition Code (Type of Termination)

0	Ordinary calls
---	----------------

1	Long duration calls
2	Answering party differs from the called party
3	Group hunting calls
4	Data call (Obsolete for internal calls)
5	Group call pickup calls
6	Direct diversion calls
7	Diversion on no answer

10.1.15 CONDITIONCODEPOS

Table 14: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	-

Table 15: The Position Part of the Condition Code (Origin of Call Data)

1	Call handled by PBX operator
5	Call that has been connected with alternative route selection
6	Recall to route
7	Incoming call or tandem call
8	Internal call

CALL LOGGING RECORD FIELDS

9	Conference call
10	Outgoing call
11	Congestion
12	Facility restricted
13	Intrusion
15	Transfer call
16	Call established due to route optimization
17	External follow me call (ECF) and Mobile extension via SIP trunk
18	Answering party differs from the called party
19	Least cost routing call (LCR)
20	DISA
21	Malicious call tracing call (MCT)
22	Call terminated due to route optimization
23	Abandoned incoming call
24	DISA abandoned
25	Abandoned internal call
26	(Obsolete) Dynamic route allocation call, DRA

27	SMS
28	Abandoned outgoing call
29	Call to busy party
30	Call to vacant number
31	Undelivered call (to CTI- or GH-group)

10.1.16 CONDITIONCODE1CHARACTER, CONDITIONCODE2CHARACTER, CONDITIONCODE3CHARACTER

Table 16: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
1	2	2	1	0-3

These fields contain the result of a translation from the condition code to readable format.

Table 17: Condition Codes for Ordinary Calls

Type of call	Condition	condition CodeSeg	condition CodPos	CC1	CC2	CC3
Call handled by PBX operator	1	0	1	A	A	A
Alternative route selection	5	0	5	G	G	G
Recall to route	6	0	6	H	H	H
Incoming call or tandem call	7	0	7	I	I	I

CALL LOGGING RECORD FIELDS

Type of call	Condition	condition CodeSeg	condition CodPos	CC1	CC2	CC3
Internal call	8	0	8	J	J	J
Conference call	9	0	9	L	L	L
Outgoing call	10	0	10			
Intrusion	13	0	13	R	R	R
Data call	14	0	14	V	V	V
Transfer call	15	0	15	T	T	T
Route optimization established. call	16	0	16	Y	Y	Y
External follow me (ECF) call and Mobile extension via SIP trunk	17	0	17	X	X	X
Least Cost Routing (LCR) call	19	0	19	M	M	M
DISA	20	0	20	Z	Z	Z
Malicious Call Tracing (MCT) call	21	0	21	Q	Q	Q
Route optimization terminated call	22	0	22	W	W	W
Abandoned incoming call	23	0	23	C	CI	CI
DISA abandoned	24	0	24	C	CZ	CZ
Abandoned internal call	25	0	25	C	CJ	CJ

Type of call	Condition	condition CodeSeg	condition CodPos	CC1	CC2	CC3
Dynamic Route Allocation (DRA) call. (Obsolete)	26	0	26	S	S	S
SMS	27	0	27	P	P	P
Abandoned outgoing call	28	0	28	C	CO	CO
Calls to busy party	29	0	29	B	B	B
Calls to vacant number	30	0	30	K	K	K
Undelivered call	31	0	31	U	U	U

Table 18: Condition Codes for Long-Duration Calls

Type of call	conditionCode	condition CodeSeg	condition Code POS	CC1	CC2	CC3
Call handled by PBX operator	32	1	1	D	DA	DA
Alternative route selection	37	1	5	D	DO	DO
Recall to route	38	1	6	D	DO	DO
Incoming call or tandem call	39	1	7	D	DI	DI
Internal call	40	1	8	D	DJ	DJ
Conference call	41	1	9	D	D	D
Outgoing call	42	1	10	D	DO	DO
Intrusion	45	1	13	D	D	D

CALL LOGGING RECORD FIELDS

Type of call	conditionCode	condition CodeSeg	condition Code POS	CC1	CC2	CC3
Data call	46	1	14	D	DV	DV
Transfer call	47	1	15	D	D	D
Route optimization established. call	48	1	16	D	D	D
External follow me (ECF) call and Mobile extension via SIP trunk	49	1	17	D	DX	DX
Least Cost Routing (LCR) call	51	1	19	D	DO	DO
DISA	52	1	20	D	D	D
Malicious Call Tracing (MCT)	53	1	21	D	D	D
Route optimization terminated call	54	1	22	D	D	D
Abandoned incoming call	55	1	23	C	DI	DI
DISA abandoned	56	1	24	D	DZ	DZ
Abandoned internal call	57	1	25	D	DJ	DJ
Dynamic Route Allocation, DRA, call. (Obsolete)	58	1	26	D	DO	DO
SMS	59	1	27	D	D	D
Abandoned outgoing call	60	1	28	D	DO	DO
Calls to busy party	61	1	29	D	DB	DB

Type of call	conditionCode	condition CodeSeg	condition Code POS	CC1	CC2	CC3
Calls to vacant numbers	62	1	30	D	DK	DK
Undelivered call	63	1	31	D	DU	DU

Table 19: Condition Codes for: The Answering Party is Different from the Dialed Party

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Call handled by PBX operator	65	2	1	A	NA	NA
Alternative route selection	69	2	5	N	NG	NG
Recall to route	70	2	6	N	NH	NH
Incoming call or tandem call	71	2	7	N	NI	NI
Internal call	72	2	8	N	NJ	NJ
Conference call	73	2	9	N	NL	NL
Outgoing call	74	2	10	N	NO	NO
Intrusion	77	2	13	N	N	N
Data call	78	2	14	N	NV	NV
Transfer call	79	2	15	T	NT	NT
Route optimization established. call	80	2	16	Y	Y	Y
External follow me (ECF) call and Mobile extension via SIP trunk	81	2	17	X	NX	NX

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Least Cost Routing (LCR) call	83	2	19	M	M	M
DISA	84	2	20	Z	NZ	NZ
Malicious Call Tracing (MCT) call	85	2	21	Q	Q	Q
Route optimization terminated call	86	2	22	W	W	W
Abandoned incoming call	87	2	23	C	NC	NCI
DISA abandoned	88	2	24	C	NC	DCZ
Abandoned internal call	89	2	25	C	NC	NCJ
Dynamic Route Allocation, DRA, call. (Obsolete)	90	2	26	S	S	S
SMS	91	2	27	P	P	P
Abandoned outgoing call	92	2	28	C	NC	NCO
Calls to busy party	93	2	29	B	NB	NB
Calls to vacant numbers	94	2	30	K	NK	NK
Undelivered call	95	2	31	U	NU	NU

Table 20: Condition Codes for Group Hunting Calls

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Call handled by PBX operator	97	3	1	A	FA	FA

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Alternative route selection	101	3	5	F	FG	FG
Recall to route	102	3	6	F	FH	FH
Incoming call or tandem call	103	3	7	F	FI	FI
Internal call	104	3	8	F	FJ	FJ
Conference call	105	3	9	F	FL	FL
Outgoing call	106	3	10	-	FO	FO
Undelivered call (No member)	107	3	11	U	UA	UA
Intrusion	109	3	13	N	N	N
Data call	110	3	14	F	FV	FV
Transfer call	111	3	15	T	FT	FT
Route optimization established. call	112	3	16	Y	Y	Y
External follow me (ECF) call and Mobile extension via SIP trunk	113	3	17	X	FX	FX
Least Cost Routing (LCR) call	115	3	19	M	M	M
DISA	116	3	20	Z	Z	Z
Malicious Call Tracing (MCT) call	117	3	21	Q	Q	Q
Route optimization terminated call	118	3	22	W	W	W

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Abandoned incoming call	119	3	23	C	FC	FCI
DISA abandoned	120	3	24	C	FC	FCZ
Abandoned internal call	121	3	25	C	FC	FCJ
Dynamic Route Allocation, DRA, call. (Obsolete)	122	3	26	S	S	S
SMS	123	3	27	P	P	P
Abandoned outgoing call	124	3	28	C	FC	FCO
Calls to busy party	125	3	29	B	FB	FB
Calls to vacant numbers	126	3	30	K	FK	FK
Undelivered call (No answer)	127	3	31	U	U	U

Table 21: Condition Codes for Data Calls

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Call handled by PBX operator	129	4	1	V	VA	VA
Alternative route selection	133	4	5	V	VO	VO
Recall to route	134	4	6	V	VO	VO
Incoming call or tandem call	135	4	7	V	VI	VI
Internal call	136	4	8	V	VJ	VJ

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Conference call	137	4	9	V	V	V
Outgoing call	138	4	10	V	VO	VO
Intrusion	141	4	13	V	V	V
Data call	142	4	14	V	V	V
Transfer call	143	4	15	V	V	V
Route optimization established. call	144	4	16	V	V	V
External follow me (ECF) call and Mobile extension via SIP trunk	145	4	17	V	VX	VX
Least Cost Routing (LCR) call	147	4	19	V	VO	VO
DISA	148	4	20	V	V	V
Malicious Call Tracing (MCT) call	149	4	21	V	V	V
Route optimization terminated call	150	4	22	V	V	V
Abandoned incoming call	151	4	23	V	VI	VI
DISA abandoned	152	4	24	V	VZ	VZ
Abandoned internal call	153	4	25	V	VJ	VJ
Dynamic Route Allocation, DRA, call. (Obsolete)	154	4	26	V	V	V

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
SMS	155	4	27	V	V	V
Abandoned outgoing call	156	4	28	V	VO	VO
Calls to busy party	157	4	29	V	VB	VB
Calls to vacant numbers	158	4	30	V	VK	VK
Undelivered call	159	4	31	U	U	U

Table 22: Condition Codes for Group Call Pick-Up Calls

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Call handled by PBX operator	161	5	1	A	EA	EA
Alternative route selection	165	5	5	E	EG	EG
Recall to route	166	5	6	E	EH	EH
Incoming call or tandem call	167	5	7	E	EI	EI
Internal call	168	5	8	E	EJ	EJ
Conference call	169	5	9	E	EL	EL
Outgoing call	170	5	10	E	E	E
Intrusion	173	5	13	N	N	N
Data call	174	5	14	E	EV	EV

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Transfer call	175	5	15	T	ET	ET
Route optimization established. call	176	5	16	Y	Y	Y
External follow me (ECF) call and Mobile extension via SIP trunk	177	5	17	X	X	X
Least Cost Routing (LCR) call	179	5	19	M	M	M
DISA	180	5	20	Z	Z	Z
Malicious Call Tracing (MCT) call	181	5	21	Q	Q	Q
Route optimization terminated call	182	5	22	W	W	W
Abandoned incoming call	183	5	23	C	EC	ECI
DISA abandoned	184	5	24	C	EC	ECZ
Abandoned internal call	185	5	25	C	EC	ECJ
Dynamic Route Allocation, DRA, call. (Obsolete)	186	5	26	S	S	S
SMS	187	5	27	P	P	P
Abandoned outgoing call	188	5	28	C	EC	ECO
Calls to busy party	189	5	29	B	EB	EB
Calls to vacant numbers	190	5	30	K	EK	EK

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Undelivered call	191	5	31	U	EU	EU

Table 23: Condition Codes for Direct Diversion Calls

Type of call	condition Code	conditionC odeSeg	condition CodePos	CC1	CC2	CC3
Call handled by PBX operator	193	6	1	A	DA	DA
Alternative route selection	197	6	5	D	DG	NDG
Recall to route	198	6	6	D	DH	NDH
Incoming call or tandem call	199	6	7	D	DI	NDI
Internal call	200	6	8	D	DJ	NDJ
Conference call	201	6	9	D	DL	NDL
Outgoing call	202	6	10	D	DO	NDO
Intrusion	205	6	13	N	N	N
Data call	206	6	14	D	DV	NDV
Transfer call	207	6	15	T	DT	NDT
Route optimization established. call	208	6	16	Y	Y	Y
External follow me (ECF) call and Mobile extension via SIP trunk	209	6	17	X	DX	NDX
Least Cost Routing (LCR) call	211	6	19	M	M	M

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
DISA	212	6	20	Z	Z	Z
Malicious Call Tracing (MCT) call	213	6	21	Q	Q	Q
Route optimization terminated call	214	6	22	W	W	W
Abandoned incoming call	215	6	23	C	DC	DCI
DISA abandoned	216	6	24	C	DC	DCZ
Abandoned internal call	217	6	25	C	DC	DCJ
Dynamic Route Allocation, DRA, call. (Obsolete)	218	6	26	S	S	S
SMS	219	6	27	P	P	P
Abandoned outgoing call	220	6	28	C	DC	DCO
Calls to busy party	221	6	29	B	DB	DB
Calls to vacant numbers	222	6	30	K	DK	DK
Undelivered call	223	6	31	U	U	U

Table 24: Condition Codes for Diversion on No-Answer Calls

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Call handled by PBX operator	225	7	1	A	DA	DRA
Alternative route selection	229	7	5	D	DG	DRG

CALL LOGGING RECORD FIELDS

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
Recall to route	230	7	6	D	DH	DRH
Incoming call or tandem call	231	7	7	N	DI	DRI
Internal call	232	7	8	D	DJ	DRJ
Conference call	233	7	9	N	DL	DRL
Outgoing call	234	7	10	N	DO	DRO
Intrusion	237	7	13	N	N	N
Data call	238	7	14	N	DV	DRV
Transfer call	239	7	15	T	DT	DRT
Route optimization established. call	240	7	16	Y	Y	Y
External follow me (ECF) call and Mobile extension via SIP trunk	241	7	17	X	DX	DRX
Least Cost Routing (LCR) call	243	7	19	M	M	M
DISA	244	7	20	Z	Z	Z
Malicious Call Tracing (MCT) call	245	7	21	Q	Q	Q
Route optimization terminated call	246	7	22	W	W	W
Abandoned incoming call	247	7	23	C	DC	DCI

Type of call	condition Code	condition CodeSeg	condition CodePos	CC1	CC2	CC3
DISA abandoned	248	7	24	C	DC	DCZ
Abandoned internal call	249	7	25	C	DC	DCJ
Dynamic Route Allocation, DRA, call. (Obsolete)	250	7	26	S	S	S
SMS	251	7	27	P	P	P
Abandoned outgoing call	252	7	28	C	DC	DCO
Calls to busy party	253	7	29	B	DB	DB
Calls to vacant numbers	254	7	30	K	DK	DK
Undelivered call	255	7	31	U	U	U

10.1.17 CONDITIONCODEUSERDEFINED

Table 25: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-200

This field contains the translated strings from the condition code, set by command *call- info_condcode_set*. This field can be used to compose verbose formats of the printouts for greater readability. If a custom value is not set, the conditionCode3Character will be used.

10.1.18 CONNECTEDNUMBER

Table 26: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-8	8	8	0	0-20

The connected number is the answering party's number. In most cases the dialed and the connected number are the same, but they can be different due to number translation, diversion, and so on.

10.1.19 CSTCORRDATA

Table 27: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-20

Csta correlator data

10.1.20 CURRENTTIME

Table 28: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-20

This field contains the current time (UTC or LOCAL time, where default is UTC).

10.1.21 DIALEDNUMBER

Table 29: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
15	20	20	15	0-20

The dialed number is the called party's number which could be an external number for outgoing calls or an internal number when a call is internal or incoming.

Only the leading digits of the dialed number are recorded depending on the field width of the output format used.

The digits may include any access or destination code as dialed. If this access or destination code is subsequently extracted out of the dialed number string to be displayed separately, then only the remaining digits are displayed in the dialed number field. Any digits that cannot be displayed due to field width limitations are truncated and lost. This field has right adjusted format (Leading blanks are used).

When a Telephony Name Selection is used to place a call, any non-digit character which is programmed will be displayed in the CIL output just as it appears on the Telephony Name Selection key.

When dialing in the CDP environment, the complete number including the network destination code, is located in the dialed number field. The network destination code is also located in the Access Code 1 field as is described in the "Access Code 1 and 2" section, [See "accessCode1 and accessCode2"](#).

10.1.22 DURATION

Table 30: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
4	5	5	4	0-10

Call duration is the elapsed time between call answer and call termination. Answer time is defined by exactly one of the following:

- Answer detected by the MX-ONE Service Node
- End of selection of a external line

When the telephony application is unable to detect the answer signal from an external line, the call duration will begin upon external line seizure.

A call is terminated when either party hangs up in a two party call and when either of the last two parties hangs up in a conference call.

There are four or five characters used to display call duration depending on the format selected, see the chart above.

The four characters are defined as follows:

- One character for hour (0-9)
- One character for minutes TENS (0-5)
- One character for minutes UNITS (0-9)
- One character for tenth of minutes (0-9)

The five characters are defined as follows:

- One character for hour (0-9)
- One character for minutes TENS (0-5)
- One character for minutes UNITS (0-9)
- One character for tenth of minutes (0-9) or for seconds TENS (0-5)
- One character for space or for seconds UNITS (0-9)

The maximum duration for a call record is normally one unit less than 10 hours.

An MDP parameter determines the maximum call duration time for a call record. When a call duration exceeds this limit, a call record is generated and another call record is started. There is no continuity marking from one call record to its succeeding record. However, the dialed number, calling number, and access codes are the same. There are special condition codes attached to call records that exceed the maximum duration, and they are described in the “Condition Code” section, [See “conditionCode”](#).

Leading zeros are used for a call duration of less than one hour.


10.1.23 ENCRYPTEDMEDIA

Table 31: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	2-6

This is a left-adjusted field containing the information of the media encryption status. The values are: no, yes and partly. No means no endpoint is encrypted. Yes means all

endpoints are encrypted. Partly means one endpoint is encrypted, e.g. the call is gateway with one party using other (non-IP) media type, like ISDN.

 **Note:**
This data is only available in XML and general formats.

10.1.24 INCTRNKID

Table 32: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-10

This is a left-adjusted field containing the incoming external line ID for incoming calls, network transfer calls, and route optimization calls. The left most three digits identify the route, the middle three digits identify the LIM (computer), and the last four digits identify the external line number within the LIM.

For MX-ONE Version 4 the new trunkId format is RRRCCCXXLL, with

RRR = Route number	0 - 250
CCC = Computer (LIM number)	1 - 124
XXLL = Line number within the route and computer	1 - 1920

10.1.25 INCTRNKID9

Table 33: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-9

This is a left-adjusted field containing the incoming external line ID for incoming calls, network transfer calls, and route optimization calls. The left most three digits identify the route, the middle three digits identify the LIM, and the last three characters identify the external line number within the LIM.

The incTrunkId9 format is RRRCCCYLL, with

RRR = Route number	0 - 250
CCC = Computer (LIM number)	1 - 124

YLL = Line number within the route and computer	1 - 1920
-------------------------------------------------	----------

To be backwards compatible the left-most character, Y, in the line number part represents two digits according to the following table:

Table 34: Most significant trunk line number digit

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9
0x	0	1	2	3	4	5	6	7	8	9
1x	A	B	C	D	E	F	G	H	I	J
2x	K	L	M	N	O	P	Q	R	S	T
3x	U	V	W	X	Y	Z	@	+	\$	%
4x	a	b	c	d	e	f	g	h	i	j
5x	k	l	m	n	o	p	q	r	s	t
6x	u	v	w	x	y	z	#	&	<	>

Example 1:

Route = 102, Processor (LIM) = 66, Line number = 100 gives incTrunkId9 = 102066100, compared to the ten digits format incTrunkId = 1020660100

Example 2:

Route = 102, Processor (LIM) = 66, Line number = 1100 incTrunkId9 = 102066B00, compared to the ten digits format incTrunkId = 1020661100

Example 3:

Route = 102, Processor (LIM) = 66, Line number = 2300 incTrunkId9 = 102066N00, compared to the ten digits format incTrunkId = 1020662300

10.1.26 INCTRUNKIDVALID

Field indicating if incTrnkId contains data or not. 'True = 1, false = 0

10.1.27 OGTRNKID

Table 35: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	9 (shared)	-	0-10

This is a left-adjusted field containing the outgoing external line ID for outgoing calls, network transfer calls, and route optimization calls. The left most three digits identify the route, the middle three digits identify the LIM, and the last four digits identify the external line number within the LIM.

10.1.28 OGTRNKID9

Table 36: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	9 (shared)	-	0-9

This is a left-adjusted field containing the outgoing external line ID for outgoing calls, network transfer calls, and route optimization calls. The left most three digits identify the route, the middle three digits identify the LIM, and the last three characters identify the external line number within the LIM.

The ogTrunkId9 format is RRRCCCYLL, with

RRR = Route number	0 - 250
CCC = Computer (LIM number)	1 - 124
YLL = Line number within the route and computer	1 - 1920

To be backwards compatible the left-most character, Y, in the line number part represents two digits according to a conversion table. See table [Most significant trunk line number digit](#) and the description on how the trunk line number is formed.

10.1.29 OGTRNKVALID

Field indicating if ogkTrnkId is containing data or not. True = 1, false = 0

10.1.30 OPQUETIME

Table 37: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	(shared)	(shared)	-	0-2

This is a two-digit field that represents the amount of time that an incoming call waited in an operator queue.

An incoming external line call that has been placed in the operator queue will wait some number of seconds before the operator answers the call. That number of seconds is called the Operator Queue Time. The Operator Queue Time can have a value of up to 99 seconds. Any queue time longer than 99 seconds will be recorded as 99 seconds.

The default output format for this field is decimal, but it is possible to choose the output in hex- or time-format by command.

10.1.31 OPQUETIMEVALID

Field indicating if opQueTime contains data or not. True = 1, false = 0

10.1.32 OSACODE

Table 38: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	2	2	-	0-2

This is a left adjusted field.

This is an ISDN capability by which the call can be routed to a specific operator.

10.1.33 OSACODEVALID

Field indicating if osaCode contains data or not.

True = 1, false = 0.

10.1.34 QUEUETIMECOUNTER

Table 39: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	2	2	-	0-2

The maximum value of the queue time is 999 seconds. If the queue time is longer than the maximum value, it will be recorded as 999 seconds. If the time is less than one second, it will be recorded as zero seconds.

This is a field defining the queue time before the incoming external line call (including public external line and non-CCS tie line) or internal call is answered.

10.1.35 RINGTIMECOUNTER

Table 40: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-9

The maximum value of the ring time is 999 seconds. If the ring time is longer than the maximum value, it will be recorded as 999 seconds. If the time is less than one second, it will be recorded as zero seconds.

This is a field defining the ring time before the incoming external line call (including public external line and non-CCS tie line) or internal call is answered.

The called party for the incoming external line can be one of the following:


- Voice extension
- Operator
- ACD/CTI group
- PBX group/cascade ring group
- Outgoing trunk line

The called party for the internal call can be one of the following:

- Voice extension
- Operator
- ACD/CTI group
- PBX group/cascade ring group
- Outgoing trunk line

The ring time is calculated starting from the time the called party is alerted until the time the alerting is stopped. If there is more than one party alerted in the sequence before the call is answered, as in the case of call diversion at no answer, each duration of the ringing time is accumulated. This field is available in the CIL output record for the internal call or incoming external line call regardless if the call is answered or abandoned. When the called party is an outgoing trunk line the ring time is started when the line is seized.

The ring time and queue time can not be differentiated in a network, as in the case when the call is diverted to another node before it is answered. In this case, the ring time is assumed.

 **Note:**
When calling party line is in a “reserving” state, for example when an RVA welcome message is played, the call in “reserving” state is registered as abandoned call if calling party clears (i.e. no ring timer is started).

10.1.36 SEQLIM

Table 41: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-3


LIM number where the data was generated (combined with seqNumber it forms a data unit securing that no call logging records are lost unnoticed)

10.1.37 SEQNUMBER

Table 42: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-10

Sequence number is a 32-bit integer starting at 0. It is reset at lim or system restart. (Combined with seqLim it forms a data unit securing that no call logging records are lost unnoticed)

 **Note:**
The seqNumber in a call logging record counts up independently, that is, as two separate counters, carried in the same field.

10.1.38 STARTTIME

Table 43: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-10

Start time of the call. Time is expressed in 24-hour format.

10.1.39 STOPTIME

Table 44: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	-	-	0-10

Stop time of the call. Time is expressed in 24-hour format.

10.1.40 TAXPULSES

Table 45: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-4	-	-	0-4

Tax pulse is a four-character field that can output a value up to 9999. If a call exceeds 9999 tax pulses, the value 9999 will be output.

A prerequisite for this function is that the public exchange is capable of sending tax pulses to the MX-ONE.

10.1.41 TAXPULSESVALID

Field indicating if taxPulses contains data or not. True = 1, false = 0

10.1.42 TIMECOUNTERSVALID

Field indicating if ringTimeCounter or queTimeCounter contains data or not. True = 1, false = 0

10.1.43 TNSCODE

Table 46: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	3	3	-	0-4

This is an ISDN capability that identifies how a call was routed through a specific inter-exchange carrier.

10.1.44 TNSCODEVALID

Field indicating if tnsCode contains data or not. True = 1, false = 0

10.1.45 TRUNKQUETIME

Table 47: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	2	-	0-2

This field is a two character field defining the minutes in TENS and UNITS.

This field defines the time for which a call has been queued before a callback to a busy route or external line. The queue to a route or external line is applicable to on-hook queuing.

The maximum value is 59 minutes.

10.2 QOS FIELDS

The collection of QoS data is based on ITU-T rec.H.460.9 Annex A, with extended codec information.

Quality of Service data fields are only available when a TCP/IP-connected party is involved in the call. QoS information is not printed in subtypes fp15, mdfp15, asb501, and asbumdfp15. Select one of the subtypes postgresql, commaSeparated, xml, or general when this information is required.

Each call can have up to three TCP/IP hops in a gateway scenario. Each call can thus have three QoS-data each with two endpoints, totalling six endpoints per call record.

The data here is described per field where **x** is the endpoints index in the call.

If the field "qosDataValid_x" is true then the data record contains quality of service data.

10.2.1 QOSENDPOINTVALID_X

Is this endpoint valid or not. True = 1, false = 0

10.2.2 RTPADDRESS_X

The TCP/IP address of A-party's H.225.0 Real Time Protocol (RTP) connection.

10.2.3 ENDPOINTTYPE_X, ENDPOINTTYPENAME_X, ENDPOINTTYPE- DESC_X

EndpointType_x is an integer describing the endpoint type. EndpointTypeName_x is a terse string describing the endpoint type. EndpointTypeDesc_x is an expanded string describing the endpoint type.

Table 48: Endpoint Types

endpointType	endpointTypeName	endpointTypeDesc
0	a_ep	A-party terminal end point
1	a_eemg	A-party, Media Gateway
2	a_gw	A-party, other gateway
3	a_multiLim	A-party, Multi LIM
4	a_operator	A-party, operator
5	a_voiceMail	A-party, voice mail
6	a_voiceMenu	A-party, voice menu, ETSI*/ Stored Voice Services
7	a_vaa	A-party, voice-activated attendant, voice dialing
8	a_hold	A-party, music-on-hold or other hold service

endpointType	endpointTypeName	endpointTypeDesc
9	a_remote	A-party, remote party
10	a_other	A-party, other
11	a_spare_b	A-party, spare
12	a_spare_c	A-party, spare
13	a_spare_d	A-party, spare
14	a_spare_e	A-party, spare
15	a_spare_f	A-party, spare
16	b_ep	B-party terminal end point
17	b_eemg	B-party, Media Gateway
18	b_gw	B-party, other gateway
19	b_multiLim	B-party, Multi LIM
20	b_operator	B-party, operator
21	b_voiceMail	B-party, voice mail
22	b_voiceMenu	B-party, voice menu, ETSI*/ Stored Voice Services
23	b_vaa	B-party, voice-activated attendant, voice dialing
24	b_hold	B-party, music-on-hold or other hold service

endpointType	endpointTypeName	endpointTypeDesc
25	b_remote	B-party, remote party
26	b_other	B-party, other
27	b_spare_b	B-party, spare
28	b_spare_c	B-party, spare
29	b_spare_d	B-party, spare
30	b_spare_e	B-party, spare
31	b_spare_f	B-party, spare
32	o_ep	Other party terminal end point
33	o_eemg	Other party, Media Gateway
34	o_gw	Other party, other gateway
35	o_multiLim	Other party, Multi LIM
36	o_operator	Other party, operator
37	o_voiceMail	Other party, voice mail
38	o_voiceMenu	Other party, voice menu, ETSI*/ Stored Voice Services
39	o_vaa	Other party, voice-activated attendant, voice dialing
40	o_hold	Other party, music-on-hold or other hold service

endpointType	endpointTypeName	endpointTypeDesc
41	o_remote	Other party, remote party
42	o_other	Other party, other
43	o_spare_b	Other party, spare
44	o_spare_c	Other party, spare
45	o_spare_d	Other party, spare
46	o_spare_e	Other party, spare
47	o_spare_f	Other party, spare
48	otherKind	Other kind of endpoint

10.2.4 EXTENSION_X

String identifying the A-party, number or internet address.

10.2.5 QOSDATAVALID_X

Field indicating if the endpoint delivers QoS measurement or not. True = 1, false = 0

10.2.6 ESTIMATEDTHROUGHPUT_X

Senders packet count reduced with the packets lost, multiplied with the average packet size, per second.

10.2.7 FRACTIONLOSTRATE_X

Sum of fraction lost per second

10.2.8 CODECNAME1_X, CODECNAME2_X, CODECNAME3_X

Codec numbers used by party x. Values for audio types defined as below. Other codec types not yet defined.

Table 49: Audio Codec Types as per ITU-T rec. H.323 annex F

Codec type	X-bit	Au-Type
G.711 A-law 64kbps	0	1
G.711 A-law 56kbps	0	2
G.711 my-law 64kbps	0	3
G.711 my-law 56kbps	0	4
G.729	0	10
Annex A/G.729	0	11
GSM full rate	1	3
GSM half rate	1	4
GSM Enhance Full Rate	1	5

Audio codec numbering according to ITU-T rec. H.323 annex F. Bits => 0xaa aaaa. x = X-bit, aa aaaa = AuType

10.2.9 CODECTYPE1_X, CODECTYPE2_X, CODECTYPE3_X

Character indicating type of codec used by this party. Values: S = speech, V = video, D = data, O = other or I = invalid.

10.2.10 CUMULATIVENUMBEROFPACKETSLOST_X

The number of cumulative packets lost of Real Time Transmission Control Protocol (RTCP) receiver reports

10.2.11 MEANESTIMATEDEND2ENDDELAY_X

Mean value of round trip measurement divided by 2. (single trip delay)

The value is in 32-bit NTP format (the middle 32 bits of the 64-bit NTP timestamp). To calculate the value in milliseconds:

$$\text{time_in_milliSec} = (\text{meanEstimatedEnd2EndDelay_x} * 1000) / 65535$$

10.2.12 MEANJITTER_X

Average value of jitter

The value is in 32-bit NTP format (the middle 32 bits of the 64-bit NTP timestamp). To calculate the value in milliseconds:

$$\text{time_in_milliSec} = (\text{meanJitter_x} * 1000) / 65535$$

10.2.13 PACKETLOSTRATE_X

Number of packets lost per second

10.2.14 SIMPLER_VALUE_X

Calculated relative quality of service value. Integer with fixed 1 digit decimal point. Range 0-1 000 (00.0%-100.0%).

The value is a simplified value calculated according to ITU-T G.107 and G.113 annex I, and is taking the effects of network and transmission into account, apart from the audio transmission characteristics.

$$R = R_o - I_s - I_d - I_e$$

which is a simplified to $R = 93.2 - I_{dd} - I_e(\text{eff})$

93.2 is a calculated constant.

I_{dd} is an impairment based on delay. 00 = 000 - 150ms 01 = 151 - 175ms 03 = 176 - 200ms 06 = 201 - 225ms 09 = 226 - 250ms 15 = 251 - 300ms 20 = 301 - 350ms 24 =

351 - 400ms 31 + (3 * every 100 ms) for values above 500 ms $I_e(\text{eff}) = I_{ec} + ((95 - I_{ec}) * P_{pl} / (P_{pl} + B_{pl}))$

lec is the Equipment Impairment factor which is Codec dependent. 00 = g.711, 05 = GSM, 11 = g.728, g.729, 15 = g.723

Bpl is The Codecs Packet Loss Robustness.

4.30 = g.711, 10.0 = GSM, 16.1 = g.723, 19.0 = g.728, g.729

Ppl is PacketLostRate.

10.2.15 WORSTESTIMATEDEND2ENDDELAY_X

Peak value of round trip measurement divided by 2. (single trip delay)

The value is in 32-bit NTP format (the middle 32 bits of the 64-bit NTP timestamp). To calculate the value in milliseconds:

$\text{time_in_milliSec} = (\text{worstEstimatedEnd2EndDelay_x} * 1000) / 65535$

10.2.16 WORSTJITTER_X

Peak value of jitter

The value is in 32-bit NTP format (the middle 32 bits of the 64-bit NTP timestamp). To calculate the value in milliseconds:

$\text{time_in_milliSec} = (\text{worstJitter_x} * 1000) / 65535$

10.3 MOBILITY LOGGING FIELDS

If the field "isMobileLogging" is true then the data record is generated by a mobility event and will contain data as below.

Mobility logging fields are not printed in subtypes fp15, mdfp15, asb501 and asbumd- fp15. Select one of subtypes postgresql, commaSeparated, xml, and general when this information is required.

10.3.1 CALLINGNUMBER

This number is the directory number that generates the mobility event.

This field is left adjusted. Trailing blanks are used if the calling number is less than the maximum number of digits.

10.3.2 EVENTLEVEL

Numeric field for mobilityEvent.

10.3.3 EQU

The equipment position used in the mobility logging to indicate where a party is currently located.

This field is a twelve character field. If LIM number is less than three digits, the unused positions will be filled with blanks. If no information about EQU exists, these 8 positions will be filled with blanks.

Example: 3-0-70-00

10.3.4 EQU LIM

The LIM number part of the equ where a party is currently located.

This field is a three character field. If the LIM number is less than three digits, the unused positions will be filled with blanks. If no information exists, the position will be left blank.

10.3.5 EQU MAGAZINE

The magazine position part of the equ where a party is currently located.

This field is a one character field. If no information exists, the position will be left blank.

10.3.6 EQU BOARD POSITION

The board position part of the equ where a party is currently located.

This field is a two character field. If no information exists, the position will be left blank.

10.3.7 EQU BOARD INDIVIDUAL

The board individual part of the equ where a party is currently located.

This field is a two character field. If no information exists, the position will be left blank.

10.3.8 EQU VALID

Field indicating if equ (equLim, equMagazine, equBoardPosition and equBoardIndividual) is containing data or not.

True = 1, false = 0

10.3.9 IS MOBILE LOGGING

Field indicating if this call record is produced by traffic or by mobility events. True = 1, false = 0

10.3.10 MOBILITYEVENT

This field is a three character field defining the event level by abbreviations.

Table 50: Mobility Events

Characters	Value	Explanation
Axx	20	Abnormal call release, release reason xx. See GDI Interface Description for values for CXN
BCA	18	Call from CXN
BCB	17	Call to CXN
CHS	11	Connection Handover, Successful
CHU	12	Connection Handover, Unsuccessful
DS	3	Detach, Successful
DU	4	Detach, Unsuccessful
EHS	13	External Handover, Successful
EHU	14	External Handover, Unsuccessful
HFS	15	Handover Fallback, Successful
HFU	16	Handover Fallback, Unsuccessful
LRU	1	Location Registration, Successful

Characters	Value	Explanation
LRS	2	Location Registration, Unsuccessful
TNA	19	Temporarily Not Available

10.3.11 SEQLIM

Table 51: Number of characters in different output formats

fp15	mdfp15	asb501	asbumdfp15	Other
-	-	4	-	0-3

Lim number where the data was generated (combined with seqNumber it forms a data unit securing that no call logging records are lost unnoticed).

10.3.12 STOPTIME

This is a four to six character field defining the time the call was terminated. Two characters define the hour in tens and units and two characters define the minutes in tens and units. In the General format there are two variants of this field, with or without seconds. If the field with seconds is used the last two characters define the seconds in tens and units. Time is expressed in 24 hour format.

The field is used to store the time of the event.

LOGGING OF UNCOMPLETED CALLS 11

This chapter contains the following sections:

- [LOGGING OF CALLS TO VACANT NUMBER](#)
- [LOGGING OF CALLS TO BUSY PARTY](#)

LOGGING OF ABANDONED CALL

The incoming external, internal, and outgoing calls which are abandoned during queuing or ringing can be logged. Also if the calls fail due to the ringing time-out, they will be logged.

11.1 LOGGING OF CALLS TO VACANT NUMBER

All incoming and outgoing calls to Vacant Numbers are logged.

11.2 LOGGING OF CALLS TO BUSY PARTY

Calls to a busy party are as follows:

- Calls to an individual user which is not idle and the call is not queued.
- Calls to a group which has no available free members and no free queue records are available.

All internal, incoming, and outgoing calls which meet a busy condition will be logged.

INTERACTION WITH OTHER FEATURES

12

This chapter contains the following sections:

- ACCOUNT CODE
- ALTERNATIVE ROUTING
- AUTHORIZATION CODE
- ADDITIONAL NUMBER
- CALLBACK
- CALL DEFLECTION, SINGLE-STEP TRANSFER
- CALL METERING
- CALL WAITING
- COLLECT CALL
- CONFERENCE
- COORDINATED DIALING PLAN CALLS
- DIRECT CALL PICKUP
- DIVERSION
- DTS
- DYNAMIC ROUTE ALLOCATION
- EMERGENCY EXTENSION CALL
- EXTERNAL FOLLOW-ME, ECF
- FOLLOW-ME
- GROUP CALL PICKUP
- GROUP HUNTING OR CASCADE RING GROUP
- INTERNAL TRANSFER FROM VOICE TO DATA CALL
- INTRUSION
- ISDN CALL BY CALL
- LEAST COST ROUTING
- MALICIOUS CALL TRACE
- MOBILITY EVENTS
- NETWORK CALLS
- NETWORK DIVERSION
- NETWORK REROUTING
- NETWORK TRANSFER
- OPERATOR
- RECALL AS A RESULT OF EXTENDING
- EXTENDING OF EXTERNAL LINE TO EXTERNAL LINE TRAFFIC
- SERIAL CALL
- OPERATOR - INTRUSION
- OPERATOR QUEUE TIME
- LONG DURATION CALLS
- REPEATED INDIVIDUAL DISTRIBUTION (RID)
- ROUTE OPTIMIZATION

- [TRANSFER](#)

12.1 ACCOUNT CODE

Only the last entered valid account code will be recorded in a call record in the field `accountCode`.

12.1.1 ACCOUNT CODE - DNIS

If the optional DNIS (Dialed Number Information Service) feature is used, the DNIS number will be inserted as Account Code at the beginning of a call, both internally and in relevant private network scenarios.

12.1.1.1 ACCOUNT CODE - TRANSFER

For the transfer calls, the CIL record for the resulting transferred call will contain an account code if one was dialed either for the original call, or for the inquiry call. If an account code was dialed for both calls, then the account code for the call with which the transferring party was connected when the transfer feature was invoked will be recorded in the transferred CIL call record.

12.1.2 ACCOUNT CODE - OPERATOR EXTENDING

For an operator call extended (transferred) after answer, the extended call record will receive the account code entered by the operator, if any. If both calls contain account codes, the same rules as for Transfer are valid.

12.1.3 ACCOUNT CODE - CONFERENCE

If a party is included in a conference, any account code associated with the inquiry call will remain unique to that call record unless the leader enters a new account code for the conference in the speech state. The last entered valid account code in the conference speech state will be distributed to all call records and will overwrite the account code in the existing call records.

12.1.4 ACCOUNT CODE - CALLBACK

The account code entered will be recorded and shown on the CIL, QoS output record.

12.1.5 ACCOUNT CODE - LAST NUMBER REDIAL

The account code entered will be recorded and shown on the CIL, QoS output record.

12.2 ALTERNATIVE ROUTING

Alternative routing calls are identified with a condition code in the CIL, QoS output.

12.3 AUTHORIZATION CODE

If a call has been established using an authorization code then the entered code will be recorded in the cilCode field.

12.4 ADDITIONAL NUMBER

Traffic to or from an analog telephone will be recorded as a normal internal or external call. However, outgoing traffic from a secondary telephone will only record its primary number.

12.5 CALLBACK

A callback call will be treated as a normal internal or external call. The call back requesting party is recorded as the “calling party” and the other party is recorded as the “dialed party”.

12.6 CALL DEFLECTION, SINGLE-STEP TRANSFER

If the call has been deflected with maintained ACD or CTI queue to a public destination it is treated like an ECF call, that is, the charged number field will contain the deflecting party's number and the answering position will be recorded in the connected number field.

When a call has been single step transferred, the existing CIL or QoS record is output at the time the call is transferred and a new CIL or QoS record is created with the new connected party. If the original party is an outgoing PSTN call then the new connected party will be the A-party. The new CIL or QoS record will receive the condition code for transfer calls.

If a call has been deflected with maintained ACD or CTI queue and answered, and therefore re-enters the queue again, the existing CIL or QoS record is output and a new CIL or QoS record is created with the remaining parties data. The new CIL or QoS record will receive the condition code for transferred calls.

12.7 CALL METERING

The tax pulse and charging information sent from the public exchange will be recorded in the CIL or QoS when a call is made over an outgoing charging line.

12.7.1 CALL METERING - TRANSFER OR EXTEND

The tax pulses or charging information is added to the original record up to the moment that transfer takes place if the call is performed over an outgoing charging line. After the transfer, the tax pulses and call charge are recorded in the inquiry or in the transferred new record.

In an operator extended call to a public external line, the tax pulses or charging information is controlled by the MDP value. By changing the MDP value the operator tax pulses are moved to the assisted party.

If the metered line has been transferred or extended to a public external line, the transferring or extending party will be charged for the entire call. When the operator extend the metered line to a DPNSS or ISDN external tie line, the new party within the private network will be charged for the entire call.

12.7.2 CALL METERING - CONFERENCE

If the outgoing charging line is included in a conference connection, the tax pulses or charging information are recorded in the original record which introduced the charging line into the conference. After the conference leader drops out, the tax pulses or charging information will continue to be recorded against the original CIL or QoS record.

12.7.3 CALL METERING - CHARGED PARTY NUMBER CHANGE

When a change in the calling party number is received from private network or from a VPN and the own node has the public outgoing charging line connected, as in the case of a transfer occurring on another node over the network, the existing call record is output and a new call record with the updated calling party number is created.

12.8 CALL WAITING

The call waiting requesting party is recorded as the “calling party” and the other party is recorded in the dialed number field.

12.9 COLLECT CALL

The collect call initiating party is recorded as the “calling party”, the other party is recorded as “dialed party” and the charged party field will contain the dialed party’s number.

12.9.1 COLLECT CALL - TRANSFER OR EXTEND

The charging information for the collect call is added to the original record up to the moment that transfer takes place and the dialed party is charged for the call. After the transfer, the collect call charging information is recorded in the new record with the party to whom the call was transferred being the charged party.

If a call has been transferred onto a public external line, then the transferring or extending party will be charged for the entire call.

12.9.2 COLLECT CALL - CONFERENCE

If the outgoing charging line is included in a conference connection along with an incoming collect call, where the CIL or QoS is initiated, the tax pulses or charging information and the incoming collect call charging information is recorded in the corresponding records.

After the conference leader drops out, the tax pulses or charging information and the collect call charging information will continue to be recorded against the original CIL or QoS record.

12.9.3 COLLECT CALL - CALL PICKUP

If individual pick up or group call pick up is initiated on an collect call, then the charged party field consists of Picking Up party's number.

12.10 CONFERENCE

A new call record will be initiated for each party connected to a conference by means of the inquiry, or original calls. Each record is attached to each member which accounts for the call with the leader before conference and continuing through. There is no common record for the whole conference. A condition code identifies each call record as a conference call.

A record is recorded as each conference member leaves a conference.

The conference leader accounts for all CIL or QoS records in the conference that this person initiated even after having left the conference. The new leader is accounted for all subsequent conference records.

If more than two parties remain when the conference leader leaves a conference, no call record will be completed and output. If two parties remain, a new call record will be created with the information about the remaining parties. The original call records will be completed and output. If a conference leader who is not the original conference leader leaves a conference with more than two members, then the call record between the leaving conference leader and the original leader will be output.

12.10.1 CONFERENCE - CALL WAITING

A new call record is initiated when the conference leader answers the waiting call. The new call record is output after the waiting call is terminated.

12.10.2 CONFERENCE - INTRUSION

A new call record is initiated when the conference leader makes an intrusion. The new call record is output after the intrusion call is terminated.

12.11 COORDINATED DIALING PLAN CALLS

Coordinated Dialing Plan (CDP) calls are a subset of private network calls and are identified by the dialed number type. If the number of dialed digits is five or less and the dialed number contains access code digits, a call is classified as a CDP call. Only dialed numbers, which are displayed for outgoing calls, are affected. CIL or QoS must be set up to display the number as dialed, not as out-pulsed.

12.12 DIRECT CALL PICKUP

The originating party will be recorded in the calling number field. The dialed number will be recorded in the dialed number field.

12.13 DIVERSION

For all the diversion calls, the dialed number will be recorded in the dialed number field.

12.14 DTS

A DTS can have many different directory numbers (ODN, ADNs, and MDNs) that can be used to initiate or receive calls.

The following tables describe the calling party and the called party identification that are presented in a CIL or QoS output when the different directory numbers (in the same DTS) are used:

a	Calling party	CIL or QoS output calling party number
	ODN	directory of ODN
	ADN	directory of ODN on the DTS where the ADN key is situated
	MDN	directory of ODN on the DTS where the MDN key is situated
b	Answering party	CIL or QoS output called party number
	ODN	directory of ODN

ADN	directory of ADN
MDN	Main directory (ODN or ADN)

12.15 DYNAMIC ROUTE ALLOCATION

Obsolete function. Calls made between nodes in a network using a dynamic route allocation trunk were recorded with a condition code indicating this type of call, but the function is not supported from MX-ONE 6.0.

12.16 EMERGENCY EXTENSION CALL

A new call record is created for each new party introduced into an emergency conference (which means that the party has called a busy emergency extension). When a party leaves the emergency conference, the corresponding call record is completed and output. If the emergency extension generates a clearing signal, all remaining participants are released and the corresponding call records are completed.

12.17 EXTERNAL FOLLOW-ME, ECF

One CIL or QoS record is created for each ECF call. The calling number field contains the station number which invoked the ECF. The dialed number contains the external number to which the call has been forwarded.

However if General formats are used while initiating CIL or QoS, then the calling number field contains the station number which made the call to the ECF-extension. The dialed number field will contain the number of the ECF-extension. The charged number field will contain the number of the ECF-extension.

The CIL or QoS functions also have the ability to log abandoned ECF calls.

If the MX-ONE Service Node is acting as a gateway node for all calls (as a result of an ECF executed in the previous node) and going to a public destination the charged number will be the incoming A-number.



Note:

Mobile extension call via SIP trunk is logged with the same condition code as ECF.

12.17.1 ECF - TRANSFER

The following conventions are used to describe the parties involved in a transfer:

A-ext: Station which invoked the ECF feature. B-ext: Internal station. C-ext: Station which transfers the call to the ECF station.

In the examples dealing with ECF where transfer before answer occurs, A-ext is recorded in the calling number field for record number 2. Even though C-ext is the actual calling party, A-ext has invoked the ECF and therefore should be charged for the external call.

1. Incoming call transfer to external party before answer Incoming external line ---> C-ext
---> A-ext (ECF) ---> external party Two CIL or QoS records will be output.

Record 1:

Calling number

Incoming external line ID

Dialed number

C-ext

Condition code

Incoming call (= I)

Call duration

From the time C-ext answers to the termination of the transfer call or to the incoming external line disconnects in case of no answer from external party

Call identity

Call Id of record 1.

Record 2:

Calling number

A-ext which invoked the ECF

Dialed number

External party number

Condition code

ECF call (= X)

Call duration

From external party answers to transfer call terminated

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

2. Incoming call transfer to external party after answer Incoming external line ---> C-ext
---> A-ext (ECF) ---> external party Two CIL or QoS records will be output.

Record 1:

Calling number

Incoming external line ID

Dialed number

C-ext

Condition code

Incoming call (= I)

Call duration

From the time C-ext answers to the termination of the transfer call or to the incoming external line disconnects in case of no answer from external party

Call identity

Call Id of record 1.

Record 2:

Calling number

A-ext which invoked the ECF

Dialed number

External party number

Condition code

ECF call

Call duration

From external party answers to transfer call terminated

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

3. Station transfer to ECF station before answer B-ext ---> C-ext ---> A-ext (ECF) >

external party Two CIL or QoS records will be output.

Record 1:

Calling number

B-ext

Dialed number

C-ext

Condition code

Incoming call

Call duration

From C-ext answering to the time of transfer

Call identity

Call Id of record 1.

Record 2:

Calling number

A-ext which invoked the ECF

Dialed number

External party number

Condition code

ECF call

Call duration

From external party answers to termination of transfer call

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

4. Station transfer to ECF station after answer B-ext ---> C-ext ---> A-ext (ECF) >
external party Two CIL or QoS records will be output.

Record 1:

Calling number

B-ext

Dialed number

C-ext

Condition code

Incoming call

Call duration

From C-ext answering to the time of transfer

Call identity

Call Id of record 1.

Record 2:

Calling number

A-ext which invoked the ECF

Dialed number

External party number

Condition code

ECF call

Call duration

From external party answers to call transferring

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

Record 3:

Calling number

A-ext which invoked the ECF

Dialed number

External party number

Condition code

ECF call

Call duration

From call transferring to transfer call complete

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

12.18 FOLLOW-ME

For follow-me calls, the dialed number will be recorded in the dialed number field. For formats that have an outgoing external line ID field, the answering party will be recorded in that field. For the FP15 format the answering party will not be recorded, since FP15 does not contain an outgoing external line ID field. The condition code will be set according to the traffic case.

12.19 GROUP CALL PICKUP

With respect to CIL or QoS the information provided for Group Call Pickup is as follows: Party A calls party B. The call is picked up by party C.

CIL or QoS records party A as the “calling number”, party B as the “dialed number”.

For formats that have an outgoing external line ID field, party C, the answering party, will be recorded in that field. For the FP15 format the answering party will not be recorded, since FP15 does not contain an outgoing external line ID field. The condition code assigned will be according to the traffic case.

12.20 GROUP HUNTING OR CASCADE RING GROUP

Party A calls the Group Hunting/Ring group directory number. Extension B is selected and a call is established between A and B.

CIL or QoS records party A as the “calling number”, the group hunting/ring group directory number as the “dialed number”. For formats that have an outgoing external line ID field, the answering party will be recorded in that field. For the FP15 format the answering party will not be recorded, since FP15 does not contain an outgoing external line ID field. The condition code assigned will be according to the traffic case.

12.21 INTERNAL TRANSFER FROM VOICE TO DATA CALL

When the transfer of voice call to data call takes place the CIL or QoS will generate a data call record with condition code “V” or “VJ” in addition to the existing voice call record.

12.22 INTRUSION

A request for intrusion will result in the initiation of a new call record in addition to the established record. The condition code “R” used for intrusion will also be used for intrusion call.

The following conventions are used to describe the parties involved in an intrusion:

A: Intrusion initiating party **B:** Whom intrusion was initiated upon **C:** The third party connected to the “B” party

Intrusion is terminated when:

1. A-party hangs up or clears
 - a. “B”- “C” connection remains
 - b. “A”- “B” connection is completed and output
2. B-party hangs up or clears
 - a. “A”- “B” and “B”-“C” connections are completed and output
 - b. A new record is seized when B answers the recall from party A
3. C-party hangs up or clears
 - a. “B”- “C” connection is completed and output
 - b. “A”- “B” connection remains

If the A-party leaves the intrusion call by alternating to a parked party, the processing is the same as if the A-party hangs up or disconnects from the intrusion call.

If the A-party is an operator who requests disconnection of the third party, the call record is completed both for the "A"- "B" and "B"- "C" connection, and a new record will be created for "A"- "B".

12.23 ISDN CALL BY CALL

All calls made through an ISDN Call by Call service will be recorded by CIL or QoS. Both outgoing and incoming calls made using this service will be recorded in the second access code field with ASCII character "S" followed by the service number used. For example, an outgoing call made using the ISDN CBC service number 14 would record a data of "S14" in the second access code field. It should be noted that if a call was made through LCR that the real second access code will be overwritten by the CBC information.

12.24 LEAST COST ROUTING

The Least Cost Routing (LCR) access code and the fictitious destination code are recorded in the route access code "1" and "2" fields respectively. Either the dialed number or the out-pulsed number will be recorded, which depends on an MDP value per exchange.

12.25 MALICIOUS CALL TRACE

The determination of whether to record Malicious Call Trace (MCT) calls in CIL is controlled by the ASPAC command (parameter 135).

Default setting is that MCT is not recorded (if parameter 135 is not changed), except for Application System Spain.

12.26 MOBILITY EVENTS

For each mobility event logged, a new call logging record is created. Data in the record for an ongoing call will not be changed due to mobility events. Mobility Events can only be recorded in the General format.

For the possible value range, see [Mobility Events](#)

12.26.1 LOCATION REGISTRATION

Successful and unsuccessful attempts of Location Registration will be logged and displayed in the EVENT field.

12.26.2 DETACH

Successful and unsuccessful attempts of Detach will be logged and displayed in the EVENT field.

12.26.3 HANDOVER

All types of Handover will be logged and displayed in the EVENT field.

12.26.4 CALL TO OR FROM CXN

Calls to or from CXNs are defined as mobility events because they might provide additional information about the users movement pattern.

12.26.5 ABNORMAL CALL RELEASE

Abnormal Call Releases will be displayed in the EVENT field.

12.27 NETWORK CALLS

Calls made between nodes in a network are not specifically distinguished as such, for example, no separate condition code exists to indicate these call types. They are handled as outgoing or incoming calls relative to that node. Call Logging should be active in each node in order to coordinate results from each node to obtain a more complete record of inter-node call cases. One difference between network calls and other incoming calls is that if a call originator's number is signaled through the network, then this number will be recorded as the calling number.

A call made to the centralized operator in another node is also recorded as an outgoing call. The dialed number field will contain the common operator number.

A call from another node to the common operator will be recorded as an incoming call answered by a party other than the dialed party. The calling number from the other node will be displayed if this number is signaled through the network. The dialed number field will record the PBX operator access number. The answering operator number will be shown in the external line ID field. The condition code "NI" will be recorded.

12.28 NETWORK DIVERSION

A call diverted to another node, whether on busy, no answer, or by direct diversion, will be recorded with a condition code indicating an outgoing call. The calling number field will show the calling party and the dialed number field will contain the dialed number. The external line ID will be shown in the external line ID field.

A diverted call from another node will be shown as an incoming call answered by a party other than the dialed party. The calling number and dialed numbers will be those of the other node if signaled through the network. The answering party number will be shown in the external line ID field.

12.29 NETWORK REROUTING

An incoming call rerouted to an operator in another node will result in an outgoing call record. The calling number field will contain the originating party external line ID or the originating party number if signaled through the network. The dialed number field will contain the dialed number. The outgoing external line ID will be shown in the external line ID field.

A call rerouted from another node to the operator will be recorded as an incoming call answered by a party other than the dialed party. The calling number and dialed numbers will be those of the other node if signaled through the network. The answering party number will be shown in the external line ID field.

12.30 NETWORK TRANSFER

When a call is transferred in a network and a new connected party number is available, the existing CIL or QoS record is output at the time the call is transferred and a new CIL or QoS record is created with the new connected party. If the original party is an outgoing PSTN call then the new connected party will be the A-party. The new CIL or QoS record will use a condition code to indicate the transfer, and the outgoing external line ID field will contain the external line ID.

12.31 OPERATOR

Traffic to or from an operator will be recorded as a normal internal or external call.

12.31.1 OPERATOR - EXTENDING

The data in the CIL or QoS call records for the operator extending function depends on whether the party to which the call was extended has answered or not before the extending is executed.

Regardless if the call is extended before or after an answer, the call connected to the left side of the console is recorded as a separate CIL or QoS call record. With respect to the operator this call is either a received or an initiated call.

The operator directory number recorded by CIL or QoS is always the individual number except for the case of a call to a common operator (or group of operators). When dialing the common number, this number is recorded for the initial call as the dialed number. However, any recall associated to this call will record the corresponding individual number.

12.31.1.1 Extending Before Answer

CIL or QoS outputs three records. Record 1 at the time when the extended call is answered and record 2 when the operator makes a call and presses extending button and hangs-up before answer, and record 3 when the extended call is terminated.

Record 1:

Calling number

Originator of the original call (A-party).

Dialed number

Dialed number (B-party) or terminator of the original call (Operator).

Connected number

Terminator of the call connected, that is, the individual operator number.

Condition code

According to the traffic case

Ring Time

Time starting from the B-party is alerted until the alert stops (B-party answered).

Call duration

The time from when the call (B-party) is answered until the answered (operator) party (B-party) extends the call to the final terminating party (C-party).

Call identity

Call Id of record 1.

Record 2

Calling number

Answered party in record 1 (B-party) or connected number or originator of the inquiry call (Operator).

Dialed number

Dialed number (C-party) or terminator of the inquiry call.

Connected number

Terminator of the call connected. That is, the C-party may have diversion.

Condition code

According to the traffic case.

Ring time

Time starting from the C-party is alerted until the time B-party extends the call.

Call duration

Zero.

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

**Note:**

This record for the inquiry call is output only when the option of logging abandoned calls is invoked.

Record 3**Calling number**

Originator of the original call (A-party) or calling number in record 1.

Dialed number

Terminating party or the inquiry call (C-party) or dialed number in record 2.

Connected number

Terminator of the call connected. That is, the C-party may have diversion or same connected number in record 2.

Condition code

According to the traffic case along with A-party.

Ring time

Time starting from the C-party is alerted after extending by B-party until the time the alerting stopped (C-party answered).

Call duration

Time from the C-party is answered after extend by B-party until the call termination.

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

12.31.1.2 Extending After Answer

CIL or QoS outputs three records. Record 1 and 2 at the extending and record 3 when the extended call is terminated

Record 1

Calling number

Originator of the call (A-party).

Dialed number

Dialed number (B-party) or terminator of the original call (Operator).

Connected number

Terminator of the call connected, that is, the individual Operator number.

Condition code

According to the traffic case.

Ring time

Time starting from the B-party is alerted until the time alerting stopped (B-party answered).

Call duration

The time from when the call (B-party) is answered until the answered party (B-party) extends the call to final terminating party (C-party).

Call identity

Call Id of record 1.

Record 2

Calling number

Answered party (Operator) in the record 1 or originator of the inquiry call.

Dialed number

Dialed number or terminator of the inquiry call.

Connected number

Terminator of the call connected, that is, the C-party may have diversion.

Condition code

According to the condition code.

Ring time

Time starting from the C-party is alerted until the time alerting stopped (C party answered).

Call duration

Time from the inquiry call is answered until the Operator, that is, the originator of the inquiry call (B-party) extends the call to final terminating party (C-party).

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

Record 3

Calling number

Originator of the original call (A party) or calling number in record 1.

Dialed number

Terminating party or the inquiry call (C party) or dialed number in record 2.

Connected number

Terminator of the call connected, that is, the C-party may have diversion or same connected number in record 2.

Condition code

According to the traffic case along with A party.

Ring time

Zero.

Call duration

The time from the final terminating party (C-party) is answered after extended by the originator (operator) of the inquiry (B) party until the call terminated.

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

12.32 RECALL AS A RESULT OF EXTENDING

When the operator is recalled due to a non-answered extended call (extending before answer), the following cases exist:

1. The operator answers the recall before the called party. When the recall is answered, the connections and the associated CIL or QoS information are the same as before the call was extended. The CIL or QoS information recorded depends on what action the operator takes after answering the recall, for

example, a call on the right side is cleared or re-extended. No CIL or QoS data is output when the recall is initiated or answered.

Operator clears the right side:

CIL or QoS outputs one record when clearing the right side.

Calling number

Originator of the call connected to the left side.

Dialed number

Operator

Condition code

According to the traffic case.

Call duration

The time from Operator answer of the initial call until the call on the right side is cleared.

Call identity

Call Id of this record.

Operator re-extends the call

CIL or QoS outputs one record at the re-extending. After the re-extending, the CIL or QoS information specified in the “extending before answer” case above applies.

Calling number

Originator of the call connected to the left side.

Dialed number

Operator

Condition code

According to the traffic case.

Call duration

The time from Operator answer of the recall until the re-extending.

Call identity

Call Id of this record.

2. The called party answers before the recall is answered by the Operator.

The CIL or QoS information specified in the “extending before answer” case above applies.

12.33 EXTENDING OF EXTERNAL LINE TO EXTERNAL LINE TRAFFIC

Incoming external line extended to outgoing external line

Record 1

Calling number

Incoming external line ID

Dialed number

Operator

Condition code

Incoming call

Call duration

From Operator answer to termination of the extended call or to the time the call is extended

Call identity

Call Id of record 1.

Record 2

Calling number

Incoming external line ID

Dialed number

External number dialed by the Operator

Condition code

Attendant (operator) extended call

Call duration

From external party answer to termination of the extended call or to the time the call is extended

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

For an incoming external line extended to an outgoing external line after answer, CIL or QoS outputs an extra record.

Calling number

Operator

Dialed number

External number dialed by the Operator

Condition code

According to the traffic case.

Call duration

From external party answer until the time the call is extended

Call identity

Call Id of this record.

If an incoming message oriented signaling tie-line, which is connected to an extension within the private network, is extended to an outgoing external line to the PSTN, record 1 and record 2 are output after the call is extended. The third CIL record is created to reflect the new calling party.

Record 3**Calling number**

Extension number in the private network

Dialed number

External number dialed by the operator

Condition code

Attendant (operator) extended call

Call duration

From start of the extended call to the termination of the extended call

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.



Note:

If the call is a Collect Call then there will be 2 records output.

Outgoing external line extended to outgoing external line

When extending an outgoing external line call to an outgoing external line, CIL or QoS outputs two records

Record 1

Calling number

Operator

Dialed number

External number dialed by the operator from the left side

Condition code

Outgoing call

Call duration

From answer to termination of the extended call or to the time the call is extended

Call identity

Call Id of record 1.

Record 2**Calling number**

Operator

Dialed number

External number dialed by the operator

Condition code

Attendant (operator) extended call

Call duration

From external party answer to termination of the extended call or to the time the call is extended

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

If an outgoing message oriented signaling tie-line, which is connected to an extension within the private network, is extended to an outgoing external line to the PSTN, record 1 and record 2 are output after the call is extended. The third CIL record is created to reflect the new calling party. The same applies to extending an outgoing PSTN external line to an outgoing MOS tie line.

Record 3**Calling number**

Extension number in the private network

Dialed number

External number dialed by the operator.

Condition code

Attendant (operator) extended call

Call duration

From start of the extended call to the termination of the extended call

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

12.34 SERIAL CALL

One separate CIL or QoS record is output for each of the calls processed in a serial call. The information recorded is identical to the call cases specified under the section “Extending”.

If the operator answers a recall of a serial call before the called party answers, the information recorded is the same as that of the call cases specified under the section “Recall as a result of extending”.

12.35 OPERATOR - INTRUSION

The following conventions are used to describe the parties involved in an intrusion

B: Whom intrusion was initiated upon

C: The third party connected to the B party

A CIL or QoS record is output for intrusion if any of the three parties involved in the intrusion goes on hook.

The operator disconnects from the intrusion

CIL or QoS outputs one record

Calling number

Operator

Dialed number

Party B (= intruded party)

Condition code

Intrusion

Call duration

The time from when the intrusion takes place until the operator disconnects.

Call identity

Call Id of this record.

C goes on-hook

CIL or QoS outputs two records: Record 1 when C goes on hook and record 2 when the intrusion call is terminated.

Record 1:

Calling number

According to the original call

Dialed number

According to the original call

Condition code

According to the original call

Call identity

Call Id of record 1.

Record 2:

Calling number

Operator

Dialed number

Party B (= intruded party)

Condition code

Intrusion

Call duration

Duration of the intrusion

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

12.36 OPERATOR QUEUE TIME

An incoming external line call that has been placed in the operator queue will have some number of seconds before the operator answers the call. That number of seconds is called the Operator Queue Time and will be recorded in the dialed number field for MDFP15 or ASB 501 04 standard formats. There is no risk of overwriting the dialed number because the dialed number is only up to five digits and is right adjusted. The Operator Queue Time replaces the left most two digits of the dialed number field and can have a value of up to 99 seconds. Any queue time longer than 99 seconds will be recorded as 99 seconds.

12.37 LONG DURATION CALLS

Calls with duration greater than the maximum duration limit are treated as long duration calls. Here in such cases, a new record is seized and the existing one is output. The duration of the call is computed and if it exceeds 9hrs 59 minutes and 59secs the duration is set to 9:59:59.

12.38 REPEATED INDIVIDUAL DISTRIBUTION (RID)

One CIL or QoS record is created for each call towards Personal Number with RID service active. The information stored during the basic call to the PN is not changed until the call is answered or distributed towards an operator or a group.

During the basic call, the information stored in the CIL or QoS record is as follows:

Calling number

A-ext

Dialed number

Personal Number

Condition code

Incoming call, Outgoing call, or Internal call

Call duration

-

Call identity

Call Id of this record.

If the call is not answered, the information related to this call is only stored in the PBX where the Personal Number is located.

If one of the positions in the active list answers the call, the call information logging is updated, so the new answering party is considered:

- Answering party is internal

Calling number

A-ext

Dialed number

Personal Number

Called Number

Deflected-to party

Condition code

Incoming call, Answering party different from dialed or Internal call, Answering party different from dialed

Call duration

The time from when the answering party answers to when the call is cleared.

Call identity

Call Id of this record.

- Answering party is in the private network

Calling number

A-ext

Dialed number

Personal Number

Outgoing trunk id

Trunk identity of the Deflected-to party

Condition code

Outgoing call, Answering party different from dialed or New answering party

Call duration

The time from when the answering party answers to when the call is cleared.

Call identity

Call Id of this record.

- Answering party is in the public network

Calling number

Personal Number

Dialed number

Answering party in the public network.

Outgoing trunk id

Trunk identity of the Deflected-to party

Condition code

ECF call

Call duration

The time from when the answering party answers to when the call is cleared.

Call identity

Call Id of this record.

However, if the General formats are used then the calling number field will contain the number of the party which called the personal number extension and the dialed number field will contain the personal number.

12.39 ROUTE OPTIMIZATION

Route optimization affects the call records for the originating and terminating PBXs. When route optimization takes place CIL or QoS will seize a new call record for the call to continue over the optimized route. When this happens the original call records will receive the condition code for “call terminated due to route optimization”. The new call records will receive the condition code for “call established due to route optimization”. In the new call records, the outgoing external line ID field will contain the optimized external line ID. Also, the A-party and B-party will be the same as in the original records, but may be swapped if the direction is changed due to the order of call initiation in a network transfer. The original records are released and output while the new records continue the call. Route optimization occurs only for voice calls.

12.40 TRANSFER

The CIL or QoS call records output data for a transfer call case depends on whether the party to which the call was transferred has answered or not before transferring is executed. In addition, there is a special external line to external line transfer case.

Transfer Before Answer CIL or QoS outputs three records. Record 1 is output at the time when the call is answered and record 2 is output when the transferring party hangs up before answer, and record 3 output when the transferred call is terminated.

In this following output it is assumed that the originator of the original call made the inquiry call.

Record 1

Calling number

Originator of the original call (A-party).

Dialed number

Dialed number (B-party) or terminator of the original call.

Connected number

Terminator of the call connected, that is, answered party.

Condition code

According to the traffic case.

Ring time

Time starting from the Dialed (B) party is alerted until the alerting stopped (Dialed party answered).

Call duration

Time from the Dialed party (B) party answered until the answered party (B party) transfers the call.

Call identity

Call Id of record 1.

Record 2

Calling number

Answered (B) party in Record 1 or connected party or originator of the inquiry call.

Dialed number

Dialed number (C-party) or terminator of the inquiry call.

Connected number

Terminator of the call connected, that is, the Dialed (C) party may have diversion.

Condition code

According to the traffic case.

Ring time

Time starting from the Dialed (C) party is alerted until the Originator of the Inquiry call (B) party transfers the call before answer.

Call Duration

Zero.

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.



Note:

This record for the inquiry call is output only if the option of logging abandoned calls is invoked.

Record 3

Calling number

Originator of the original call (A-party).

Dialed number

Terminating of the Inquiry call (C-party) or dialed number in Record 2.

Connected number

Terminator of the call connected, that is, the Dialed (C) party may have diversion or the same connected number in the Record 2.

Condition code

According to the traffic case with transfer call.

Ring time

Time starting from the C-party is alerted after extending by B party until the alert stops.

Call duration

Time from when the C-party is answered after extend by B party until the termination.

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

Transfer After Answer CIL or QoS outputs three records. Records 1 and 2 are output at the time when the call is transferred, and record 3 is output when the transferred call is terminated.

Record 1

Calling number

Originator of the original call (A-party).

Dialed number

Dialed number (B-party) or terminator of the original call.

Connected number

Terminator of the call connected, that is, answered party.

Condition code

According to the traffic case.

Ring time

Time starting from the B-party is alerted until the time the alerting stopped (B-party answered).

Call duration

Time from, when the call (B-party) is answered until the answered party (B-party) transfers the call to final terminating party (C-party).

Call identity

Call Id of record 1.

Record 2

Calling number

Answered party in record 1 (B-party), or connected number, or originator of the inquiry call.

Dialed number

Dialed number (C-party) or terminator of the inquiry call.

Connected number

Terminator of the call connected, that is, C-party may have diversion.

Condition code

According to the traffic case.

Ring time

Time starting from the C-party is alerted until the time C-party answers the inquiry call.

Call duration

The time from the inquiry call is answered until the originator of the inquiry call (B party) transfers the call to final terminating party (C-party).

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

Record 3

Calling number

Originator of the original call (A party) or calling number in record 1.

Dialed number

Terminating party or the inquiry call (C party) or dialed number in record 2.

Connected number

Terminating party or the inquiry call (C party) or dialed number in record 2.

Condition code

According to the traffic case with transfer.

Ring time

Zero

Call duration

The time from the final terminating party (C-party) is answered after transferred by the originator of the inquiry party (B-party)

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

External line to external line CIL or QoS outputs two records. Both records will be output together at the time when the transferred call is terminated.

Incoming external line transferred to outgoing external line. Record 1

Calling number

Incoming external line ID

Dialed number

Dialed number of the terminator of the original call

Condition code

Incoming call

Call duration

From answer to termination of the transferred call or to starting of the transferred metered call.

Call identity

Call Id of record 1.

Record 2

Calling number

Incoming external line ID

Dialed number

External number dialed by the originator of the inquiry call

Condition code

Transfer

Call duration

From external party answer to termination of the transferred call or to the starting of the transferred metered call

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

For the case of an incoming external line transferred to an outgoing external line after answer, CIL or QoS outputs an extra record.

Calling number

Originator of the inquiry call

Dialed number

External number dialed by the originator of the inquiry call

Condition code

According to the traffic case

Call duration

From external party answer to until the call is transferred.

If an incoming Message-Oriented Signaling (MOS) tie line, which is connected to an extension within the private network, is transferred to an outgoing external line to the PSTN, record 1 and record 2 are output after the transfer. The third CIL record is created to reflect the new calling party.

Record 3

Calling number

Extension number in the private network

Dialed number

External number dialed by the originator of the inquiry call

Condition code

Transfer

Call duration

From the start of the transferred metered call to the termination of the transferred metered call

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

Outgoing external line transferred to outgoing external line

Record 1

Calling number

Originator of the original call

Dialed number

External number dialed by the originator of the original call

Condition code

Outgoing call

Call duration

From answer to termination of the transferred call or to the start of the transferred metered call.

Call identity

Call Id of record 1.

Record 2

Calling number

Originator of the original call

Dialed number

External number dialed by the originator of the inquiry call

Condition code

Transfer

Call duration

From external party answer to termination of the transferred call or to the start of the transferred metered call.

Call identity

Call Id of record 2.

Call identity Associated 1

Call Id of record 1.

If an outgoing call to the PSTN is transferred to a message oriented signaling (MOS) tie line, which is connected to an extension within the private network, record 1 and record 2 are output after the transfer. The third CIL record is created to reflect the new calling party. The same applies to transferring an outgoing MOS tie line to an outgoing PSTN external line.

Record 3

Calling number

Extension number in the private network

Dialed number

External number dialed by the originator of the inquiry call

Condition code

Outgoing call

Call duration

From the start to the termination of the transferred metered call

Call identity

Call Id of record 3.

Call identity Associated 1

Call Id of record 1.

Call identity Associated 2

Call Id of record 2.

Single Step Transfer The CIL or QoS call records output for a single step transfer call case are the same as that of a transfer before answer.

This chapter contains the following sections:

- [SETUP OF CALL INFORMATION OUTPUT IS INVALID](#)
- [FAILED WHILE WRITING TO CALL INFORMATION OUTPUT](#)
- [FAILED WHILE CONNECTING CALL INFORMATION OUTPUT](#)
- [CALL INFORMATION OUTPUT QUEUE ALMOST FULL](#)
- [SPEECH QUALITY VALUE AT YELLOW LEVEL](#)
- [SPEECH QUALITY VALUE AT RED LEVEL](#)

Call logging will send alarms if the configuration is deemed faulty, the connection to a device fails, or if buffers are filling up.

Quality of Service will send alarms if the calculation of the speech quality and the subsequent supervision of the r-value hits the supervision values.

The CIL or QoS feature generates alarms to indicate the following:

- Faulty configuration
- An error has occurred during output of call records
- An output device is disconnected or not available.
- Output queues are almost full. (indicating low bandwidth on output stream)
- Speech quality is poor (warning)
- Speech quality is failing (bad)

13.1 [SETUP OF CALL INFORMATION OUTPUT IS INVALID](#)

This fault is generated if the configuration is faulty. A new initiation of the configuration has to be done.

13.2 [FAILED WHILE WRITING TO CALL INFORMATION OUTPUT](#)

This error indicates that writing on the device or database is not possible. The device can be off-line or reconfigured so that the permissions are changed.

13.3 [FAILED WHILE CONNECTING CALL INFORMATION OUTPUT](#)

The server or device can not be found, the network might be down, or the device does not exist.

13.4 CALL INFORMATION OUTPUT QUEUE ALMOST FULL

This alarm will alert the personnel that the bandwidth of the output device or queue is too low. The data is collected faster than the device can write the data out of this output. Reconfigure the output, to increase speed, or split up the data to more than one output.

13.5 SPEECH QUALITY VALUE AT YELLOW LEVEL

An alarm is generated when the monitor level of the QoS R-values is below the threshold. This alarm indicates minor problems in the network used for IP telephony.

13.6 SPEECH QUALITY VALUE AT RED LEVEL

An alarm is generated when the monitor level of the QoS R-values is below the threshold. This alarm indicates severe problems in the network used for IP telephony.

This chapter contains the following sections:

- [CALLINFO_STATUS_SET](#)
- [CALLINFO_OUTPUT_SET](#)
- [CALLINFO_STATUS_PRINT](#)
- [CALLINFO_MASK_SET](#)
- [CALLINFO_MASK_PRINT](#)
- [CALLINFO_OUTPUT_CHANGE](#)
- [CALLINFO_FORMAT_PRINT](#)
- [CALLINFO_OUTPUT_INFO](#)
- [CALLINFO_CONDCODE_SET](#)
- [CALLINFO_CONDCODE_PRINT](#)
- [CALLINFO_SQL_TO_FILE](#)
- [CALLINFO_FILE_TO_FILE](#)
- [CALLINFO_FILE_TO_SQL](#)
- [CALLINFO_TCP_PRINT](#)
- [CALLINFO_LIMIT_SET](#)
- [CALLINFO_LIMIT_PRINT](#)

The exchange allows the administration personnel to initiate the CIL or QoS function and to specify a call record format, output criteria, and output frequency into the exchange by using commands from the maintenance or I/O terminal. For a detailed description of the commands, see the command description for *CALL INFORMATION LOGGING*.

14.1 CALLINFO_STATUS_SET

This command allows the administration personnel to enable or disable the CIL or QoS outputs. Before enabling an output, a type and subtype with an optional format criteria must be set. The command can also be used to enable forwarding of CIL or QoS data for up to three LIMs for central output to a central collecting point.

14.2 CALLINFO_OUTPUT_SET

This command allows the administration personnel to specify the outputs type and subtype, formatting of the output data, output criteria to store data on this output, file name, target server name and socket, database name, username, and password on the target server, V.24 characteristics, type of end of line, and type of logging.

14.3 CALLINFO_STATUS_PRINT

This command allows the administration personnel to print the current CIL or QoS data. If an output is specified, the information pertaining to the specified output file will be printed. When no output file is specified, all the information pertaining to all the initiated output files will be printed.

14.4 CALLINFO_MASK_SET

This command sets the number of discrimination digits per number-length. 20 different values can be set. When at least the number-length stated is reached the number of digits set in the command is removed or substituted with an predefined character.

14.5 CALLINFO_MASK_PRINT

Prints the current setting of what number-lengths are used to remove digits, and how many, and what character is used to substitute the removed digits.

14.6 CALLINFO_OUTPUT_CHANGE

This command allows the General output formats to be changed. The predefined output formats cannot be changed, but the output filtering can be altered.

The command can append to an existing definition but there is no facility to edit only a part of a general output format, that is, in that case the entire format definition is initiated from the beginning.

14.7 CALLINFO_FORMAT_PRINT

This command is used to print the preconfigured format strings of a subtype. It can be used to create a dedicated format.

14.8 CALLINFO_OUTPUT_INFO

This command prints information about the type and subtypes command formats. This includes syntax information of the format script language.

14.9 CALLINFO_CONDCODE_SET

This command allows the condition code character to be changed. It is also possible to suppress a certain call record to be output by removing the condition code character.

14.10 CALLINFO_CONDCODE_PRINT

This command allows the condition code characters to be displayed.

14.11 CALLINFO_SQL_TO_FILE

Additional command that will copy data from an sql database to a file with a specific subtype and format specified in the command. This command is useful to extract data especially when setting up and testing format scripts.

14.12 CALLINFO_FILE_TO_FILE

Additional command that will copy data from a file to another file with subtype and format specified in the command. The input file format in this case must be one of commaSeparated or xml as they always contain all data verbatim.

14.13 CALLINFO_FILE_TO_SQL

Additional command that will copy data from a file to an sql database. The input file format in this case must be one of commaSeparated or xml as they always contain all data verbatim.

14.14 CALLINFO_TCP_PRINT

Additional program, namely, a TCP/IP daemon that will listen to a specific port and display the received data on the screen. Primarily intended as an installation and fault-finding tool.

14.15 CALLINFO_LIMIT_SET

This command will set the limits used to generate the QoS alarms.

The command will set limits for the calculated R-value for bad and warning level calls.

The command will also set how many samples the running list of samples will collect and how many bad samples that will raise the red alarm, and how many warning level to raise the yellow alarm.

14.16 CALLINFO_LIMIT_PRINT

This command will print the data assigned to the R-value supervision, the current state of the alarms, and the current running buffer status.

The following parameters are classified as market dependent parameters (MDP) of which the values can vary among the different application systems.

- The maximum duration time for a call record range from 1 to 10 hours.

- The maximum number of outputs per LIM is 10.
- The maximum number of active outputs per LIM is 10.
- The maximum number of data forwarding positions per LIM is three.
- The maximum number of simultaneous calls to be recorded per LIM is 500.
- It is not immediately obvious when using the FP15 output format as to whether an account code or an authorization code has been entered, due to the fact that both are recorded in the same field.
- Considerations have to be taken if the SQL database should reside in the same computer as the telephone application or on a separate (external) computer. In a situation with less than seven calls per second and fewer than 70,000 call records in the database and a low query rate, the database can coexist on the same computer. If there are more than 20 calls per second and more than 200,000 call records in the database and a high query rate, an external database server is needed.

This chapter contains the following sections:

- [LIM ISOLATION](#)
- [DATA OR PROGRAM RELOAD](#)
- [RESTART](#)
- [OUTPUT DEVICE FAILURE](#)

This section describes how the CIL or QoS feature is affected by different exchange failures.

17.1 LIM ISOLATION

DATA ADMINISTRATION

The CIL or QoS data can be changed even if isolated LIMs exist in the exchange. However, the CIL or QoS data changes will be lost at the data reload, after the system has recovered all LIMs. If a LIM is isolated or faulty during the execution of any command this LIM will be skipped and the command will continue with the next LIM but the system will make a printout that the LIM in question failed.

17.1.1 CIL OR QOS OUTPUT

Only local CIL or QoS output data will be output from a LIM that is isolated. Data that is forwarded to another LIM will be lost.

17.2 DATA OR PROGRAM RELOAD

DATA ADMINISTRATION

A reload (of data or program) does not affect the CIL or QoS data if a data dump is performed after every data modification.

Print commands can be used during a reload or data dump but other commands can not be used.

Program reload causes a loss of calls in progress, and of call records waiting to be output.

17.3 RESTART

CIL AND QOS OUTPUT

Any system restart causes a loss of calls in progress, and of call records waiting to be output.

17.4 OUTPUT DEVICE FAILURE

DATA ADMINISTRATION

The CIL and QoS output status will be changed to reflect the status of the output stream.

17.4.1 CIL AND QOS OUTPUT

The CIL and QoS output data will be interrupted if the output unit device fails.

17.4.2 BACK UP CIL AND QOS OUTPUT

The CIL and QoS outputs can be configured to have many simultaneous outputs at the same time. They can each have different output types, subtypes and formats. Thus, it is recommended to initiate one output as safety backup with type = file, subtype = comma-separated or xml for storing all data to local files. Then the rest of the outputs can be assigned to do the main outputs in any type and subtype format as desired.

With this method of overlapping recording it is possible to have a local safety copy in case the main device is off-line. The safety backup can then be used to create the missing data to the main device, with the help of the tree conversions commands that are accompanying this feature.

