

SIEMENS

Information

**Description of Output Data
P30308-X6800-K 12-02-7617**

**IACAMA
Inter Administration Charging by means of AMA**

EWSD V12

The document comprises 101 pages, all pages have issue no 02.

0 General Information	5
0.1 Issue Control.....	5
0.2 History	5
0.3 References	5
0.4 Glossary and Abbreviations	6
0.5 Keyword / Descriptor.....	6
0.6 List of Figures and Tables	7
1 Introduction	9
2 File Format Description	15
2.1 File format on disk.....	15
2.2 File format on tape or MOD.....	15
3 Remarks on File Contents	17
3.1 IACAMA Records	17
3.2 IACAMA Record Generation	18
3.2.1 Inter administration charging.....	18
3.2.2 Intermediate Charging	19
3.2.3 IACAMA for not answered calls.....	20
4 Record Format Description	21
4.1 IACAMA Record	21
4.2 System and Filler Records.....	22
4.3 General Record Coding Details	23
4.3.1 Coding of 'binary' fields	23
4.3.2 Coding of "packed digit" fields.....	23
4.3.3 Coding of "character" fields.....	24
5 Field Description	25
5.1 AMA Inter administration charging call Record	25
5.1.1 Fixed part (Header).....	25
5.1.2 Part 2 : Data Packages	30
5.1.2.1 Package 100 : DATE / TIME / DURATION.....	30
5.1.2.2 Package 101 : PARTNER DIRECTORY NUMBER	32
5.1.2.3 Package 102 : SERVICE INFO.....	34
5.1.2.4 Package 103 : CHARGE UNITS FOR CONNECTION.....	38
5.1.2.5 Package 105 : TRUNK IDENTIFICATION INCOMING	39
5.1.2.6 Package 106 : TRUNK IDENTIFICATION OUTGOING.....	40
5.1.2.7 Package 107: TRUNK IDENTIFICATION INCOMING (CIC FORMAT).....	41
5.1.2.8 Package 108: TRUNK IDENTIFICATION OUTGOING (CIC FORMAT).....	42
5.1.2.9 Package 110 : CONNECTION IDENTIFICATION NUMBER	43
5.1.2.10 Package 117 : PROJECT SPECIFIC DATA	44
5.1.2.11 Package 119 : TRANSMISSION MEDIUM REQUIRED.....	45
5.1.2.12 Package 120 : CATEGORY.....	46
5.1.2.13 Package 128 : NEW DESTINATION	47

5.1.2.14 Package 130 : TRAFFIC QUALITY DATA	48
5.1.2.15 Package 134 : DURATIONS BEFORE ANSWER.....	51
5.1.2.16 Package 138 : B METER PULSES	52
5.1.2.17 Package 142 : CALLING PARTY NUMBER.....	53
5.1.2.18 Package 145 : CAC NUMBER	56
5.1.2.19 Package 146 : THIRD PARTY NUMBER.....	59
5.1.2.20 Package 147 : TARIFF INFO.....	63
5.1.2.21 Package 148 : CAC NEW DESTINATION	72
5.1.2.22 Package 149 : GLOBAL CHARGE REFERENCE	73
5.1.2.23 Package 150 : AUDIT TRAIL	75
5.1.2.24 Package 154 : ADDITIONAL CALL INDICATORS	77
5.1.2.25 Package 157 : OUTGOING PA-SLAVE IDENTIFICATION	80
5.2 Date Time Change Record	82
5.3 Overflow Record	84
5.4 Breakdown Record	85
5.5 Filler Records	86
 6 Operation Instructions	 87
6.1 Operating instructions for activating the IACAMA feature.....	87
6.2 Buffer Handling in EWSD.....	87
6.3 Operating instructions for customizing the IACAMA ticket.....	88
6.4 Operating instructions for controlling the generation of IACAMA-tickets ...	90
6.5 Operating instructions for handling of the cyclic file IA.ICIAR(1)(2)	90
6.5.1 Cyclic file characteristics.....	90
6.5.2 Local transfer of the cyclic file.....	91
6.5.3 Remote transfer of the cyclic file	91
6.6 Hints for the dimensioning of the IACAMA buffers and IACAMA file	94
6.7 Hints for the Postprocessing.....	95
6.7.1 Identifying the contents of the IA.ICIAR(1)(2) file	95
6.7.2 Identifying the contents of one variable length IACAMA record	96
6.7.3 Average Record length	97
 7 Delta to Previous Versions	 99
7.1.1 Version 6.2.....	99
7.1.2 Version 7.1.....	99
7.1.3 Version 8.....	99
7.1.4 Version 9.....	99
7.1.5 Version 10.....	99
7.1.6 Version 11.....	99
7.1.7 Version 12.....	99
 8 Examples	 101

0 General Information

0.1 Issue Control

The document comprises 101 pages, all pages have issue no 02.

0.2 History

History and/or delta to previous versions is shown in chapter 7

0.3 References

- /1/ Description of output data
 P30305-X3228-K***-**-7617
 EWSD
 AMA for Operator Input
- /2/ Description of output data
 P30308-X1438-K***-**-7617
 EWSD
 AMA for Operator Service System
- /3/ Description of output data
 P30308-X5114-K***-**-7617
 EWSD
 AMA for Intelligent Networks (IN) calls
- /4/ Description of output data
 P30308-X2058-K***-**-7617
 EWSD
 AMA for - PSTN subscriber charging
 - TFS service subscriber charging
- /5/ Volume Description of Output Data on Magnetic Tape
 P30305-X317-K**-**-7618
- /6/ CML V11
- /7/ ITU-T Q.850
 Usage of Cause and Location in the Digital Subscriber Signaling
 System No. 1 and the Signaling System No. 7 ISDN User Part
- /8/ ITU-T Q.763
 Formats and Codes of the ISDN User part of Signaling System No. 7
- /9/ Functional Specification (Level 2)
 P30304-A1380-J***-**-7659
 IACHASTA: Inter Administration
 Charging and Statistics
 V11
 [Internal document]

- /10/ Description of output data
P30304-A1380-K***-**-7617
IACHASTA: Inter Administration
Charging and Statistics
V10
- /11/ Description of output data
P30308-A6800-K***-**-7617
IACAMA: Inter Administration Charging by means of AMA
V11
- /12/ An Arithmetic Checksum for Serial Transmissions
John G. Fletscher
IEEE Transaction on communications
vol.com-30, no. 1
January 1982, pages 247..251

0.4 Glossary and Abbreviations

ACM	: Address Complete Message
ANM	: Answer Message
AMA	: Automatic Message Accounting
CAC	: Carrier Access Code
CIC	: Circuit Identification Code
CONN	: Connection
CTX	: Centrex
DN	: Directory Number
EOF	: End Of File
EWSD	: Elektronisches Waehlsystem Digital
HDR	: Header
IACAMA	: Inter administration Charging by AMA
IACHASTA	: Inter administration Charging and Statistics
IACMET	: Inter administration Charging by metering
ISDN	: Integrated Service Digital Network
LAC	: Local Area Code
LGC	: Large Conference Call
MML	: Man Machine Language
ODAGEN	: Online Database Generation
OSS	: Operator Service System
RID	: Record Identifier
TGRP	: Trunk Group
TMR	: Transmission Medium Requirement
TPC	: Three Party Conference Call
TRK ID	: Trunk Identification
VOL	: Volume

0.5 Keyword / Descriptor

IACAMA

0.6 List of Figures and Tables

2. Figure 1 : Trunk group as registration object	10
3. Figure 2 : PBX PA-slave as registration object.....	11
4. Figure 3 : DLU as registration object	11
5. Figure 4 : Directory number block as registration object	12
6. Figure 5: A- and B-side of a call.....	13
7. Figure 6 : Structure of file IA.ICIAR.....	15
8. Figure 7 : Format of the IACAMA record	21
9. Figure 8 : Provision of package 142 (Calling Party Number).....	54
10. Figure 9 : Third Party Number in case of Call Diversion.....	60
11. Figure 10 : Third Party Number in case of A-side Call Transfer	61
12. Figure 11 : Third Party Number in case of B-side transfer	62
13. Figure 12 : Receipt of Tariff Info before Answer (call duration < intermediate timer)	69
14. Figure 13 : Examples of Intermediate Charging in case Tariff Info is received	70
15. Figure 14 : Checksum algorithm	75
16. Figure 15 : illustration of Additional Call Indicators	79
17. Figure 16 : Example of the use of package 157	81
18. Figure 17 : Buffering of IACAMA tickets	87
19. Figure 18 : Security measures for buffers.....	88
20. Figure 19: Handling of the Cyclic File	91
21. Figure 20 : Example of FTAM flow	92
22. Figure 21 : Establishing of a copy area in a cyclic file.....	93
23. Figure 22 : Aborted FTAM transfer of a cyclic file.....	93
24.	
25. Table 1 : Overview of collected data packages	22
26. Table 2 : Layout of the Fixed Part of the IACAMA record.....	25
27. Table 3 : Overview of the Flags in the Fixed Part of the IACAMA Record	25
28. Table 4 : Values of Record Sequence	27
29. Table 5 : Values of Charge Status.....	27
30. Table 6 : Layout of GCR	73
31. Table 7 : Overview of Record Identifiers.....	96
32. Table 8 : Examples for IACAMA Record length.....	97

1 Introduction

A large percentage of telephone communication is conducted at the national and international long-distance level. Indeed, often a subscriber is called in the service area of another telecommunication administration (in this document also expressed as network operator). This is always the case for international calls at which a subscriber in an other country is called.

In some countries, it is also possible that even for national calls :

- more than one network operator serves a number of subscribers of this one country or
- the call is routed via more than one network operator who acts as carrier network. This depends upon the country's specific telecommunication structure.

The revenues for such calls must be distributed over the different network operators. To enable this inter administration revenue accounting, each network operator registers the provided traffic. A method for this traffic registration is given by the EWSD IACHASTA concept which generates so called IACAMA-tickets (IACAMA : Inter administration Charging with Automatic Message Accounting) for defined traffic nodes. A traffic node is defined by an origin and/or a destination registration object. The network operator can determine the traffic nodes for which the call data must be registered.

The generated IACAMA-tickets are transferred to a Postprocessing center, where the tickets are the base for the inter administration revenue accounting.

An alternative method offered by IACHASTA for the traffic registration is metering (IACMET). This alternative is also denoted as online charging, where durations and eventually charge units are calculated inside EWSD, and accumulated to traffic meters (as a network option eventually accumulated with the meter pulse units received backwards from the neighbouring exchange). The meter contents are transferred to a remote Postprocessing center, where these data are now the base for the inter administration revenue accounting.

This document gives a description of the IACAMA tickets, as they are stored in EWSD on the disk file IA.ICIAR (IA.ICIAR1 or IA.ICIAR2).

AMA subscriber records, AMA records for operator input (via MML) and for OSS-calls, are subject of separate descriptions. Refer to /4/, /1/ and /2/.

A description of the IACMET meters, as they are stored in EWSD on the disk file is subject of /10/.

An IACAMA-ticket registration can be defined for the following objects :

Trunk group

The different networks are connected to each other via a gateway exchange at the network border. For the registration of the incoming and leaving traffic, each incoming and outgoing trunk group of the gateway exchange is defined as an originating and/or destination registration object.

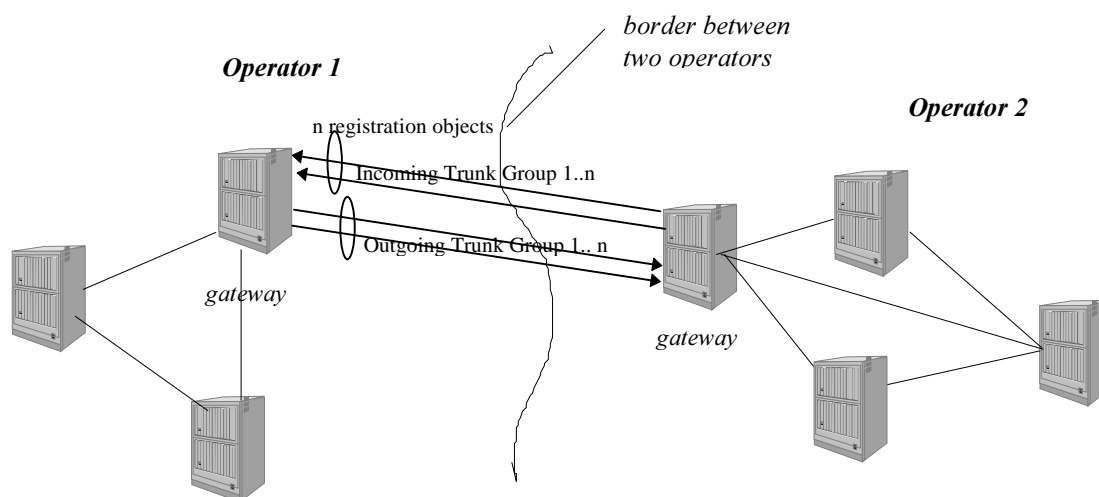


Figure 1 : Trunk group as registration object

ISDN PBX (used as PA-slave)

The inter-connection between two exchanges of different network operators is usually performed with trunk based signaling (e.g. ISUP, TUP or MFC). In this case the above mentioned objects are used as registration objects. However EWSD supports also the inter-connection between different network operators based on the ISDN-DSS1 (subscriber) signalling. In this case the inter-connection is administered in EWSD as an ISDN PBX Primary Rate Access (also denoted as "PA-slave"). Here the ISDN PBX is defined as a IACAMA registration object.

Remark : in the IACAMA ticket the incoming and outgoing side is identified by the name/number (as character string) of the involved incoming/outgoing trunk group. As in EWSD, a PBX-PA has no name/number (as character string), the PA-slave is in the IACAMA ticket identified by the number used for the AMA registration. This is usually the pilot directory (with LAC and LAC length) of the PBX registered in the field "RECORD OWNER" of the fixed part.

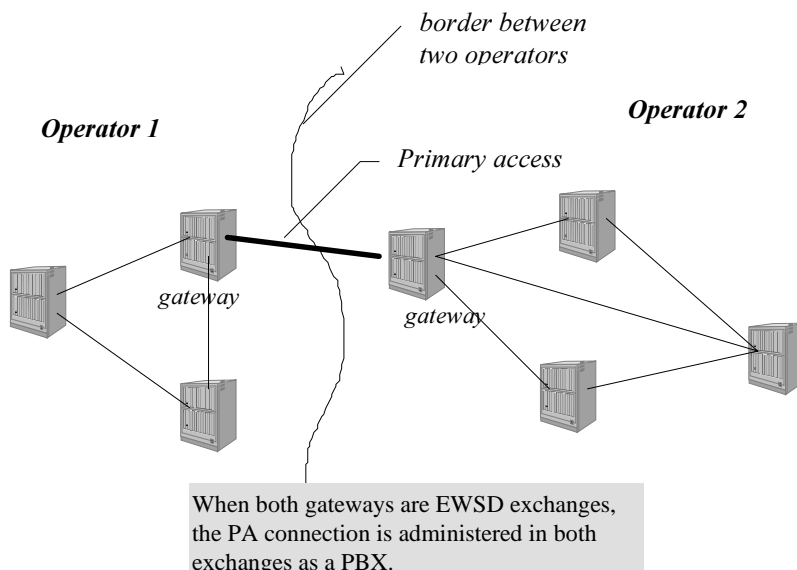


Figure 2 : PBX PA-slave as registration object

V5.x interfaces

The V5 interfaces are standard interfaces defined by ETSI. They are designed for support of Access Networks (AN) at the local exchanges.

An access network can eventually belong to an other operator.

Activating a V5.x interface as registration object allows a network operator to activate IACHASTA registration for this particular access network, allowing an easy accounting with this other operator.

Digital Line Unit (DLU)

Another inter-network connection is given by the configuration where an operator serves an own set of subscribers which are connected to EWSD with the DLU interface (with ANALOG and/or ISDN subscriber signaling).

In this case the inter-connection is administered in EWSD as DLU subscribers and the DLU as a whole is here defined as a IACAMA registration object.

Remark : in the IACAMA ticket the incoming and outgoing side is identified by the name/number (as character string) of the involved incoming/outgoing trunk group. As in EWSD, a DLU has no name/number (as character string), the DLU is in the IACAMA ticket identified by the directory number of the involved DLU-subscriber. This directory number (with LAC and LAC length) is registered in the field "RECORD OWNER" of the fixed part.

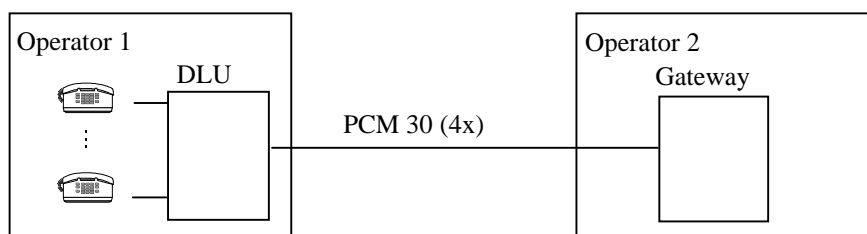


Figure 3 : DLU as registration object

Charging zone point as destination object

For each leaving traffic it is possible that depending on the destination network, a different charging zone point is assigned (e.g. zone x for network operator 2 and zone y for network operator 3).

In this case it is possible to assign an own statistic index to this zone point, and define only this statistic index as registration object.

Remark : it should be noted that although defined as registration object, the zone is not registered in the IACAMA ticket. The involved network operators are still identified by the incoming/outgoing side of the call, or eventually via the called number in the package 101 "PARTNER DN" and the calling number in the field "RECORD OWNER".

Directory number block

With the IACHASTA concept it is possible to define the registration points as 'single side' registration points (only the incoming or only the outgoing registration object) or as 'double side' registration points (a combination of incoming and outgoing registration object).

If the exchange with the IACAMA registration is not a pure relay/gateway exchange, but it has also directly connected subscribers, then it must still be possible to register all traffic going to or coming from an other operator. When it was decided to use double side registration points, combinations with the own subscribers must be activated (e.g. an own subscribers calls a subscriber via a trunk going to an other operator).

To define the own subscribers as registration object, the 'directory number block' can be used as object element.

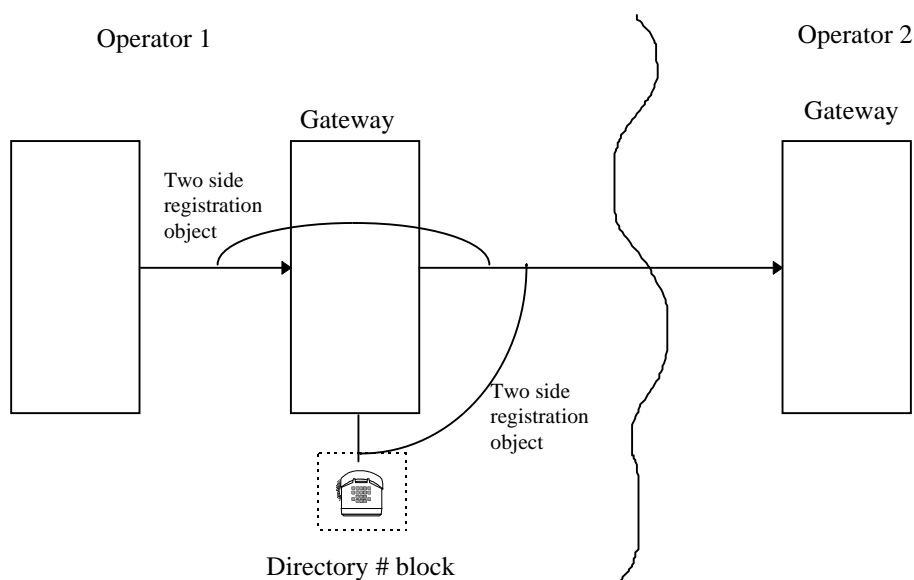


Figure 4 : Directory number block as registration object

Overview of the document

Chapter 2 (FILE FORMAT DESCRIPTION) gives a description of the file IA.ICIAR from a data processing point of view.

Chapter 3 (REMARKS ON FILE CONTENTS) contains a functional description of the IACAMA feature and gives an overview of the logical information which is collected in the IACAMA tickets.

Chapter 4 (RECORD FORMAT DESCRIPTION) describes the structure of one IACAMA record.

In chapter 5 (FIELD DESCRIPTION), all fields in the IACAMA records are described in detail. For each field, a syntax and a functional description is given, completed with operational and postprocessing remarks.

Chapter 6 (OPERATING INSTRUCTIONS) gives a short overview of the operational handling necessary for activating the AMA feature, controlling the contents of the IA.ICIAR file and transferring the IA.ICIAR file to the Postprocessing. Also some general hints for the Postprocessing are included.

Terminology

- the terms "**IACAMA ticket**", "**ticket**", "**IACAMA record**" and "**record**" all refer to the variable length record which is generated for one call or one call part. Remark the difference between an 'IACAMA record' and a 'file record' of the file IA.ICIAR : such a file-record has a fixed length of 2, 4 or 8 kbyte. The variable length IACAMA records are grouped together in one such fixed length file record, possibly with a filler record at the end (refer to chapter 2 for a detailed description). When referring to such a fixed length file-record of 2, 4, or 8 kbyte, the term "**file record**" or "**data block**" is used.
- the term "**System record**" refers to following records which can be collected together with the IACAMA records in the data blocks of the IA.ICIAR file :
 - * *Date/Time Change record*
 - * *Overflow record*
 - * *Breakdown record*
- the term "**record**" in this document refers to the IACAMA records, the System records and the fillers.
- the terms "**Postprocessing**" and "**Billing Center**" refer both to the entity outside the EWSD switch, which receives the tickets after having issued a transfer command, and which calculates the inter-administration revenues.
- the term "**crafts person**" refers to the person or persons which enter the MML-commands in the EWSD exchange.
- unless otherwise specified, the term "**subscriber**" does not only refer to analog and ISDN subscribers but also to analog and ISDN PBX's. In this sense, it is opposed against the term "**trunk**", which is used to denote the inter-connection (analog, TUP, ISUP, ...) between two exchanges.
- the notation "**H'**" is used for the hexadecimal notation. Example : H'84 is equivalent to the decimal value 132.
- the "**A-side**" and "**B-side**" of a call refer only to the exchange which is currently considered for the ticket generation (the A-side is the trunk on which the seizure arrives in the exchange generating the IACAMA ticket).

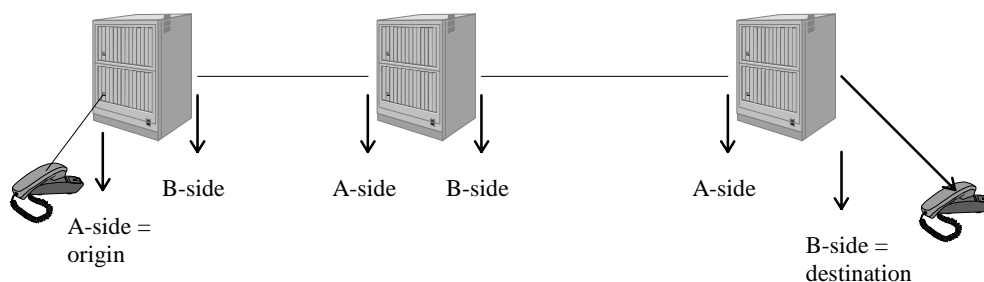


Figure 5: A- and B-side of a call

General remark concerning MML commands

At many locations in this document, MML commands are mentioned. These MML commands **do not tend to be complete**; they are just meant to clarify the relation between IACAMA and Postprocessing. When e.g. only a create command (CR ..) is mentioned to administer a particular information field, it is quite possible that also the corresponding modification command (MOD ...) can be used. Parameters which have to be replaced by real values are written between < >. For a detailed syntax description of each MML-commands, refer to /6/ and /9/.

2 File Format Description

2.1 File format on disk

On EWSD disk, the IACAMA records are stored in one cyclic file IA.ICIAR or in two cyclic files IA.ICIAR1 and IA.ICIAR2. From a data processing point of view, this file has fixed length records of 2, 4 or 8 Kbyte (the actual value can be administered). One such a file record contains several variable length IACAMA records, possibly completed with a filler record.

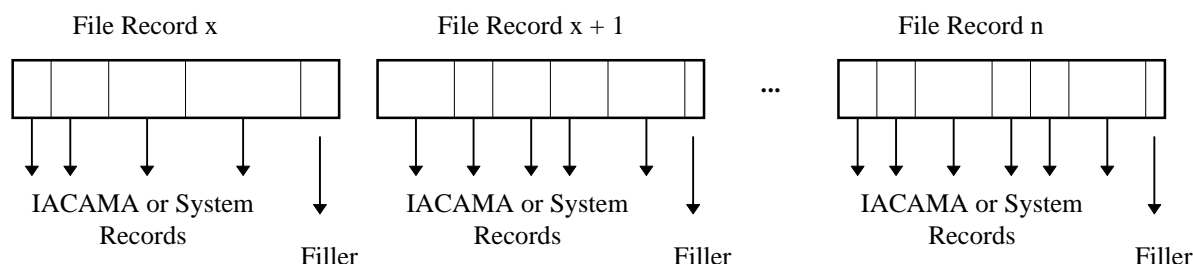


Figure 6 : Structure of file IA.ICIAR

A cyclic file has following characteristics :

- File records can continuously be written into the file. When the max. file size is reached, the oldest stored file records are overwritten, on condition those records have been marked as overwriteable.
- The file contains a header with control information. This control information contains a.o. :
 - * the write pointer, which marks the position in the cyclic file where the next file record has to be written to
 - * pointers which mark the begin and the end of the copy area. The copy area is that part of the cyclic file which has been transferred, but has not yet been released for overwriting. Each time a new transfer is issued on the cyclic file, the contents of the copy area is transferred again.
 - * the number of times the copy area has been transferred

For a detailed description of cyclic file handling in EWSD, refer to chapter 6 (OPERATING INSTRUCTIONS)

2.2 File format on tape or MOD

In case of a transparent copy from the IA.ICIAR file to tape or Magnetical Optical Device (MOD), the file on tape or MOD is structured in the same way as the IA.ICIAR file on disk, i.e. in fixed length data blocks of 2, 4 or 8 Kbyte containing the individual IACAMA records with variable length.

The information on tape is organized according international standards (DIN 66029, May 79 and ISO 1001-1979). Refer to /5/ for a detailed description of tape organization, Volume Labels and Tape Marks. Following parameters are IACAMA specific :

- File Identifier
The IACAMA file on tape is recommended to be identified by the file name IA.ICIAR.XXXXXXXX (IA.ICIAR1.XXXXXXXX and IA.ICIAR2.XXXXXXXX, where the positions XXXXXXXX identify the EWSD exchange. The file name is entered with the MML command to copy the IACAMA disk file on magnetic tape or MOD.
- Block length : The Block Length is equal to the File Record Size, i.e. 2, 4 or 8 kbyte.
- File Record Size: 2, 4 or 8 kbyte.
- File Record Format: Fixed (F)

3 Remarks on File Contents

3.1 IACAMA Records

Following records can be present in the IA.ICIAR file :

- Call Detail Records (CDR's) for inter administration charging (IACAMA).
- System Records
These records are used to register some switch events which are important for charging :
 - * Date/Time/Change Records to register the change of time in the switch
 - * Breakdown Records to register a recovery in the switch
 - * Overflow Records to register the fact that tickets have been lost due to overflow conditions in the switch
- Filler Records
These are used to fill out the 2, 4 or 8 Kbyte data block in the IA.ICIAR file

System and filler records are described in detail in chapter 5.

3.2 IACAMA Record Generation

3.2.1 Inter administration charging

Following information is or can be gathered in the inter administration tickets :

- *Date/time and Duration*

The **Date/time** and the **Duration** are always collected in IACAMA tickets.

When no intermediate charging is done, the Date/time either refers to the begin date/time of the call, i.e. the time when the answer signal of the B-party is received, or as a network option to the end date/time of the call, i.e. the time when the release signal of either A- or B-party is received. The Duration is the time interval between answer and release.

In case intermediate charging is performed, the Date/time and Duration refer to the call part for which the ticket is generated, unless Cumulated Duration is required. Refer to chapter 5.1.2.1 for more information.

Durations Before Answer is gathered in the first or single IACAMA ticket for the connection ticket. Durations Before Answer consists of :

- * duration between seizure and receipt of Address Complete signal of the B-party
- * duration between Address Complete and Answer

Date/time and Duration for inter administration tickets are stored in **package 100**. Durations Before Answer is collected in **package 134**.

- *All kinds of digits*

- * **Record Owner**

The digits identifying the **calling party number**.

- * **Partner Directory Number** (package 101)

For **non-IN calls**, the B-party number is stored in **package 101** (Partner Directory Number).

- * **New Destination** (package 128)

For IN-calls, the B-party number is stored in **package 128** (New Destination); package 101 contains in this case the dialed IN number.

- * **Carrier Access Code** (package 145)

The **Carrier Access Code** which is used to reach the destination via a specific carrier is stored in **package 145**.

- * **Third Party Number** (package 146)

Package 146 (Third Party Number), when activated by the crafts person, is generated in case of:

- Call Forwarding, where it contains the DN of the original calling party number
- in case of call transfer. At the moment of call transfer, intermediate accounting is performed. After call transfer, the IACAMA tickets of the controller may contain the public DN of the third party, when requested by MML and when available.

- *Online Charge information*

When activated by the crafts person, **package 103 (Connection Charge Units)** is collected in the AMA ticket.

The Connection Charge Units are the own generated units based on the zone via tariff administration and/or the units received from a higher exchange.

Package 138 (B Meter Pulses) contains only the meter pulse units received from the higher exchange.

- *Security Information*

The **Connection Identification**, stored in **package 110**, is a unique identification for one call. The same Connection Identification is present in all intermediate IACAMA tickets for one call.

The **Global Charge Reference** is meant to be a unique identification for all kinds of tickets generated in the own and in different exchanges for the same call. In the common EWSD V12 functionality, the Global Charge Reference is not transported via ISUP; it is therefore only useful within one exchange. When activated by the crafts person, the Global Charge Reference is collected **instead of** the Connection Id.

Package 150 (Audit Trail) contains two kinds of information, separately administrable by the crafts person. The **Sequence Number** is a running number, increased with 1 for each new ticket collected in the AMA file. The **Checksum** is calculated with a certain algorithm on the bit pattern of the ticket. By repeating the calculation, the Postprocessing can verify whether all bits have kept their integrity during transfer and further manipulation of the AMA tickets.

- *Other kinds of information*

Package 102 (Service Info) collects the used ISDN service, and is therefore only relevant for ISDN calls.

The **incoming and outgoing Trunk Identifications and the Outgoing PA Slave Identification** are collected in the ticket. The trunk identifications can be stored in two different formats. **Packages 105, 106, 107 and 108** are used for these trunk identifications. **Packages 107 and 108** only appear when activated by the crafts person.

Package 119 (Transmission medium required) indicates the transmission medium which is originally required by the ISDN calling party.

Package 120 (Category) gives more information about the calling party (e.g. whether the call was originated from a coin box, or from a test equipment).

Package 130 (Traffic Quality data) gives more information about the location and the reason why the call is released.

- *Project Specific Information*

Each EWSD version contains a package of features which are denoted as “common features”. A specific project can get additional features. When these additional features require charging specific data to be collected in the AMA-ticket, **package 117 (Project Specific Data)** is used.

3.2.2 Intermediate Charging

A call is recorded either by one single ticket or by several intermediate tickets.

An intermediate ticket is generated in following cases :

- in case of long duration calls, i.e. calls for which the conversation duration exceeds the intermediate time duration. This intermediate time duration is administrable between 15 minutes and 2 hours. The default value is 30 minutes.
- when a new tariff information is received or sent during the call.
- in case of large conference and three party conference, at the moment the conference starts or ends (on condition the IACAMA ticket is generated on the side of the subscriber initiating the conference).
- in case of call transfer at the moment the transfer is invoked (on condition the IACAMA ticket is generated on the side of the subscriber making the call transfer).
- in case the ISDN service is changed during the call for an ISDN 1TR6 (Redbook) subscriber (on condition the IACAMA is generated on the side of the 1TR6 subscriber).
- in case the SPC (Semipermanent Connection) feature is activated or deactivated for an ISDN 1TR6 (Redbook) subscriber (on condition the IACAMA is generated on the side of the 1TR6 subscriber).

- in case the number of charge units during the call must be registered, and exceeds 65535
- in some cases of recovery

The Record Sequence field in the fixed part of the IACAMA ticket indicates whether only one or several tickets are generated for the call. In the latter case, the Record Sequence indicates the first, the intermediate and the last ticket generated for the call.

3.2.3 IACAMA for not answered calls

- As default (i.e. without administration of the marks UCALLIAC with the command ENTR CDTDAT), IACAMA tickets are generated only for successful calls, i.e. calls which have been answered.
- If the registration of unanswered calls is also required, the mark UCALLIAC has to be set. If this mark is set, IACAMA tickets are generated for all successful and unsuccessful calls, which have to be charged with IACAMA.

For more information, refer to the description of the Charge Status in chapter 5.1.1 and to the Operating Instructions for controlling the generation of IACAMA-tickets.

4 Record Format Description

4.1 IACAMA Record

An IACAMA Call Detail Record consists out of two main parts :

- a fixed part, with a fixed structure
- a variable part, which contains several data packages

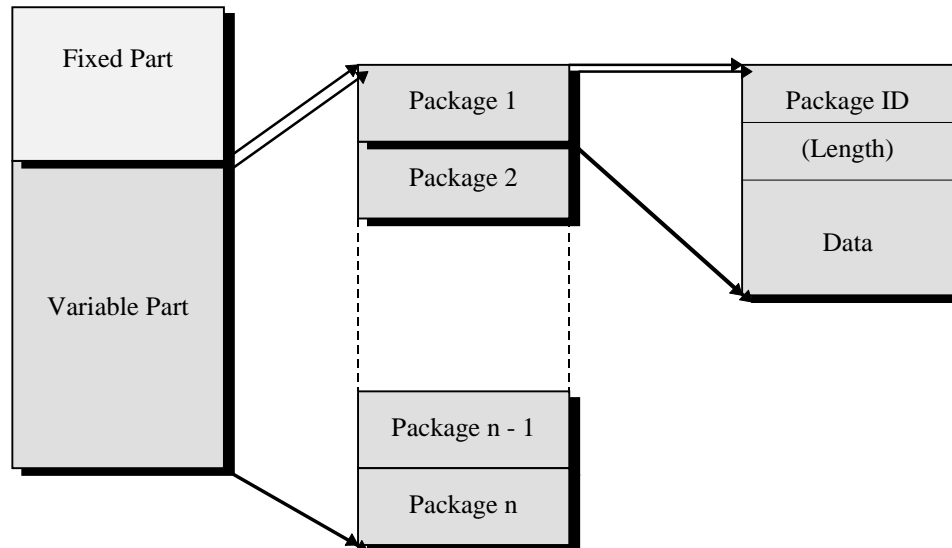


Figure 7 : Format of the IACAMA record

Fixed Part

The fixed part is the "header" of the record and contains several standard fields. For a detailed description of the respective fields, refer to chapter 5.1.1.

Variable part : data packages

The fixed part is followed by several data packages. The presence of a data package depends on several criteria :

- collection is or is not activated by the crafts person
e.g. package 103 (Charge units for connection)
- the data item is or is not relevant for the call
e.g. package 102 (Service Info) for analog calls
- the value of the data item is the default value
e.g. package 120 (Category) in case the category = "subscriber"

The first field of each data package contains the package identification (a unique number). Each package has its own layout. In most of the data packages, the package length is indicated at the second byte.

Following table gives an overview of the data packages collected in IACAMA Records.

For a detailed description of the data packages, and the conditions under which the data packages are collected, refer to chapter 5.1.2.

Package number	Package name	Package length	Package length identification (in bytes)	Presence
100	DATE / TIME / DURATION	fixed	none : 11	x
101	PARTNER DIRECTORY NUMBER	variable	calculated	x
102	SERVICE INFO	fixed	none : 4	[x]
103	CHARGE UNITS FOR CONNECTION	fixed	none : 4	opt
105	TRUNK ID INCOMING	fixed	none : 9	[x]
106	TRUNK ID OUTGOING	fixed	none : 9	[x]
107	TRUNK ID INCOMING WITH CIC	fixed	none : 10	[opt]
108	TRUNK ID OUTGOING WITH CIC	fixed	none : 10	[opt]
110	CONNECTION IDENTIFICATION	fixed	on byte 2	[opt]
117	PROJECT SPECIFIC DATA	variable	on byte 2	*
119	TRANSMISSION MEDIUM REQUIRED	fixed	none : 2	[x]
120	CATEGORY	fixed	none : 2	[x]
128	NEW DESTINATION	variable	calculated	[x]
130	TRAFFIC QUALITY DATA	fixed	on byte 2	[x]
134	DURATIONS BEFORE ANSWER	fixed	on byte 2	[x]
138	B METER PULSES	fixed	on byte 2	opt
142	CALLING PARTY NUMBER	variable	on byte 2	opt
145	CAC NUMBER	variable	on byte 2	[opt]
146	THIRD PARTY NUMBER	variable	on byte 2	[opt]
147	TARIFF INFO	variable	on byte 2	[opt]
148	CAC NEW DESTINATION	variable	on byte 2	[opt]
149	GLOBAL CHARGE REFERENCE	variable	on byte 2	opt
150	AUDIT TRAIL	fixed	on byte 2	opt
154	ADDITIONAL CALL INDICATORS	variable	on byte 2	opt
157	OUTGOING PA SLAVE IDENTIFICATION	variable	on byte 2	opt

Table 1 : Overview of collected data packages

Legend:

- x Data package is **always** collected in all tickets for the call.
- [x] Data package is **not always** collected, i.e. the data package is only collected when certain **conditions** are fulfilled. The collection of the package is **not administrable**.
- opt Optional data package, i.e. the collection of the data package depends on the **administration** by the crafts person by means of the MML-command ENTR CDTDAT. The collection of the data package depends **only** on the administration, i.e. if the collection of the data package is administered, then the package is **always** collected in all tickets for the call.
- [opt] Optional data package, which depends not only on the **administration** by means of MML-command ENTR CDTDAT, but also on other **conditions**.
- * Data package is collected on a project specific basis, i.e. the collection of this package is not part of the common EWSD functionality.

4.2 System and Filler Records

System records (Data/Time Change, Breakdown and Overflow) have a fixed length of 32 byte. Filler records have three formats :

- the Record Identifier H'80 is used as Filler record to fill in one byte
- the Record Identifier H'81 is used as Filler record to fill in several bytes

- the Record Identifier H'00 is used as Filler record to fill in 32 bytes

The exact layout of the System and Filler Records is described in chapter 5.

4.3 General Record Coding Details

4.3.1 Coding of 'binary' fields

Binary fields are stored in LSB_LO format, i.e. the least significant byte of the binary number is stored at the lower memory location.

EXAMPLE : Storage of a binary integer in a 3 byte field

Integer = 250 (H'FA = H'0000FA)

Bit position			
7 6 5 4 3 2 1 0		or	
byte 1	1 1 1 1 1 0 1 0	FA	value = 250 = H'FA
byte 2	0 0 0 0 0 0 0 0	00	value = 0 = H'00
byte 3	0 0 0 0 0 0 0 0	00	value = 0 = H'00

Integer = 1000 (H'3E8 = H'0003E8)

Bit position			
7 6 5 4 3 2 1 0		or	
byte 1	1 1 1 0 1 0 0 0	E8	value = 232 = H'E8
byte 2	0 0 0 0 0 0 1 1	03	value = 3 = H'03
byte 3	0 0 0 0 0 0 0 0	00	value = 0 = H'00

4.3.2 Coding of "packed digit" fields

The expression "packed digit" means that two digits are packed into one byte. The digits on odd positions of a digit string are stored into the high order part of a byte. The digits on even positions of a digit string are stored into the low order part of a byte. The first digit of the string is stored in the first byte of the digit field.

EXAMPLES :

1) Digit String of 6 digits : 123456 (e.g. dialed digits 1 2...6)

Bit position					
7	6	5	4	3	2 1 0
1				2	byte 1 of the digit field
3				4	byte 2 of the digit field
5				6	byte 3 of the digit field

2) Digit String of 7 digits : 1234567 (e.g. dialed digits 1 2...7)

Bit position							
7	6	5	4	3	2	1	0
1				2			
3				4			
5				6			
7				(*)			

byte 1 of the digit field

byte 2 of the digit field

byte 3 of the digit field

byte 4 of the digit field

(*) : The value of the low order part of byte 4 is undefined. Remark that all the digits fields have a corresponding length field that indicates the number of digits.

4.3.3 Coding of “character” fields

Character fields are coded in ISO-7-Bit-Code. The first character is stored at the lower memory location of the field. Trailing bytes are filled up with spaces.

EXAMPLE : store MUNICH in a character field that has a length of 8 bytes.

Bit position							
	7	6	5	4	3	2	1 0
M			4				D
U			5				5
N			4				E
I			4				9
C			4				3
H			4				8
space			2				0
space			2				0

byte 1 of the character field

byte 2 of the character field

byte 3 of the character field

byte 4 of the character field

byte 5 of the character field

byte 6 of the character field

byte 7 of the character field

byte 8 of the character field

5 Field Description

5.1 AMA Inter administration charging call Record

5.1.1 Fixed part (Header)

A) Syntax description

Field No.	Byte Pos	Length	Field Name	7	6	5	4	3	2	1	0	Remarks
1	1	1	RECORD IDENTIFIER									binary
2	2	2	RECORD LENGTH									binary
			res									
3	4	3	res	res	res	res	F14	res	F12	res		FLAGS
			res	res	res	F25	F24	res	res	res		
			F38	F37	res	res	res	F33	res	res		
4	7	1	RECORD SEQ.					CHARGE STATUS				binary
5	8	m	LAC LENGTH					OWNER ID LENGTH				packed digits
			LOCAL AREA CODE (LAC) + DIRECTORY NUMBER (DN)									

Table 2 : Layout of the Fixed Part of the IACAMA record

Description of Field No 3 : FLAGS

Flag	Value	Meaning
F12	1 0	IACAMA no IACAMA
F14	1 0	CONNECTION no CONNECTION
F24	1 0	Analog subscriber no Analog subscriber
F25	1 0	ISDN subscriber no ISDN subscriber
F33	1 0	no ANSWER ANSWER
F37	1 0	sequence number in package 150 no sequence number in package 150
F38	1 0	checksum in package 150 no checksum in package 150

Table 3 : Overview of the Flags in the Fixed Part of the IACAMA Record

B) Functional description

Record Identifier

The Record Identifier 132 (H'84) identifies IACAMA records. For a complete list of all possible record identifiers which can be present in the IA.ICIAR file, refer to chapter 6.7.1.

Record length

The record length refers to the total number of bytes in the IACAMA record, i.e. including Fixed Part and all packages.

For an estimation of the average length of the IACAMA record, refer to chapter 6.7 ("Hints for the Postprocessing").

Flags

- **Flag F12 : IACAMA**

Flag F12 = 1 indicates that an IACAMA ticket is made.

Remarks :

- * for IACAMA tickets this flag is always set (F12 = 1)

- **Flag F14 : CONNECTION**

This flag indicate that the ticket is made for a connection

Remarks :

- * for IACAMA tickets this flag is always set (F14 = 1)

- **Flag F24 : Analog subscriber**

This flag indicates that the A-party is an analog subscriber.

Remarks :

- * this flag is **not** relevant (i.e. it is **never set**) for IACAMA tickets in the transit exchange

- **Flag F25 : ISDN subscriber**

This flag indicates that the A-party is an ISDN subscriber.

Remarks :

- * this flag is also relevant for IACAMA tickets in the transit exchange. However, it could be that due to signalling restrictions (analog trunks between the A-party and the transit exchange), the flag is not set, although the A-party was an ISDN subscriber. Remark that in this case the flag F24 is also **not** set.

- **Flag F33 : NO ANSWER**

This flag indicates that the call has not been answered.

Remarks :

- * in EWSD, a call is treated as successful as soon as Answer has been received. When the feature "IACAMA for unsuccessful calls" is not activated (which is the default), all IACAMA tickets in the IA.ICIAR file will have the flag F33 = **not** set, i.e. F33 = 0, indicating that the call is answered.

- **Flag F37 : Sequence Number in package 150**

This flag indicates that the Sequence Number is located at the following position : end of record - 6 bytes.

- **Flag F38 : Checksum in package 150**

This flag indicates that the Checksum is located at the two last bytes of the record.

Record Sequence

Following values are foreseen :

Value	Meaning
0	not used
1	single
2	first intermediate
3	intermediate
4	reserved
5	final
6 - 15	reserved

Table 4 : Values of Record Sequence

The Record Sequence is related to intermediate charging : when only one ticket is generated for a call, the Record Sequence of this ticket = "single". When more than one ticket is generated for the same call, the generated tickets will have as Record Sequence : "first intermediate" - "intermediate" - "final". The Record Sequence "intermediate" can be absent when only 2 tickets are generated, or is repeated when more than 3 tickets are generated for the same call.

Charge Status

Following values are foreseen :

Value	Meaning
0	undefined
1	charge
2	no charge
3 - 15	reserved

Table 5 : Values of Charge Status

For **answered calls**, the Charge Status is normally "**charged**". Only in following cases, the Charge Status is "**no charge**" :

- in case of an internal call (A- and B-subscriber connected to the same exchange), if the B-party has the feature "NOCHARGE" (parameter NOCHARG in the CR SUB/PBX commands)
- in case of a call going out over an ISUP trunk, in case the Backward Call Indicator "no charge" is set in the ISUP backward messages ACM (Address Complete) or ANM (Answer)
- in case a call is made to a charge free announcement

For **not answered calls**, the Charge Status can have following values :

- if the call is released before the (project specific) criteria for call attempt are fulfilled, the Charge Status is set to "**undefined**"
- if the call is released after the criteria for call attempt are fulfilled, the Charge Status is set to "**no charge**" in following cases :
 - * non-zero call attempt charges are administered and the B-party is connected to the same exchange and has the "NOCHARGE" feature
 - * non-zero call attempt charges are administered and the ISUP ACM Message has been received with Backward Call Indicator "no charge"
 - * non-zero call attempt charges are administered and the B-party is a non-chargeable announcement
 - * no call attempt charges are administered, but the mark ATTNNOCHA (charging of charge free call attempts) is set with the command ENTR CDTDAT

- if the call is released after the criteria for call attempt are fulfilled, the Charge Status is set to “**charge**” in case non-zero call attempt charges are administered and none of the above mentioned criteria for “no charge” are fulfilled.

Remark :

- It is also possible to make calls free of charge by suppressing the Answer signal. This is in fact the only possibility in case analog trunks are present on the call path, which are not able to send the indication no charge. For such unanswered calls, no IACAMA tickets are generated, unless “IACAMA for unsuccessful calls” is administered. In this case, IACAMA tickets for unanswered calls are detected by the Postprocessing based on Flag F33 (No Answer).

Record Owner

The Record Owner consists out of the fields LAC LENGTH, OWNER ID LENGTH and the Directory Number including the LAC, if this LAC is present. The OWNER ID LENGTH includes the LAC digits.

The LAC LENGTH has a maximum value of 6.

The Directory Number contains at most 31 digits (including LAC). The national or international prefix digits are **not** included.

Hexadecimal digits (A till F) can be present.

The Directory Number identifies either

- the number of the A subscriber as present in the EWSD database, in case an IACAMA record is generated in the originating exchange.
- the inter-office signaled calling party number (if present), in case an IACAMA record is generated on an incoming trunk in a transit or terminating exchange.

Remarks :

- In case the IACAMA record is generated in the transit or terminating exchange, the LAC LENGTH is always equal to zero, because the number of area code digits is not signalled via trunk, except in case the IACAMA ticket is generated on an incoming trunk with a LAC.
- When the A-side is a PBX and an IACAMA record is generated in the originating exchange, the Record Owner in the IACAMA ticket is either the Pilot Directory Number or the Line Service Number (i.e. the Directory Number of the PBX line) . For IACAMA records generated in a transit or terminating exchange, the Record Owner is always the Calling Party Number (if present). Refer to the description of the package 142 (Calling Party Number) for more information.
- In case the A subscriber has the Multiple Subscriber Number (MSN) feature and an IACAMA record is generated in the originating exchange, the Record Owner contains the number which has been signaled as calling party number, on condition this number has been recognized by EWSD as one of the MSN numbers; otherwise, the default number from the EWSD database is taken as Record Owner.

C) Operating remarks

- An announcement is normally free of charge, unless it is made chargeable via the command CR ANGRP : TGNO = <announcement group>, GCOS = CHARGABL;
- The Record Owner in case of PBX calls is administered with the parameter CHRG = AMAPDN (Record Owner is the Pilot Directory Number) or CHRG = AMALSN (Record Owner is the Line Service Number) in the CR PBXLN command.
- A LAC can be assigned to a trunk group via the command CRTGRP : TGNO = <trunk group>, LAC = <area code>;

D) Postprocessing remarks

- In case IACAMA tickets for unsuccessful calls are required and package 134 (Durations before Answer) is collected by means of ENTR CDTDAT, Flag F33 (NO Answer) must be used to detect tickets for unsuccessful calls. A call duration = 0 (in package 100 : Date/Time/duration) is **not** sufficient to decide that the call has not been answered : since the duration has an accuracy of one second, it is possible that the call duration for an answered call equals zero.
- Not used flags have **random values**, i.e. they do **not** have a **fixed value 0**.

5.1.2 Part 2 : Data Packages

5.1.2.1 Package 100 : DATE / TIME / DURATION

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (100)								binary, fixed value 100 (H'64)
2			DATE / TIME								
	2	1	year								binary, value 0-99 (H'00 - H'63)
	3	1	month								binary, value 1-12 (H'01 - H'0C)
	4	1	day								binary, value 1-31 (H'01 - H'1F)
	5	1	hour								binary, value 0-23 (H'00 - H'17)
	6	1	minutes								binary, value 0-59 (H'00 - H'3B)
	7	1	seconds								binary, value 0-59 (H'00 - H'3B)
3	8	1	F8	F7	reserved				refer to description of FLAGS		
4	9	3	DURATION								binary

Description of Field No 3 : FLAGS

Flag	Value	Meaning
F7	0	begin date time
	1	end date time
F8	0	time secure
	1	time insecure

Description of Field No 4 : DURATION

DURATION : elapsed conversation duration in seconds

B) Functional description

For **answered** calls, the contents of this package depends on the setting of the mark DURATCUM (Cumulated Duration) in the ENTR CDTDAT command :

- if **Cumulated Duration** is required (DURATCUM has been specified in the ENTR CDTDAT command), the duration is measured from **answer to release**. The date/time is either the **date/time of answer** (Flag F7 = 0) or the **date/time of release** (flag F7 = 1).
- if **Cumulated Duration is not required**, this package contains the date, time and the duration of the **call part** to which the ticket refers.

For **not answered** calls, the duration is always zero.

Remarks :

- The call duration never exceeds the value as indicated by the intermediate timer T114, except when Cumulated Duration is required (all administrable timer values in EWSD are indicated by a running number; see also Operating remarks). This timer has a value between 15 minutes and 2 hours. When the call lasts longer than the duration indicated by T114, an intermediate ticket is generated (intermediate tickets are characterized by a Sequence Info = FIRST INTERMEDIATE or INTERMEDIATE). It is however quite possible, that in intermediate tickets the value is less than the value indicated by T114.

- The duration has an accuracy of 1 second. This implies that **a duration of 0 seconds is possible**, in exceptional cases.
- “Time insecure” means that the date/time indication is not guaranteed to be completely accurate. A typical case in EWSD where the time is insecure is at the start of the APS, when the time is entered with the command ENTR TIME, but has not yet been adjusted with CORR TIME.

C) Operating remarks

- the choice between begin or end date/time is made via command
ENTR CDTDAT : ALTNSD = RANSWTIM;
- the intermediate timer T114 is administered via command :
MOD TIOUT : TIMER = T114,TIMVAL=<hours-minutes-seconds-milliseconds>;

D) Postprocessing remarks

- This package has a fixed length of 11 bytes. The package length is **not** included in the package itself.
- A possible use of the flag F8 (time secure/unsecure) is to charge the subscriber with the lowest tariff when this flag is set. Remark that inside EWSD the lowest tariff is applied for metering when the time is found to be insecure.

5.1.2.2 Package 101 : PARTNER DIRECTORY NUMBER

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (101)								binary; fixed value 101 (H'65)
2	2	1	reserved				NO OF DIGITS				binary; range (0:31)
3	3	n	DIGIT (n - i)				DIGIT (n - i + 1)				packed digits, values (H'0:H'F)

B) Functional description

This package contains the called party directory number, with which the call to the called party is set up. This number is equal to the digits dialled by the subscriber, or sent by the subscriber terminal as, dialled digits, with following exceptions :

- when a service code is dialled before the B-party Number.
Package 101 contains only the B-party number, not the service code.
Examples : when an Account Code is dialled, or when the access code for DEB Next Call is dialled before the B-party number.
- when the dialled digits are translated into a long number according to a Supplementary Service. In this case, the long number is provided by the EWSD database. Package 101 contains the long number.
These cases are :
 - * when the A-subscriber uses abbreviated dialling
 - * when the A-subscriber uses Voice Mail Service (VMS)
 - * when the A-subscriber uses the hot line or the hot line delayed feature
 - * when a Centrex subscriber dials an intercom number or a CTX access code

Remarks :

- The dialled national or international prefix is included in the Partner Directory Number.
- The digits contained in this package are **not necessarily equal to the digits sent out over a trunk to the next exchange**. Inside EWSD, the dialled digits can be converted to other digits. The digits used to determine the outgoing trunk in the own switch can again be modified before they are sent out as called line digits over this trunk to the next exchange.

C) Operating remarks

- The package is not administrable.
- Conversion of the dialled digits to other digits is done via the parameters NEWCODE or CONVCODE in the command CR DEST. With these converted digits, the routing (the EWSD process to find a destination based on the digit information) is performed again.
CR DEST : DEST = <destination>, CONVCODE = <converted digits>;
CR DEST : DEST = <destination>, NEWCODE = <converted digits>;
- Modification of the digits before they are sent out over a trunk as called line digits is done with the parameters DICON and SSDI in the MML command
CR ROUTE : ROUTE = <route>, DEST = <destination>, TGNO = <trunk group name>, DICON = <digit converting code>, SSDI = <starting position for sending out dialled digits>;

D) Postprocessing remarks

- This package is always present in an inter administration ticket.
- The package length is **not** included in the package itself. The package length has to be calculated in the following way: package length in bytes = $2 + \lceil \text{integral part of the division } (\text{NO_OF_DIGITS} + 1) / 2 \rceil$

5.1.2.3 Package 102 : SERVICE INFO

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (102)								binary, fixed value 102 (H'66)
2	2	1	SERVICE INDICATOR								binary, refer to description of Service Indicator
3	3	1	reserved for ADDITIONAL INFORMATION								binary
4	4	1	reserved								Flag F1 : 0 : Bluebook 1 : Redbook

Description of Field No 2 : SERVICE INDICATOR

Value	Meaning
1	reserved
2	CM (Circuit Mode) Speech
3	CM Unrestricted Digital Info
4	CM 3.1 kHz Audio
5	CM 7 kHz Audio
6	- CM Service 06 Res
7	- CM Service 07 Res
8	Unknown Bearer Service
9	PM (Packet Mode) B-Channel
10	PM D-Channel
11	PM BD-Channel
12	Telephony ISDN 3.1 kHz
13	Telephony ISDN 7 kHz
14	Telefax Gr 2/3
15	Telefax Gr 4
16	Teletex 64 kBit/s
17	Videotex
18	Videotelephon
19	SERV19
20	SERV20
21	SERV21
22	SERV22
23	SERV23
24	SERV24
25	SERV25
26	SERV26
27	SERV27
28	Semipermanent connection
29	Non Voice Group
30	Voice Group
31	All Service Group

Description of Field No 3 : ADDITIONAL INFORMATION

Project specific.

B) Functional description

This package contains the Service as defined in EWSD. This package is only collected when the calling line is an ISDN subscriber.

Inside EWSD, the Service is determined in the following way :

based on Bearer Capability, High Layer Compatibility and Low Layer Compatibility information received from the ISDN subscriber, an **internal** service is built, which has a range from 1 to 18. This internal service is mapped onto an **administrative service**, which has a range from 1 to 31. This administrative service is used to check whether the subscriber has the authorization for this service, by comparing this administrative service with the services administered to the subscriber. Remark that the mapping table between internal and administrative service can be **modified on project specific basis**. It is e.g. possible that a project uses only the administrative services for Non Voice Group (29) and Voice Group (30).

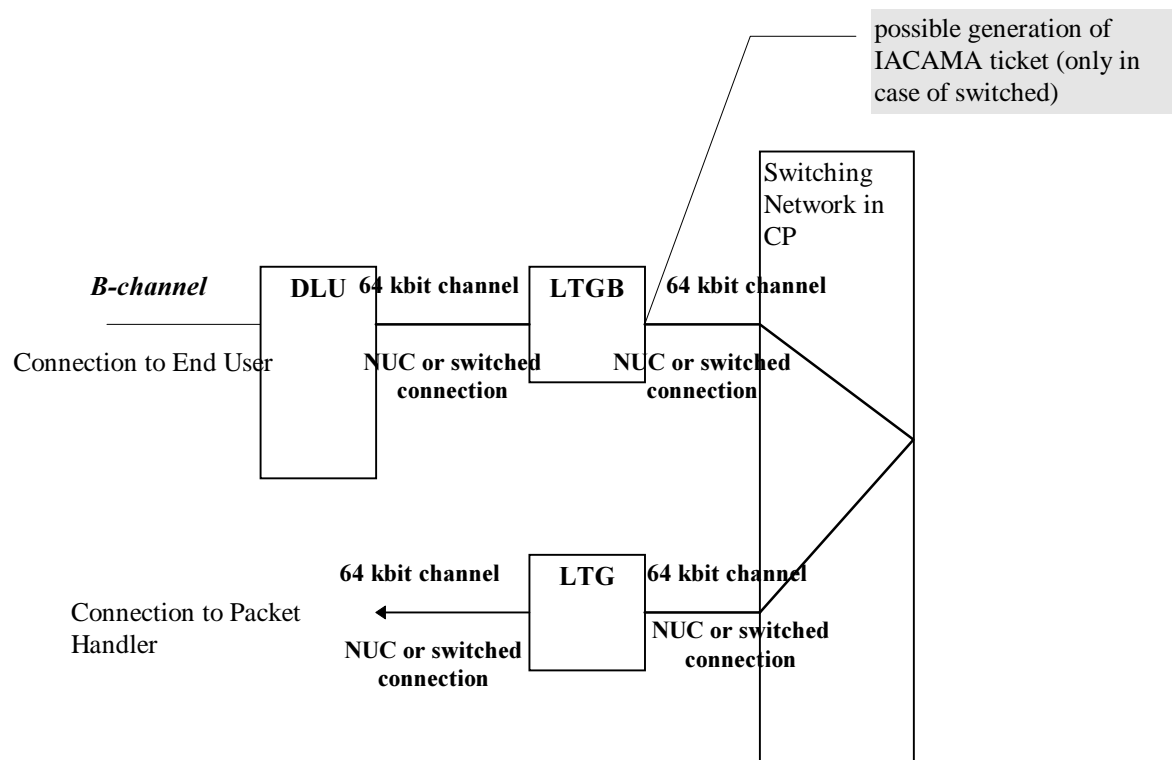
In the IACAMA tickets, the internal service is registered in package 102 (SERVICE INFO)

Remarks :

- This package is also provided in IACAMA tickets on incoming trunks, on condition ISUP is present all the way between the origin and the exchange generating the IACAMA ticket
- The table above lists all values which are possible inside EWSD. The actual values which can turn up in the IACAMA tickets depend however of **project specific settings**.
- Service **10** (PM D-channel) will in praxis **never** be present in an IACAMA ticket. The other Packet Mode services 9 and 11 are only present under limited circumstances. The following gives a description of the handling of the Packet Mode services :

* **PM (Packet Mode) B-Channel (value 9)**

This value is used when a B-channel service is requested. A B-channel service means that the Packet Data are running over a B-channel from the end-user to the DLU (Digital Line Unit : peripheral processor to which analog and ISDN subscribers are connected). From the DLU to the Packet Handler, either a Nailed up Connection or a Switched Connection is present.



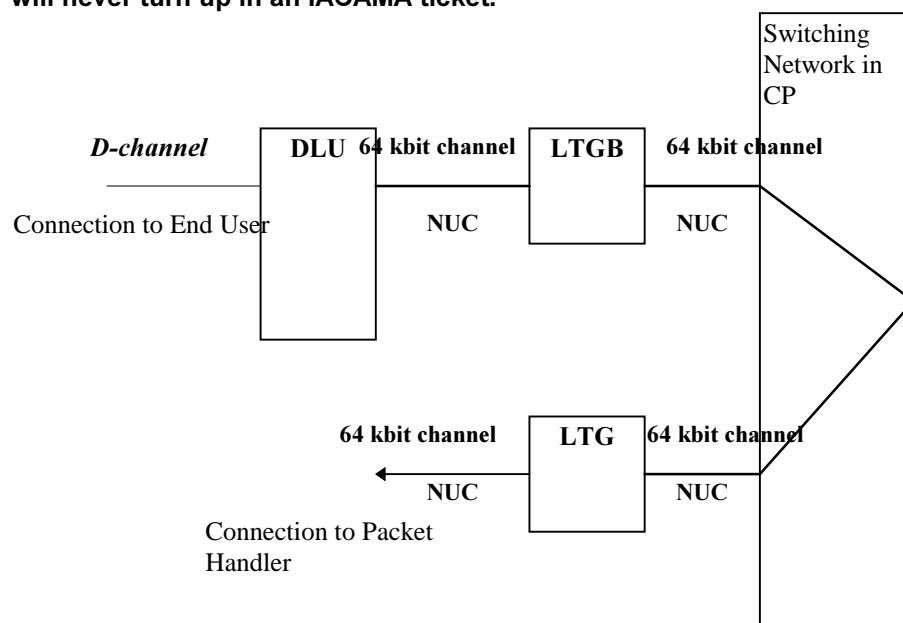
An IACAMA ticket can only arise for a switched connection between DLU and Packet Handler. In this case, the service will indicate "PM B-channel".

Remark however, that the IACAMA-ticket has no information at all concerning the packet data which are running over the switched connection. This is for EWSD completely transparent. Registration of these data is task of the Packet Handler.

For a nailed-up connection, no IACAMA ticket is generated.

* **PM D-Channel (value 10)**

A D-channel service means that the Packet Data are running over the D-channel from the end user to the DLU. In the DLU (more precise, the SLMD inside the DLU), these D-channel Packet Data are multiplexed onto a 64 kbit channel. These 64 kbit channels are **always** nailed up connections, for which in general, **no IACAMA tickets are generated**. That implies that the service "PM D-channel" **will never turn up in an IACAMA ticket**.

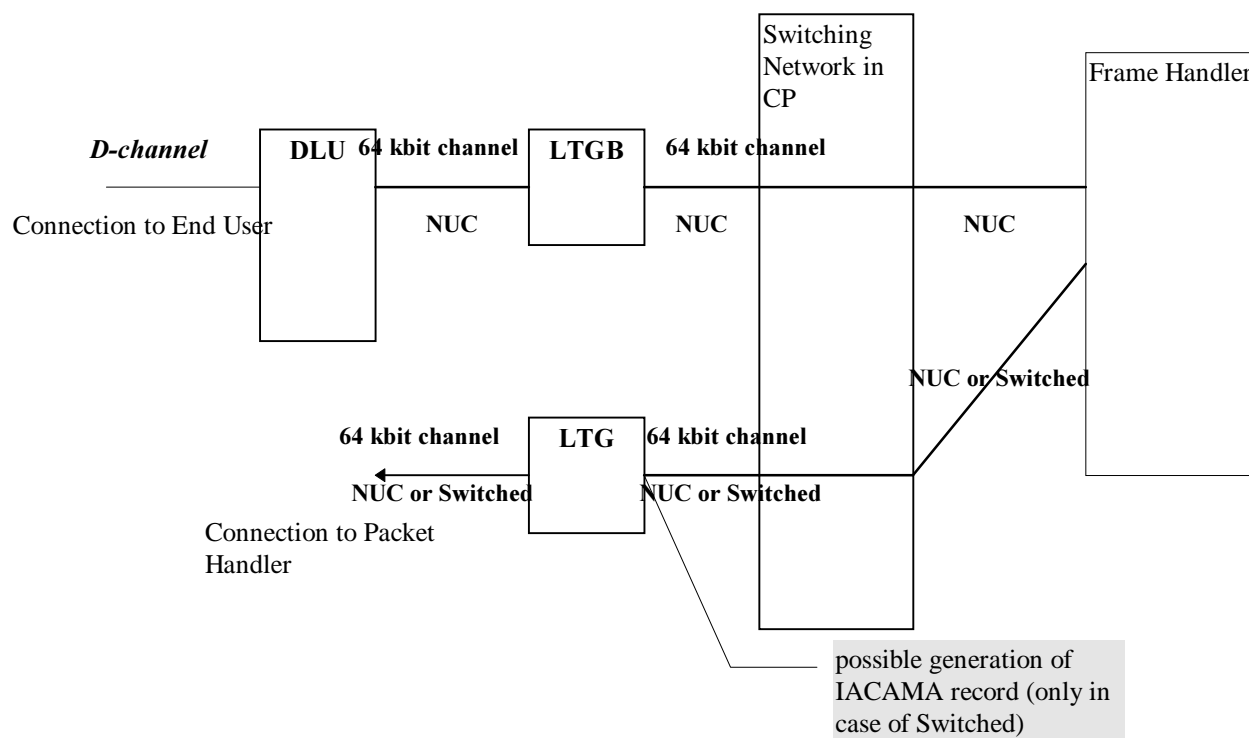


* **PM Bd-Channel (value 11)**

This value is used for the connection from Packet Handler to Frame Handler. A Frame Handler is a concentrator in EWSD, which decreases the number of channels between EWSD and Packet Handler.

A Frame Handler concentrates D-channel Packet Data onto a limited number of 64 kbit channels between the Frame Handler and the Packet Handler. These 64 kbit channels are called Bd-channels. They are either switched or nailed up. In case of switched, the Packet Handler always initiate the connection, that means the Frame Handler is always B-side from a Call Processing (set up of the switched connection) point of view.

For this connection, the service "Bd Channel" is used. It is theoretically possible to generate an IACAMA-ticket for this connection, but this IACAMA-ticket would not have any relevance for the end user.



C) Operating remarks

- This package is not administrable.
- The mapping between internal and administrative service can be administered with the command MOD CALLPOPT.

D) Postprocessing remarks

- This package has a fixed length of 4 bytes. The package length is **not** included in the package itself.

5.1.2.4 Package 103 : CHARGE UNITS FOR CONNECTION

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (103)								binary, fixed value 103 (H'67)
2	2	3	CALL CHARGE UNITS								binary

B) Functional description

This package contains the total amount of charge units known in the registering exchange that apply for the call. In case of intermediate charging, the charge units are counted per registered call part (i.e. the charge units are reset after every intermediate registration).

Remarks :

- When during the registration period the number of call charge units exceeds 65535, intermediate charging is performed.
- No differentiation is made between charge units which are generated in the own exchange and the charge units which are received from a higher exchange.

C) Operating remarks

- the inclusion of package 103 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = CHARGES;

D) Postprocessing remarks

- This package has a fixed length of 4 bytes. The package length is **not** included in the package itself.

5.1.2.5 Package 105 : TRUNK IDENTIFICATION INCOMING

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (105)								binary, fixed value 105 (H'69)
2	2	6	TRUNK GROUP NUMBER								char, max. 6
3	8	2	TRUNK NUMBER								binary, range 0 - 4095

B) Functional description

This package is only present when the A-side of the call is a trunk.

Remarks :

- the IACAMA-ticket generated on an incoming PA-slave connection is considered as a LAMA ticket, and hence will never contain the package 105.

C) Operating remarks

- When the command
ENTR CDTDAT : ALTNSD = RECTRCIC;
is executed, packages 107/108 instead of 105/106 are generated
- The Trunk Group Number corresponds to the parameter TGNO or TGN2 in the CR TGRP command. As default, TGNO is taken as Trunk Group Number. The use of TGN2 as Trunk Group Number is triggered by the command
ENTR CDTDAT : ALTNSD = TGN2;
- The Trunk Number corresponds to the parameter LNO in the CR TGRP command.

D) Postprocessing remarks

- This package has a fixed length of 9 bytes. The package length is **not** included in the package itself.

5.1.2.6 Package 106 : TRUNK IDENTIFICATION OUTGOING

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (106)								binary, fixed value 106 (H'6A)
2	2	6	TRUNK GROUP NUMBER								char, max. 6
3	8	2	TRUNK NUMBER								binary, range 0 - 4095

B) Functional description

This package is only present when the B-side of the call is a trunk (i.e. the call is not terminating in the own exchange, but routed to another one), or an announcement.

C) Operating remarks

- When the command
ENTR CDTDAT : ALTNSD = RECTRCIC;
is executed, packages 107/108 instead of 105/106 are generated
- The Trunk Group Number corresponds to the parameter TGNO or TGN2 in the CR TGRP command. As default, TGNO is taken as Trunk Group Number. The use of TGN2 as Trunk Group Number is triggered by the command
ENTR CDTDAT : ALTNSD = TGN2;
- The Trunk Number corresponds to the parameter LNO in the CR TGRP command.

D) Postprocessing remarks

- This package has a fixed length of 9 bytes. The package length is **not** included in the package itself.

5.1.2.7 Package 107: TRUNK IDENTIFICATION INCOMING (CIC FORMAT)

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (107)								binary, fixed value 107 (H'6B)
2	2	6	TRUNK GROUP NUMBER								char, max. 6
3	8	2	CIC PCM NUMBER								binary range 0-681 (DIU24) range 0-511 (DIU 30)
4	10	1	CIC CHANNEL NUMBER								binary range 0-24 range 0-31

B) Functional description

When administered by the crafts person, package 107 is generated under the same circumstances as described under package 105 (Trunk Identification Incoming), instead of this package, i.e. the package is only generated in case the A-side is a trunk.

The CIC format indicates that the trunk is identified with the number of the PCM (Pulse Code Modulation) trunk, and the number of the channel on this PCM trunk.

C) Operating remarks

- The inclusion of package 107 instead of package 105 in the IACAMA ticket is controlled via command ENTR CDTDAT : ALTNSD = RECTRKIC; which triggers at the same time, the inclusion of package 108 instead of 106.
- The Trunk Group Number corresponds to the parameter TGNO or TGN2 in the CR TGRP command. As default, TGNO is taken as Trunk Group Number. The use of TGN2 as Trunk Group Number is triggered by the command ENTR CDTDAT : ALTNSD = TGN2;

D) Postprocessing remarks

- This package has a fixed length of 10 bytes. The package length is **not** included in the package itself.

5.1.2.8 Package 108: TRUNK IDENTIFICATION OUTGOING (CIC FORMAT)

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (108)								binary, fixed value 108 (H'6C)
2	2	6	TRUNK GROUP NUMBER								char, max. 6
3	8	2	CIC PCM NUMBER								binary range 0-681 (DIU24) range 0-511 (DIU 30)
4	10	1	CIC CHANNEL NUMBER								binary range 0-24 range 0-31

B) Functional description

When administered by the crafts person, package 108 is generated under the same circumstances as described under package 106 (Trunk Identification Outgoing), instead of this package, i.e. the package is only generated in case the B-side is a trunk.

The CIC format indicates that the trunk is identified with the number of the PCM (Pulse Code Modulation) trunk, and the number of the channel on this PCM trunk.

C) Operating remarks

- The inclusion of package 108 instead of package 106 in the IACAMA ticket is controlled via command ENTR CDTDAT : ALTNSD = RECTRKIC; which triggers at the same time, the inclusion of package 107 instead of 105.
- The Trunk Group Number corresponds to the parameter TGNO or TGN2 in the CR TGRP command. As default, TGNO is taken as Trunk Group Number. The use of TGN2 as Trunk Group Number is triggered by the command ENTR CDTDAT : ALTNSD = TGN2;

D) Postprocessing remarks

- This package has a fixed length of 10 bytes. The package length is **not** included in the package itself.

5.1.2.9 Package 110 : CONNECTION IDENTIFICATION NUMBER

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (110)								binary, fixed value 110 (H'6E)
2	2	1	PACKAGE LENGTH								binary, fixed value 6 (H'06)
3	3	4	CONNECTION IDENTIFICATION								binary

B) Functional description

The Connection Id. is a unique identification for one call, that means :

- at **one point of time**, all **different calls** in one exchange have a **different** Connection Id.
- **all tickets for one call** (AMA tickets, MOB (Meter Observation) tickets, IACAMA tickets, IN AMA tickets, all intermediate tickets) get the **same** Connection Id., independent whether these tickets are generated on the **A- or B-side** of the call. (e.g. IACAMA tickets generated on the A-side of the call, and MOB terminating tickets, generated on the B-side of the same call have the same Connection Id.)
- it is guaranteed that the same Connection Id will not appear again after **at least 5 days**. This estimation is based on the highest possible traffic in EWSD. In case of normal traffic, the same Connection Id will appear again after a much longer period.

Remarks :

- In case of **call forwarding**, each leg A-B1 and B1-B2 is considered as a **separate call**, and hence gets a **different connection id**.
- The Connection Id is **not** signalled to other exchanges.

C) Operating remarks

- The inclusion of package 110 in the AMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = CONNID;
Remark that EXTNSD = CONNID is set by **default**, i.e. without entering the parameter EXTNSD = CONNID within the ENTR CDTDAT command, package 110 will be collected.
In case additionally the command
ENTR CDTDAT : EXTNSD = GCR;
is entered, **not package 110, but package 149 (Global Charge Reference) will be built**.

D) Postprocessing remarks

- This package has a fixed length of 6 bytes. The package length is however included in the package itself, to permit extensions of the Connection Id in later EWSD versions, without great impact on Postprocessing.

5.1.2.10 Package 117 : PROJECT SPECIFIC DATA**A) Syntax description**

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (117)								binary, fixed value 117 (H'75)
2	2	1	PACKAGE LENGTH								binary total length of package
3	3	n	DATA								binary, max 10 bytes

B) Functional description

This package is used to collect charging information for late features, i.e features which are implemented for one specific project for one specific EWSD version, after the common functionality of this EWSD version has been fixed. At most **10 bytes** of data can be contained in this package.

C) Operating remarks

- This package is **not** administrable. When the LTG (the peripheral EWSD processor) provides project specific charging data, these will be transparently taken over into the IACAMA ticket.

D) Postprocessing remarks

- The collection of this package is project specific, and thus, the package is collected only if agreed upon between the customer and Siemens. .
- The total package length (in byte) is located at the second byte of the package.

5.1.2.11 Package 119 : TRANSMISSION MEDIUM REQUIRED

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (119)								binary, fixed value 119 (H'77)
2	2	1	TRANSMISSION MEDIUM REQ								binary

- Field No 2 : TRANSMISSION MEDIUM REQUIRED

Value	Meaning
0	3.1 kHz Audio
1	Speech
2	64 kBit unrestricted
3	64 kBit unrestricted with fallback
4-255	reserved

B) Functional description

This package contains the requirements made by the originating exchange as far as the transmission medium quality is concerned.

When e.g. only 3.1kHz is required, it is still possible that a further exchange seizes an outgoing 64kBit trunk to forward the call, but this must not influence the accounting, because only a 3.1kHz was requested.

C) Operating remarks

- This package is not administrable.

D) Postprocessing remarks

- This package has a fixed length of 2 bytes. The package length is **not** included in the package itself.
- This package will always be present in case the transmission medium required is not 3.1kHz Audio, i.e. the package 119 will be present if and only if the transmission medium required is different from 3.1 KhZ Audio (value 0).

5.1.2.12 Package 120 : CATEGORY**A) Syntax description**

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (120)								binary, fixed value 120 (H'78)
2	2	1	CATEGORY								refer to description

Description of Field No 2 : CATEGORY

Value	Meaning
0	undefined
1	subscriber
2	priority subscriber
3	test equipment
4	coin box
5	operator national
6	data transmission national
7	subscriber or operator international w/o forward transfer
8	data transmission international
9	priority subscriber international
10	operator international with forward transfer
11-15	reserved for project specific applications

B) Functional description

This package contains detailed information about the kind of subscriber at the originating side of the call.

Remarks :

- This information is available in the originating exchange, and in transit exchanges which have been reached via MFC and ISUP trunks

C) Operating remarks

- This package is not administrable.

D) Postprocessing remarks

- This package has a fixed length of 2 bytes. The package length is **not** included in the package itself.
- This package will always be present in case the originating side is not a normal subscriber, i.e. the package 120 will be present if and only if the category is different from 1 (and 0, which is undefined).

5.1.2.13 Package 128 : NEW DESTINATION

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (128)								binary, fixed value 128 (H'80)
2	2	1	reserved				NO OF DIGITS				binary, range (0:24)
3	3	n	DIGIT(n - i)				DIGIT (n-i+1)				packed digits, values (0:H'F)

B) Functional description

This package contains the destination digits as returned from the SCP (Service Control Point) in case of an IN-call.

Remarks :

- The New Destination Digits can **change during the call**, because one IN call can consist of several call parts to different destinations (e.g. connection to different announcements before connection to the desired destination, follow on (new call is set up to another destination without the calling line going on hook), rerouting when the destination cannot be reached). In each (intermediate) ticket, the **actual** new destination digits are stored.
- Since the new destination digits cannot be signaled over trunk, the package can only be present in the IACAMA ticket, when the SSP (Service Switching Point, i.e. the switch which has the access to the SCP) is equal to the **registering exchange**.

C) Operating remarks

- The inclusion of package 128 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = NEWDEST;

D) Postprocessing remarks

- The package length is **not** included in the package itself. The package length has to be calculated in the following way :
package length in bytes = 2 + [integral part of the division ((NO_OF_DIGITS + 1) / 2)]

5.1.2.14 Package 130 : TRAFFIC QUALITY DATA

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (130)								binary, fixed value 130 (H'82)
2	2	1	PACKAGE LENGTH								binary; total length of package
3	3	2	CAUSE VALUE								binary; refer to description
4	5	1	res	CODING STAND.		res	LOCATION				binary; refer to description

Description of Field No 3 : CAUSE VALUE

- The values 0-127 contain the possible CAUSE values according to ITU-T (Q.850), on condition the Coding Standard is equal 0, indicating ITU Coding Standard
- The values greater 127 have a project specific meaning

Value	Meaning
1	Unallocated (unassigned) number
2	No route to specific transit network
3	No route to destination
4	Send special information tone
5	Misdialed trunk prefix
6	Channel unacceptable
7	Call awarded and being delivered in an established channel
8	Preemption
9	Preemption - circuit reserved for reuse
16	Normal call clearing
17	User busy
18	No user responding
19	No answer from user (user alerted)
20	Subscriber absent
21	Call rejected
22	Number changed
26	Non-selected user clearing
27	Destination out of order
28	Invalid number format
29	Facility rejected
30	Response to STATUS ENQUIRY
31	Normal, unspecified
34	No circuit/channel available
38	Network out of order
39	Permanent frame mode connection out-of-service
40	Permanent frame mode connection operational
41	Temporary failure
42	Switching equipment congestion
43	Access information discarded
44	Requested circuit/channel not available
46	Precedence call blocked
47	Resources unavailable, unspecified
49	Quality of service unavailable
50	Request facility not subscribed
53	Outgoing calls barred within CUG
55	Incoming calls barred within CUG

Value	Meaning
57	Bearer capability not authorized
58	Bearer capability not presently available
62	Inconsistency in designated outgoing access information, and subscriber class
63	Service or option not available, unspecified
65	Bearer capability not implemented
66	Channel type not implemented
69	Requested facility not implemented
70	Only restricted digital information bearer capability is available
79	Service or option not implemented, unspecified
81	Invalid call reference value
82	Identified channel does not exist
83	A suspended call exists, but this call identity does not
84	Call identity in use
85	No call suspended
86	Call having the requested call identity has been cleared
87	User not member of CUG
88	Incompatible destination
90	Non-existing CUG
91	Invalid transit network selection
95	Invalid message, unspecified
96	Mandatory information element is missing
97	Message type non-existing or not implemented
98	Message not compatible with call state or message type non-existent or not implemented
99	Information element non-existent or not implemented
100	Invalid information element contents
101	Message not compatible with call state
102	Recovery on timer expiry
103	Parameter non-existent or not implemented - passed on
110	Message with unrecognized parameter discarded
111	Protocol error, unspecified
127	Interworking, unspecified

Description of Field No 4 :

LOCATION : This field contains the possible location values according to ITU-T (Q.931)

Value	Meaning
0	User
1	Private network serving the local user
2	Public network serving the local user
3	Transit network
4	Public network serving the remote user
5	Private network serving the remote user
6	reserve
7	International network
8-9	reserve
10	Network beyond interworking point
11-15	reserve

CODING STANDARD :

Value	Meaning
0	ITU standard
1	reserved international standard
2	national standard
3	standard specific to identified location

B) Functional description

Cause, Location and Coding Standard are standardized notions, described in /7/. Cause and Location describe the location and the reason why the call is released; the Coding Standard indicates the Standard which is used for the coding of the Cause (when the Cause is generated in EWSD and no project specific changes regarding the Cause/Location are implemented, the Coding Standard is equal to 0).

Remarks :

- The Cause cannot be used to distinct between answered and unanswered calls : a normal call release by the originator of the call is indicated by Cause = 16 (Normal Call Clearing), independent whether the answer signal has been received or not.

C) Operating remarks

- The inclusion of package 130 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = TRAFQUAL;
Remark that EXTNSD = TRAFQUAL is set by **default**, i.e. without entering the parameter EXTNSD = TRAFQUAL within the ENTR CDTDAT command, package 130 will be collected.

D) Postprocessing remarks

- This package has a fixed length of 5. The package length is included on byte 2 in the package itself.
- Package 130 is only provided at the **end of the call**, i.e. in the **single or last** ticket of the call

5.1.2.15 Package 134 : DURATIONS BEFORE ANSWER

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (134)								binary, fixed value 134 (H'86)
2	2	1	PACKAGE LENGTH								binary, fixed value 6
3			DURATIONS BEFORE ANSWER								
	3	2	duration from dialtone till address complete								binary
	5	2	duration from address complete till answer								binary

B) Functional description

Durations Before Answer has two components :

- the time interval **between seizure and receipt of the address complete** signal from the B-side. When however the call is disconnected before address complete has been received, the time interval between seizure and release of the call is registered.
- the time interval between the **receipt of address complete signal and receipt of the answer signal** from the B-side. When the call is disconnected after address complete and before answer is received, the time interval between address complete and release of the call is registered.

Remark :

- for IACAMA tickets generated in the originating exchange, the time of seizure is equal to the time when the A-party receives dialtone.

C) Operating remarks

- The inclusion of package 134 in the IACAMA ticket is not administrable.

D) Postprocessing remarks

- Package 134 is present in tickets with Record Sequence = single or first intermediate.
- The value Duration from address complete till answer being zero indicates that the address complete signal has not been received.
- This package has a fixed length of 6 bytes. The package length is included on byte 2 in the package itself.

5.1.2.16 Package 138 : B METER PULSES**A) Syntax description**

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (138)								binary, fixed value 138 (H'8A)
2	2	1	PACKAGE LENGTH								binary, fixed value 6
3	3	2	B METER PULSES								binary,

B) Functional description

This package contains the total amount of charge units which are received from a higher exchange. In case of intermediate charging, the charge units are counted per registered call part (i.e. the charge units are reset after every intermediate registration).

The package enables an inter operator accounting based on charge units with the operator in the higher exchange.

C) Operating remarks

- the inclusion of package 138 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = BMETPULS;

D) Postprocessing remarks

- This package has a fixed length of 4 bytes. The package length is **not** included in the package itself.

5.1.2.17 Package 142 : CALLING PARTY NUMBER

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (142)								binary; fixed value 142 (H'8E)
2	2	1	PACKAGE LENGTH								binary; total length of package
3	3	1	res	NADI (of calling party)							binary; refer to description
4	4	1	res	NPI (calling)		reserved					binary; refer to description
5	5	1	reserved			NO OF DIGITS					binary; range 0 - 24
6	6	n	DIGITS (n - i)				DIGITS (n - i + 1)				packed digits, values (0:H'F)

Description of Field No 3 : NADI (Nature of Address Indicator) of calling party (according to /8/)

Value	Meaning
0	unknown interworking
1	subscriber number
2	unknown
3	significant national number
4	international number
5-112	spare
113	subscriber number operator request
114	national number operator request
115	international number operator request
116	no number present operator request
117	no number present cut through
118	950 from public station
119	test line
120-127	spare

Description of Field No 4 : NPI (of calling party)

Value	Meaning
0	unknown
1	ISDN numbering plan
2	spare
3	data numbering plan
4	telex numbering plan
5-7	reserved

B) Functional description

When this package is activated by the crafts person, it will be present in the following two cases :

- when the IACAMA ticket is generated in the originating exchange and the call has been made by an ISDN member of a PBX with Direct Dialling In capability. In this case, the package contains the number which results from the screening of the signalled calling party number (i.e. the number which is provided in the D-channel SETUP message as calling line number).

The screening of the signalled calling party leads to following contents of package 142 :

- * in case the PBX has the no-screening option, package 142 will contain a default number. This default number is either the Pilot Directory Number, or the Operator Number, in case this number is provided.

- * in case the PBX has the screening option, the screening can be successful or unsuccessful. Successful screening means that the signalled calling party has been identified as being DDI digits, or as PDN without LAC + DDI digits or as PDN with LAC + DDI digits. The number provided in package 142 will be a national significant number, i.e. PDN inclusive LAC + DDI digits. In case the screening is not successful, package 142 is provided with the default number as is the case when the PBX has the no-screening option.
- in IACAMA tickets in a transit or terminating exchange, this package contains the inter-office signalled calling party if present (in case of ISUP, this is the calling party number as provided in the IAM message, or in case of call diversion, the redirecting number).

Remarks :

- The **national or international prefix digits** (e.g. "0" or "00") are never present in package 142.
- In case of IACAMA tickets generated in a transit or terminating exchange, the digits contained in package 142 are **equal to the record owner digits** in the Fixed Part of the IACAMA-ticket. Package 142 brings however additional information (NADI and NPI).

The following figure illustrates the relation between the Record Owner and package 142 in the IACAMA ticket in the originating exchange, and package 142 in the IACAMA ticket in the transit exchange.

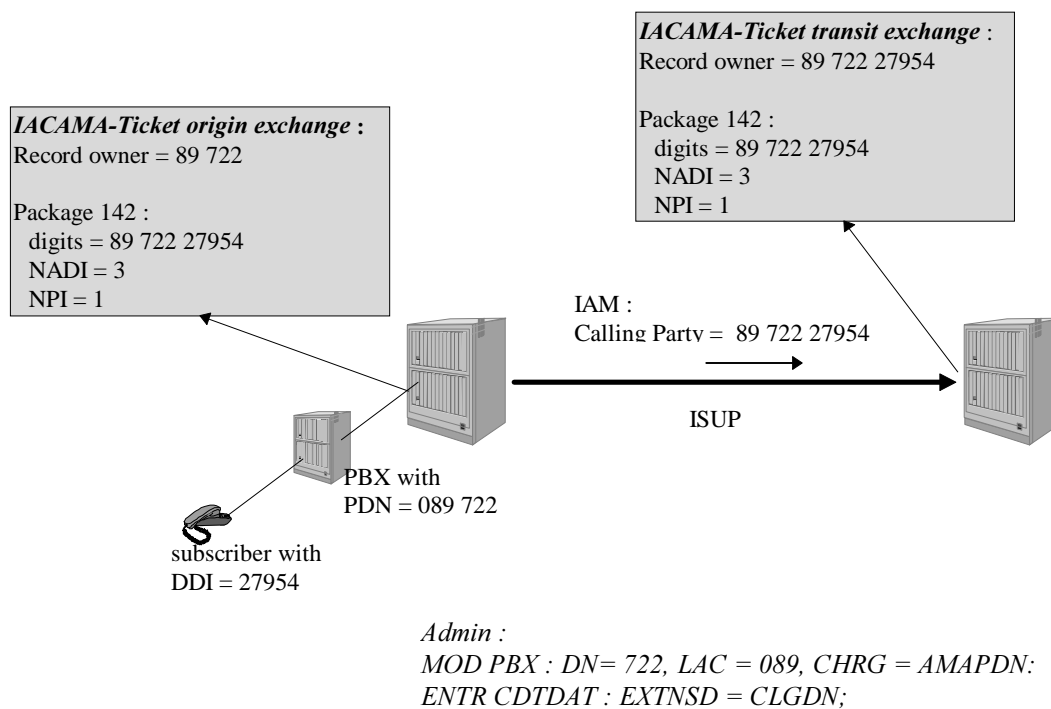


Figure 8 : Provision of package 142 (Calling Party Number)

C) Operating remarks

- The inclusion of package 142 in the IACAMA ticket is controlled via command ENTR CDTDAT : EXTNSD = CLGDN;

- The screening/no screening option is administered via the parameter NOSCCCLIP in the MOD PBX command (**not** in the CR PBX command)
- The operator number is administered via the parameter OPN in the command CR PBX. The operator number is composed out of the PDN + further digits.

D) Postprocessing remarks

- The total package length (in byte) is located at the second byte of the package.

5.1.2.18 Package 145 : CAC NUMBER

A) Syntax description

Field No	Byte Pos	Length	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (145)								binary, fixed value 145 (H'91)
2	2	1	PACKAGE LENGTH								binary, total length of package
3	3	1	CAC TYPE		PREFIX		NO OF DIG			refer to description	
4	4	n	DIGITS (n - i)				DIGITS (n -i+1)				packed digits, value (0:H'F), including prefix digits

Description of Field No 3 :

NUMBER OF CAC DIGITS : range between 0 - 7
Total length of the CAC (with prefix)

NUMBER OF PREFIX DIGITS : range between 0 - 3

CAC TYPE :

Value	Meaning
0	CAC type not available
1	network selected
2	subscriber default
3	subscriber dialed, used
4	subscriber dialed, not used
5	provided and not used

B) Functional description

A CAC (Carrier Access Code) is used to reach the destination via a specific carrier, i.e. a network of another operator. It can be explicitly dialed by the subscriber, or a value from the EWSD database can be used. The field CAC TYPE describes which kind of CAC has been used.

In the EWSD database, a CAC can be administered at following locations :

- at subscriber/PBX level
- at incoming trunk group level
- at exchange level
- as part of a Centrex code point
- as part of new digits during digit conversion (routing digits are converted to other digits, and these new digits contain a CAC)

Explanation of the different CAC types :

- **not available**

This value is set up when a CAC is available, but it cannot be determined where this CAC comes from. This value will only turn up in tickets, which are generated on an incoming ISUP trunk, on condition this trunk belongs to a trunk group without a CAC assigned to it.

Reason : ISUP transports the CAC in a separate field in the Initial Address Message, but cannot transport the type of CAC.

- **network selected (and used)**

This value is provided when the subscriber has not dialled a CAC, a CAC is necessary for routing and one of the following conditions are fulfilled :

- * the call arrives over a trunk which belongs to a trunk group with a CAC assigned to it.
- * routing digits are converted to new digits and these new digits contain a CAC
- * the CAC which was used is the network default one, i.e. the value which has been administered for the whole exchange.

- **subscriber default (and used)**

This value is provided when the subscriber has not dialled a CAC, a CAC is necessary for routing and one of the following conditions are fulfilled :

- * the subscriber has a default CAC assigned
- * a Centrex subscriber dials a digit string which is defined as a Centrex code point, with a CAC included in it

- **subscriber dialled, used**

This value is provided when the subscriber has dialled a CAC, and this CAC is also necessary for routing.

- **subscriber dialled, not used**

This value is provided when the subscriber has dialled a CAC, but this CAC is not necessary for routing. This is only the case when the routing digits, which include the dialled CAC, are first converted to a new digit string without CAC.

- **provided and not used**

This value is provided when the subscriber has not dialled a CAC, the call arrives over a trunk which belongs to a trunk group with a CAC assigned to it, but the CAC is not used for routing.

A CAC can be composed out of a Carrier Prefix and a Carrier Code. When all CAC's in the own network start with the same digits, these digits can be defined as a Carrier Prefix. This Carrier Prefix can get a project specific treatment (e.g. cut off before sending over ISUP).

Package 145 will only be present when following two conditions are fulfilled :

- the collection of the package has been activated by the crafts person via ENTR CDTDAT
- the CAC has been explicitly dialled by the subscriber,
or
the CAC has not been dialled but a value out of the EWSD database is used, because a CAC was necessary for the routing. Whether a CAC is necessary for routing or not, is administered via the command CR CPT : CODE = ..., DEST = ..., EVALDCAR = Y;
The own CAC, i.e. the CAC which has been administered as the CAC of the own network operator, is however never used for routing. That means, the parameter EVALDCAR is **not considered when the CAC to be used is the own CAC, and hence, the own CAC will never appear in the ticket.**

C) Operating remarks

- The inclusion of package 145 in the IACAMA ticket is controlled via command ENTR CDTDAT : EXTNSD = CAC;
Remark that EXTNSD = CAC is set by **default**, i.e. without entering the parameter EXTNSD = CAC within the ENTR CDTDAT command, package 145 will be collected.
- Carrier Access Code are administered in the switch via the command CR CAC : CAC = ..., CARRIER = ... ;
- The Carrier Prefix is administered via the command ENTR DNATT : PFXCAC = <CAC prefix>;
- Making a CAC necessary for routing is administered via CR CPT : CODE = <code>, DEST = <destination>, EVALDCAR = Y;
When the CAC to be used is however the own CAC, the parameter EVALDCAR is **not considered!**

- The own CAC is administered via
CR CAC : CAC = <carrier code>, CARRIER = <carrier name>, OWNCAC = Y;
- The network default CAC is administered via
CR CAC : CAC = <carrier code>, CARRIER = <carrier name>, EXCCAC = Y;
- A CAC is assigned to a subscriber/PBX via
CR SUB/PBX/PBXLN CAC : DN = <directory number>, LAC = <local area code>, COSDAT = CAC;
- Conversion of the routing digits is done via the parameters NEWCODE or CONVCODE in the command
CR DEST : DEST = <destination>, CONVCODE = <converted digits>;
CR DEST : DEST = <destination>, NEWCODE = <converted digits>;

D) Postprocessing remarks

- The total package length (in bytes) is located at the second byte of the package.

5.1.2.19 Package 146 : THIRD PARTY NUMBER

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (146)								binary; fixed value 146 (H'92)
2	2	1	PACKAGE LENGTH								binary; total length of package
3	3	1	reserved				3rd PARTY TYPE				binary; refer to description
4	4	1			F6	F5	F4	F3	F2	F1	flags; refer to description
5	5	1	NO OF DIGITS								binary; range (0:31)
6	6	n	DIGITS (n - i)				DIGITS (n - i+1)				packed digits, value (0:H'F)

Description of Field No 3 : THIRD PARTY TYPE

Value	Meaning
0	original calling party in forwarded call
1	call transfer partner
2	Remote SCI calling party
3 - 15	reserved

Description of Field No 4 : Flags

This field is only relevant in case the "third party type" indicates "call transfer partner". In this case, the flags are the Call Indicators of the other leg in the transferred call (compare with the Call Indicators for the own leg in package 154).

Flag	Meaning
F1	Local intercom call
F2	Remote intercom call
F3	Intra charging site call
F4	Inter charging site call
F5	Public call
F6	Public business call

B) Functional description

When activated by the crafts person, this package is collected in following cases :

- In case of call diversion (redirection). Package 146 is present in the record for the redirecting party and contains the DN of the original calling party (resp. the DN of the previous party in case of MFC-signalling). The flags F1 till F6 are not relevant.

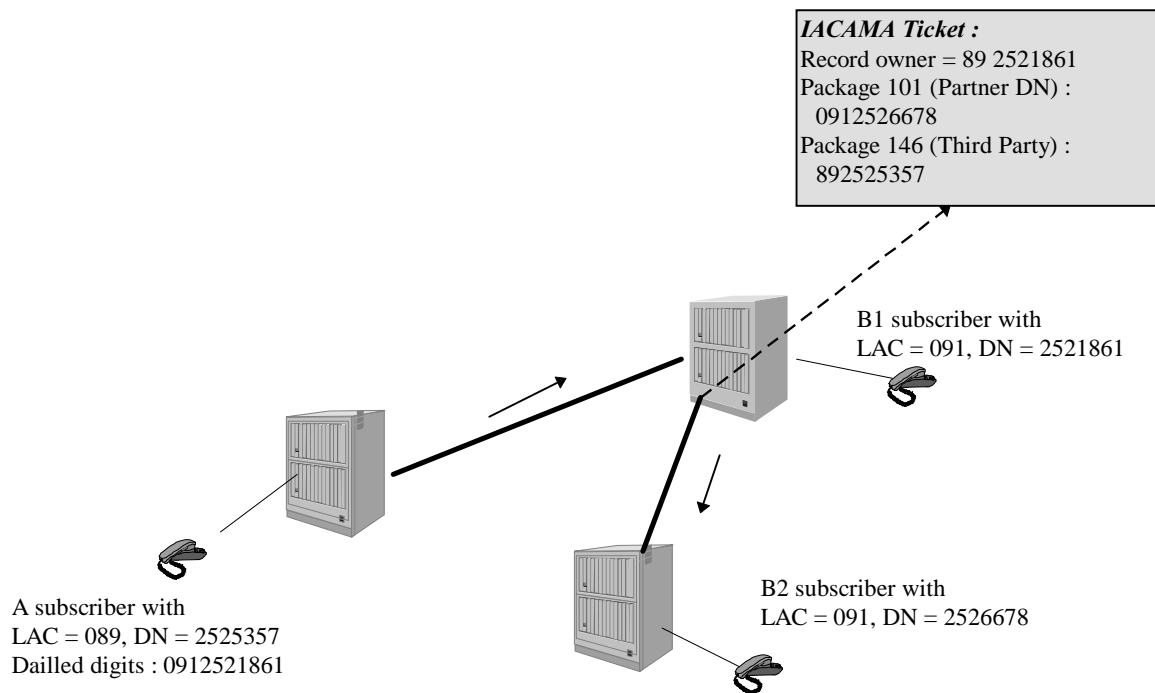


Figure 9 : Third Party Number in case of Call Diversion

- In case of Call Transfer, package 146 is present in the tickets generated for the controller who transfers the call, and contains the DN of the other partner in the transferred call. The flags F1 till F6 will contain additional call indicators with information about the type of CTX call of the other call leg.

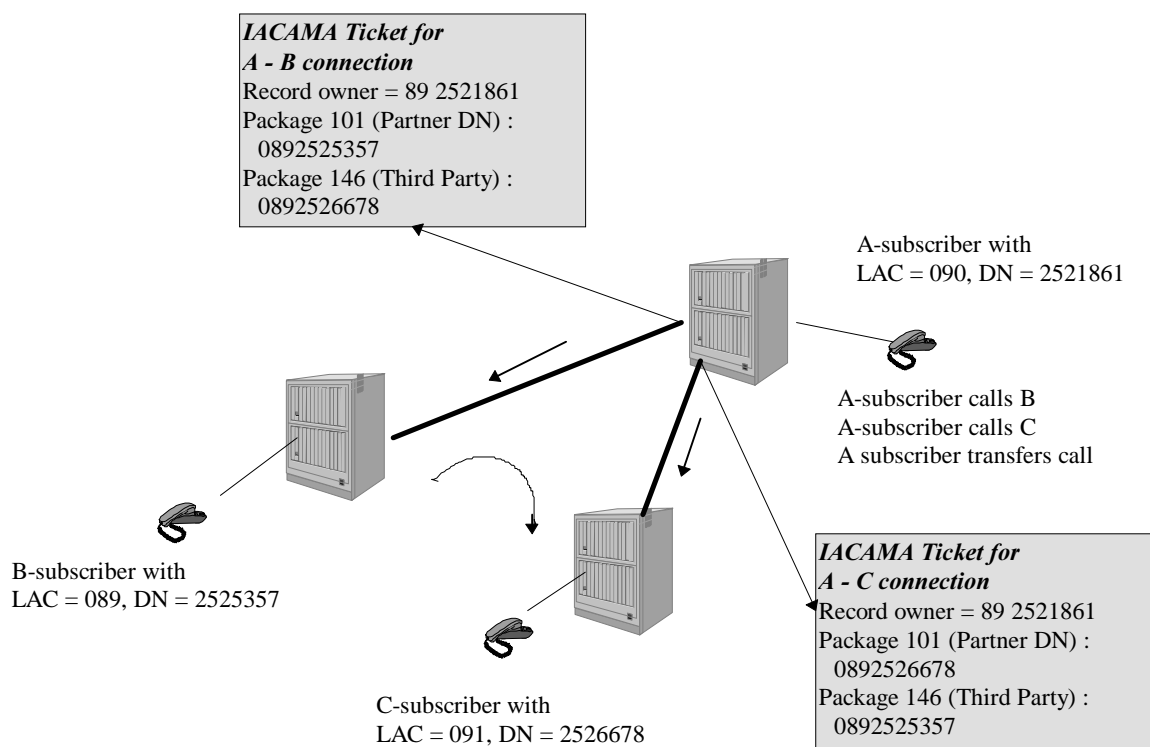


Figure 10 : Third Party Number in case of A-side Call Transfer

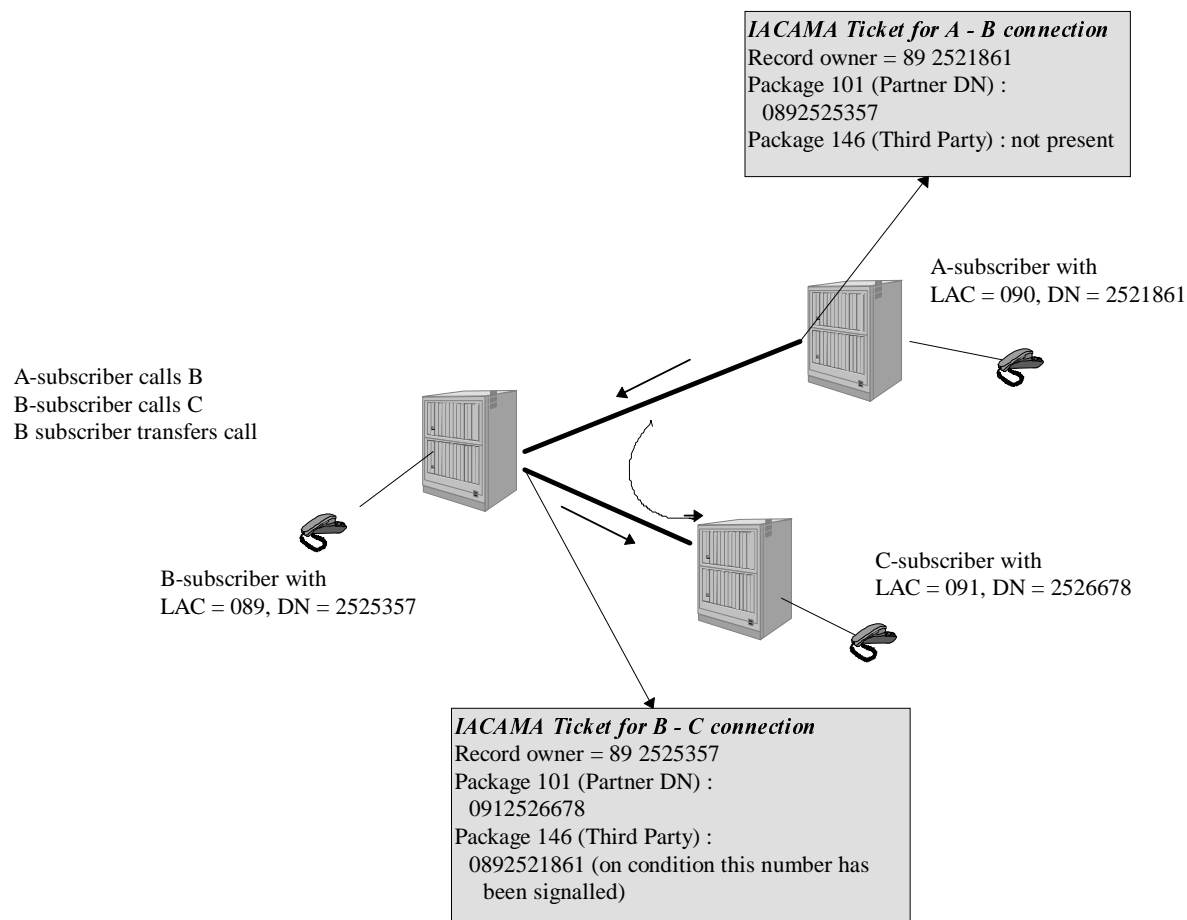


Figure 11 : Third Party Number in case of B-side transfer

Remarks :

- In case the ticket is generated for a Centrex subscriber, package 146 contains always a **public DN**, not an intercom number.

C) Operating remarks

- The inclusion of package 146 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = THRDPTY;

D) Postprocessing remarks

- The total package length (in byte) is located at the second byte of the package.

5.1.2.20 Package 147 : TARIFF INFO

A) Syntax description

Package 147 (Tariff Info) is composed out of a fixed part, followed by one or more optional subpackages. The presence of the subpackages is indicated by validity indicators. The subpackages themselves have a variable structure with a fixed part and one or more subpackages.

In the subpackages, the byte position is indicated with a "+", indicating that this byte position is relative.

Fixed Part :

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (147)								binary; fixed value 147 (H'93)
2	2	1	PACKAGE LENGTH								binary; total length of package
3	3	1	CONTROL & VALIDITY INDICATORS								flags

The Control & Validity indicators have the following meaning :

Bit	Value	Meaning
0	0	Current Tariff not present
	1	Current Tariff present
1	0	Next Tariff not present
	1	Next Tariff present
2	0	Operator Information not present
	1	Operator information present
3		reserved
4		reserved
5		reserved
6	0	Tariffs only for display and supervision
	1	Tariffs for charging
7	0	Pulse Format
	1	Currency Format

Bits 0, 1 and 2 are validity indicators : they indicate whether the optional subpackages for Current Tariff, Next Tariff, and Operator Information are present or not. Remark that the subpackages in the ticket appear in the same order as the validity indicators (the order of the validity indicators is according to their bit position). Example : when bit 0 (Current Tariff) and bit 2 (Operator Information) are set, first the Current Tariff subpackage will appear in the ticket, followed by the Operator Information subpackage.

There are two layouts for the Current Tariff subpackage and the Next Tariff subpackage : a layout for tariff information in pulse format, and a layout for tariff information in currency format. Which layout is taken is indicated by bit 7. Remark that either the Current Tariff and Next Tariff subpackage, when they are present, are both in pulse format or both in currency format; a mixture is not allowed.

Bit 6 indicates whether the tariff is to be used for charging or only for display. The latter is e.g. the case, when the charges are registered in a transit exchange, and the tariff information in the originating exchange must only be used for Advice of Charge.

1. Subpackage Current Tariff in Pulse Format :

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	1	reserved			hours					Time (24 h) indicator when the current tariff was received
+2	1	reserved			minutes					
+3	1	reserved			seconds					
+4	1	Acknowledgment status								refer to description
+5	1	Tariff Control & Validity Indicators								refer to description

The Acknowledgment status has the following meaning :

Value	Meaning
0	Waiting for acknowledgment
1	Positively acknowledged
2	Negatively acknowledged
3	Received but not sent

The Tariff Control & Validity Indicators have the following meaning :

Bit	Value	Meaning
0	0	Call Attempt Charges not present
	1	Call Attempt Charges present
1	0	Call Setup Charges not present
	1	Call Setup Charges present
2	0	Communication Charge #1 not present
	1	Communication Charge #1 present
3	0	Communication Charge #2 not present
	1	Communication Charge #2 present
4	0	Communication Charge #3 not present
	1	Communication Charge #3 present
5	0	Communication Charge #4 not present
	1	Communication Charge #4 present
6	0	reserved
	1	
7	0	Cyclic communication charge sequence
	1	Non-cyclic sequence

1.a) Subpackage Call Attempt Charges of Subpackage Current Tariff

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	1	Call Attempt Charge Pulses								numeric pulse count

1.b) Subpackage Call Setup Charges of Subpackage Current Tariff

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	1	Call Setup Charge Pulses								numeric pulse count

1.c) Subpackage Communication Charge #1 of Subpackage Current Tariff

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	1	Pulse Units								numeric
+2	2	Charge Unit Time Interval								refer to description
+4	2	Count Duration								refer to description

Charge Unit Time Interval :

Binary coded with a range from 0 to 35997 . It begins with 200 milliseconds and continues with steps of 50 milliseconds.

Examples :

Charge Unit Time Interval	Meaning
0	no periodic metering
1	200 msec
2	250 msec
...	
35997	30 minutes

Count Duration :

Indicates for how long the communication charge component is valid .

Examples :

Count Duration	Meaning
0	unlimited
1	1 second
2	2 seconds
...	
3600	1 hour

1.d) Subpackage Communication Charge #2 of Subpackage Current Tariff

This subpackage has the same structure as the previous one (Communication Charge #1). The presence of this subpackage is only possible when the subpackage for Communication Charge #1 is present.

1.e) Subpackage Communication Charge #3 of Subpackage Current Tariff

This subpackage has the same structure as the subpackage for Communication Charge #1. The presence of this subpackage is only possible when the subpackages for Communication Charge #1 and Communication Charge #2 are present.

1.f) Subpackage Communication Charge #4 of Subpackage Current Tariff

This subpackage has the same structure as the subpackage for Communication Charge #1. The presence of this subpackage is only possible when the subpackages for Communication Charge #1, Communication Charge #2 and Communication Charge #3 are present.

2. Subpackage Next Tariff in Pulse Format :

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	1	reserved			hours					Time (24 h) indicator when the current tariff was received
+2	1	reserved		minutes						
+3	1	reserved		seconds						
+4	1	Acknowledgment status								refer to description above
+5	1	Switch over time								refer to description
+6	1	Tariff Control & Validity Indicators								refer to description above

The fields hours, minutes, seconds, Acknowledgment status and Tariff Control & Validity Indicators have the same structure and meaning as for the Current Tariff in Pulse Format.

Switch over time

This time is the absolute local time at which the next tariff has to become active. It is represented in steps of 15 minutes. The coding is the following :

Value	Meaning
0	spare
1	0 hour 15 minutes
2	0 hour 30 minutes
3	0 hour 45 minutes
4	1 hour 0 minutes
.	
.	
96	24 hour 0 minutes
97 - 255	spare

The same subpackages (Call Attempt, Call Setup, Communication Charges, ...) can be present as in the case for Current Tariff

3. Subpackage Current Tariff in Currency Format :

The package has the same structure as the package for the Current Tariff in Pulse Format; only the subpackages for Call Attempt, Call Setup and Communication Charges are different.

The subpackages for Call Attempt and Call Setup Charges have the following structure

3.a) Subpackage Call Attempt and Call Setup Charge

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	1	Scale								
+2	3	Factor								

The subpackages for Communication Charges have the following structure

3.b) Subpackage Communication Charge

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	1	Scale								integer between -7 and 3
+2	3	Factor								
+5	2	Count Duration								

Scale :

The actual value of the currency Scale is given by 10x, where x is the value stored in the package. Remark that the negative values (-7 till -1) are coded in one byte as a complement of 256.

Examples :

Scale	Value	Meaning
-7	249	0.0000001
-6	250	0.000001
-5	251	0.00001
-4	252	0.0001
-3	253	0.001
-2	254	0.01
-1	255	0.1
0	0	1
1	1	10
2	2	100
3	3	1000

Factor :

Integer between 0 and 999999; 0 indicates "no charge".

4. Subpackage Next Tariff in Currency Format :

The package has the same structure as the package for the Next Tariff in Pulse Format; only the subpackages for Call Attempt, Call Setup and Communication Charges are different. See previous package.

5. Subpackage Operator Data :

Byte Pos	Length	Field Name								Remarks
		7	6	5	4	3	2	1	0	
+1	12	OPERATOR DATA								octet strings; not further specified

B) Functional description

This package contains the tariff informations as received from a higher exchange (an exchange in the call path from the actual exchange up to the destination), or as sent to the lower exchange (an exchange in the call path from the actual exchange down to the origin).

Tariff information can be expressed in **pulse or in currency format**.

A tariff info in **pulse format** indicates a number of pulses and a time interval after which those pulses have to be counted. Example : 1 pulse each 10 seconds. The time interval can be omitted; in this case only a number of pulses have to be applied.

A tariff info in **currency format** gives directly a price per second (except for Call Attempt and Call Setup Charges where it is a time independent price). The price is expressed as a scale and a factor. Example : a price of 0.003 is expressed with a factor of 3 and a scale of -3.

One Tariff can have following **components** :

- **Call Attempt charges**

These charges apply only for unsuccessful calls and as soon as a call attempt is detected. Call Attempt charges are not time dependent. In pulse format, they are expressed as a number of pulses without time interval, In currency format, they are expressed with a scale and factor, but the price is in this case time independent.

- **Call Setup charges**

These charges apply only for successful calls at the moment the Answer signal is received. Call Setup charges are not time dependent. In pulse format, they are expressed as a number of pulses without time interval, In currency format, they are expressed with a scale and factor, but the price is in this case time independent.

- **1 to 4 Communication Charges**

These charges apply during conversation, i.e. after Answer. For each Communication Charge, a duration can be specified, which indicates how long one Communication Charge has to be applied.

Example :

the first 10 minutes after Answer, a tariff of 1 pulse per 10 seconds has to be applied; after these 10 minutes, a tariff of 1 pulse per 20 seconds has to be applied.

In this case, two Communication Charges are defined (pulse format is assumed) :

- * one with a Count Duration of 600 seconds, Pulse Units = 1 and Charge Unit Time Interval = 197 (steps of 50 milliseconds)
- * a second one with a Count Duration of 0 (unlimited), Pulse Units = 1 and Charge unit Time Interval = 397 (steps of 50 milliseconds)

Inside EWSD, an **active and a passive** tariff are maintained. The active tariff is the one which is currently being applied for the call. The passive tariff is the one which will become active at a future point of time (e.g. at 6.00 p.m.). The presence of the passive tariff allows an immediate switch over (switch from active to passive tariff) during the call.

The Tariff Info package can contain a **Current** and/or a **Next Tariff**. When received before Answer, the Current Tariff is the one which will become active at the moment Answer is received; the Next Tariff will be kept in EWSD as the passive one. The Next Tariff contains always the absolute Switch Over Time, i.e. the time at which the Next Tariff has to become active or has become active (see further for the latter case). When received during the call, the received Current Tariff will become immediately the active tariff.

The presence of a Tariff Info package with Current and Next Tariff does **not imply that at the moment the ticket is generated, the Current Tariff was the active one.**

Reason : at tariff switch over time, **intermediate charging is not necessarily** done (for intermediate charging, see further). This is possible because the switch over time is an absolute time indication. It is therefore very well possible, that the Next Tariff in the ticket was the **active one inside EWSD at the moment the ticket was generated.** This is illustrated in the following figure.

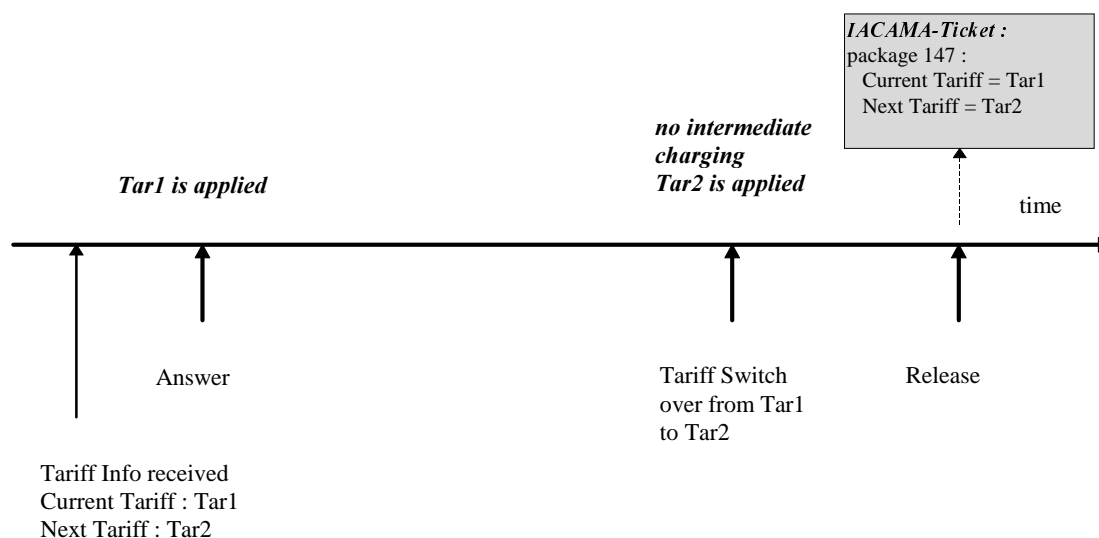
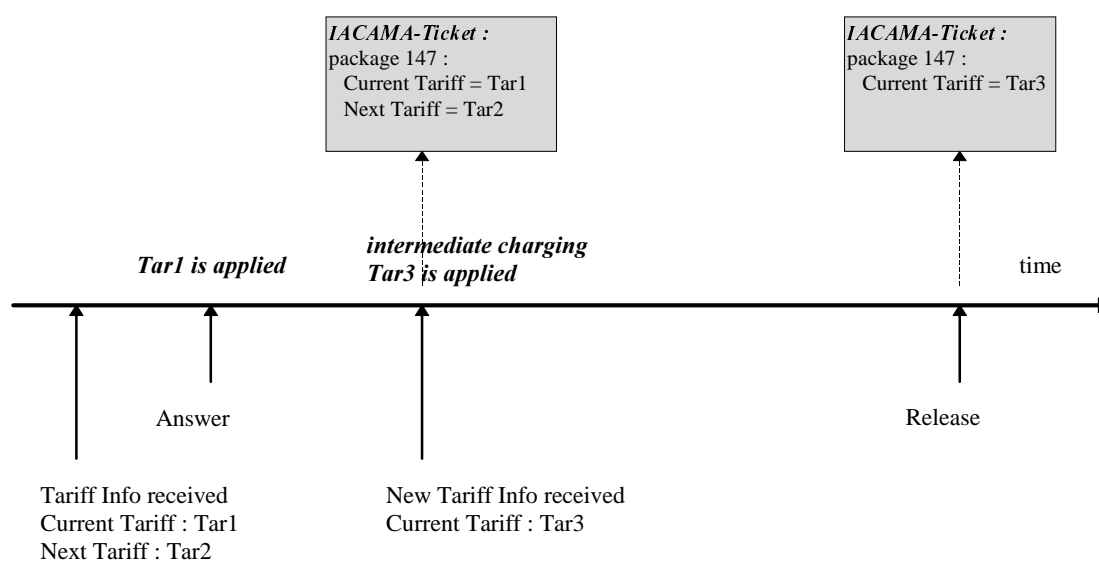


Figure 12 : Receipt of Tariff Info before Answer (call duration < intermediate timer)

For **intermediate charging** at receipt of Tariff Information and at tariff switch over time, following algorithm applies :

intermediate charging is done when old tariff information which is not yet registered in tickets is overwritten with new information.

This is illustrated in following figures :



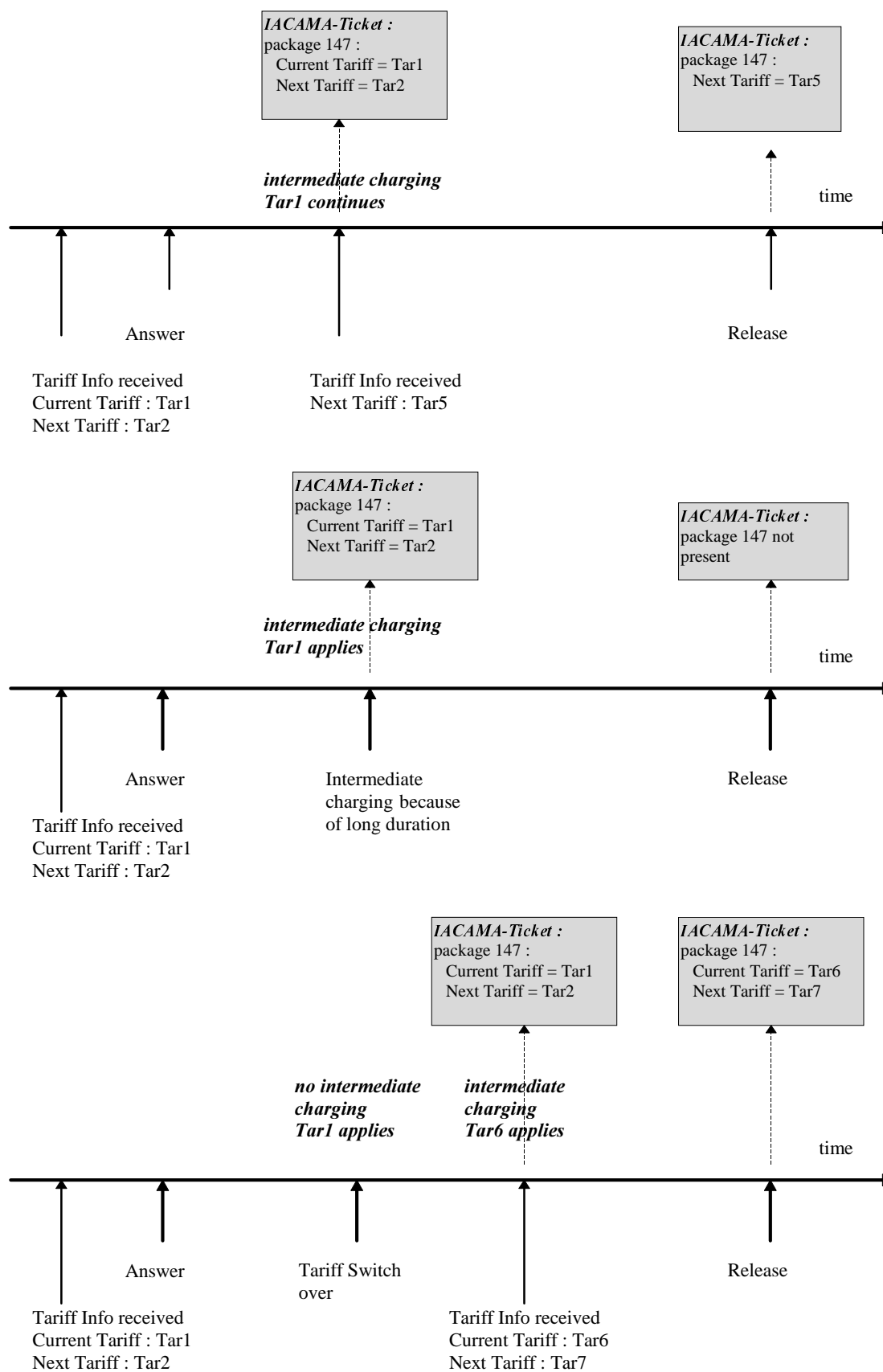


Figure 13 : Examples of Intermediate Charging in case Tariff Info is received

The **Acknowledgment Status** has the following meaning :

- **Waiting for Acknowledgment**

This value is only relevant for tickets generated in transit exchanges. It indicates that the transit exchange has sent the tariff to the originating exchange only for the purpose of Advice of Charge, and that the call was released before the acknowledgment was received.

- **Positively acknowledged**

This value indicates that the Tariff Info is positively acknowledged by the receiving exchange. It is relevant for Tariff info which was **sent** to a lower exchange, and also for Tariff info which is **received**. Remark that **received** Tariff Info is only registered in tickets when this information is **positively acknowledged**.

- **Negatively acknowledged**

This value is only relevant for tickets with Tariff Info which was **sent** to a lower exchange. Remark that **received** Tariff Info is only registered in tickets when this information is **positively acknowledged**.

- **Received but not sent**

This value is only relevant for CAMA tickets in a transit exchange. It indicates that tariff information was received from the destination side and positively acknowledged, but a relay to the lower exchanges was not possible because of signalling restrictions, or because the incoming trunk group did not have the authorization for sending tariff information (missing parameter SENDTAR in the CR TGRP command).

Remarks :

- Call Attempt and Call Setup Charges in Tariff Informations received **after answer** are **ignored**.
- The package can be present together with the flag "**no charge**". The flag "no charge" has the priority.
- The **currency** format is **not yet** available as common EWSD functionality, only the package structure is already predefined. The same is valid for the **Operator Data**.

C) Operating remarks

- The inclusion of package 147 in the AMA ticket is controlled via command ENTR CDTDAT : EXTNSD = CALLTAR;
- The parameter GCOS = SENDTAR in the CR TGRP command gives the incoming ISUP trunk group the authorization to sent the tariff information to a lower exchange.
- The parameter TKZOINF = YES in the CR ZOPT command gives the authorization to receive Tariff Information during the call.

D) Postprocessing remarks

- The total package length (in byte) is located at the second byte of the package.

E) Additional remark

Due to technical reasons, the package is in V12 only built in the IACAMA records when one of the following conditions apply:

- the IACAMA ticket is built in the originating exchange
- next to IACAMA, AMA is also active for the call in the same exchange
- the IACAMA ticket is built in a tariff determining exchange

5.1.2.21 Package 148 : CAC NEW DESTINATION

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (148)								binary; fixed value 148 (H'94)
2	2	1	PACKAGE LENGTH								binary; total length of package
3	3	1	CAC TYPE		PREFIX		NO OF DIG			binary; refer to description	
4	4	n	DIGITS (n - i)				DIGITS (n - i + 1)				packed digits, value (0:H'F), including prefix digits

Description of Field No 3 :

NUMBER OF CAC DIGITS : range between 0 - 7
Total length of the CAC (with prefix)

NUMBER OF PREFIX DIGITS : range between 0 - 3

CAC TYPE :

Value	Meaning
0	CAC type not available
1	network selected
2	subscriber default
3	subscriber dialled, used
4	subscriber dialled, not used
5	provided and not used

B) Functional description

This package contains the Carrier Access Code, which is returned from the SCP as part of the destination digits in case of an IN-call. Refer to the description of package 145 for an explanation of the CAC type.

C) Operating remarks

- The inclusion of package 148 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = CAC and ENTR CDTDAT : EXTNSD = NEWDEST;
Remark that EXTNSD = CAC is set by **default**, i.e. without entering the parameter EXTNSD = CAC within the ENTR CDTDAT command, package 148 will be collected (on condition EXTNSD = NEWDEST has been specified).

D) Postprocessing remarks

- The total package length (in byte) is located at the second byte of the package.

5.1.2.22 Package 149 : GLOBAL CHARGE REFERENCE

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (149)								binary; fixed value 149 (H'95)
2	2	1	PACKAGE LENGTH								binary; total length of package
3	3	1	reserved								F1 F1 = 1 : Global Charge Ref received (via ISUP) F1 = 0 : Global Charge Ref generated in own exchange
4	4	n	GLOBAL CHARGE REFERENCE								binary; refer to description

B) Functional description

The Global Charge Reference (GCR) is meant to be a unique identification for all kinds of tickets generated in the own and in different exchanges for the same call. A GCR should be generated in the originating exchange, and passed to all following exchanges via ISUP, such that it can be provided in all tickets generated on the call path.

The transport of the GCR via ISUP is **not yet** part of the common V12 EWSD functionality, but EWSD is prepared for future or project specific adaptations of ISUP, permitting the GCR to be transported.

When activated by the crafts person, the GCR is handled in the following way :

- in the exchange which receives a GCR via ISUP (this assumes a project specific ISUP change), the charging tickets will collect this GCR in the tickets and the Flag F1 set on 1. The GCR will also be sent via ISUP to the next exchange, assuming that the outgoing ISUP has been adapted for this. A GCR of max. 20 bytes can be received.
- in the originating EWSD exchange, or in the transit exchange when no GCR has been received, the GCR will be built by merging the Connection Id and the Signalling Point Code (SPC) of the own switch into one new field.

Byte Pos in Package 149	Length	Field	Remarks
4	3	SPC-Number	binary
7	4	Connection Id	binary

Table 6 : Layout of GCR

In this case, the flag F1 will be set on 0, indicating that the GCR has been generated in the own exchange.

Remarks :

- When in a transit exchange a GCR has been received, but the crafts person has not activated GCR, this **received** GCR will be transported to the next exchange (assuming ISUP adaptations for GCR transport).
- When in a transit exchange, no GCR has been received, the GCR generated in the own exchange will be sent to the next exchange (assuming ISUP adaptations).
- For Call Forwarding, Three Party and Large Conference, the same remarks are valid as for the Connection Id. Refer to the description of package 110.
- The Signalling Point Code is only unique within one network. In order to be unique over different networks, future extensions of the GCR will be necessary.

C) Operating remarks

- The inclusion of package 149 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = GCR;
When activated, it **replaces** the Connection Id.

D) Postprocessing remarks

- Package 110 (Connection Id) and package 149 (Global Charge Reference) are **never** available **at the same time**.
- The total package length (in byte) is located at the second byte of the package.
- A maximum package length of 23 bytes is possible (a GCR received over ISUP can have a max. length of 20 bytes).

5.1.2.23 Package 150 : AUDIT TRAIL

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (150)								binary; fixed value 150 (H'96)
2	2	1	PACKAGE LENGTH								binary; fixed value 08 (H'08)
3	3	4	SEQUENCE NUMBER								binary
4	7	1	CHECKSUM VALUE 1								binary
5	8	1	CHECKSUM VALUE 2								binary

B) Functional description

This package is gathered (when activated) to detect loss or corruption of tickets during transfer and post-processing .

The package contains two different kinds of information : Sequence Number and a Checksum. They can be activated separately by the crafts person.

- Sequence Number

The Sequence Number is a 4 byte running counter, which is incremented by one, each time a ticket is buffered (tickets are buffered in main memory before they are written to disk). That means, the Sequence Number reflects the order in which tickets are buffered and at the same time, the order in which tickets are written into the IACAMA file on disk.

The Sequence Number is only reset to zero, in case of EWSD recovery with initialization of transient data (this includes initial system start), and in case the Sequence Number rolls over from 4.294.967.295 to 0.

When the crafts person has only activated the Checksum, package 150 will be present with Sequence Number equal 0.

Remarks :

- * a Sequence Number is **not** provided in the system tickets (breakdown, overflow, date/time change), and is **not** incremented when a system ticket is generated.
- * a roll-over of the Sequence Number is **not** reported
- * the Sequence Number is **not** reset to zero, when the IACAMA feature is cancelled (MML command CAN IAFEAT) and activated again (ACT IAFEAT). That means, the Sequence Number can only be treated as a relative, not as an absolute number.

- Checksum

When activated by the crafts person, EWSD will calculate a checksum over the IACAMA ticket, with the algorithm as described in /12/. The algorithm results in a 2 byte checksum, which is always located **at the end** of the ticket. The algorithm is such that the receiving side (the Postprocessing) has to execute following algorithm to control whether the bit integrity is kept :

```

CHECKSUM_VALUE_1 = 0
CHECKSUM_VALUE_2 = 0

DO FOR i = 1 TO last byte of ticket

    CHECKSUM_VALUE_1 = (CHECKSUM_VALUE_1 + 'byte i of ticket') modulo 256
    CHECKSUM_VALUE_2 = (CHECKSUM_VALUE_1 + CHECKSUM_VALUE_2) modulo 256

END DO FOR

```

Figure 14 : Checksum algorithm

The result should be that both CHECKSUM_VALUE_1 = 0 and CHECKSUM_VALUE_2 = 0. If not, at least one of the bits in the AMA ticket has been corrupted.

When the crafts person has only activated the Sequence Number, package 150 will be present with CHECKSUM_VALUE_1 and CHECKSUM_VALUE_2 equal 0.

Remarks :

- * a Checksum is **not** provided in the system tickets (breakdown, overflow, date/time change)
- * the Checksum modulo 256 guarantees that all 1 bit corruptions are detected. Bit corrections are not possible. For a complete description of the quality of this algorithm, refer to /9/.

Important note :

because of the rather large number of calculations, the use of the Checksum has a **considerable negative impact** on the call throughput of EWSD.

C) Operating remarks

- The inclusion of package 150 in the AMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = SEQUENCE and/or ENTR CDTDAT : EXTNSD = CHECKSUM;

D) Postprocessing remarks

- This package has a fixed length of 8 bytes. The package length is included on byte 2 in the package itself.
- When present, this package is always the **last** package of the ticket
- The Sequence Number and Checksum can be activated separately. In order to permit the Postprocessing to quickly know whether the Sequence Number and/or Checksum are really present in the ticket, the **flags F37** and **F38** in the Fixed Part of the IACAMA ticket can be checked. When F37 and/or F38 is set, the Sequence Number and/or Checksum can quickly be located by using the Record Length which is always located at the second and third byte of the ticket. The Sequence Number is located at the end of the record - 5 bytes; the Checksum is located at the two last bytes of the record.
- The Checksum can be used to check whether the contents of the AMA ticket have not been corrupted between the moment of buffering in EWSD, and the moment of processing the tickets in the Postprocessing system.

5.1.2.24 Package 154 : ADDITIONAL CALL INDICATORS

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (154)								binary; fixed value 154 (H'9A)
2	2	1	PACKAGE LENGTH								total length of package binary; fixed value 4
3	3	1	reserved				F14	res	F12	F11	flags
4	4	1			F26	F25	F24	F23	F22	F21	flags

Description of Field No 3 : FLAGS

Flags	Value	Meaning
F11	1	CTX subscriber
F12	1	CTX attendant
F14	1	International call
F21	1	Local intercom call
F22	1	Remote intercom call
F23	1	Intra charging site call
F24	1	Inter charging site call
F25	1	Public call
F26	1	Public business call

B) Functional description

These flags have the following meaning :

- **F11**
This flag indicates that the calling line is a Centrex subscriber.
- **F12**
This flag indicates that the calling line is a Centrex attendant.
- **F14**
This flag is set in following cases :
 - * the call comes in over an international trunk group (an international trunk group is administered by the parameter GCOS = INTLTRK in the CR TGRP command)
 - * the received MFC category indicates international (values 7, 8, 9, 10 of the Category)
 - * received calling party number is an international number (NADI = international)
 - * incoming international call in O:IAM (bit in Forward Call Indicator)
- **F21**
This flag is only relevant for Centrex subscribers. It indicates that the call is a Local Intercom call, i.e. a call between two members of the same Centrex Group
- **F22**
This flag is only relevant for Centrex subscribers. It indicates that the call is a Remote Intercom call, i.e. a call between two members of a different Centrex Group

- **F23**

This flag is only relevant for Centrex subscribers. It indicates that the call is an Intra Charging Site call, i.e. a call between two Centrex subscribers whose Charging Site has the same number. A Charging Site is a subdivision of one Centrex Group (administered via COSDAT = CHSITE-<charging site> in the CR CXSUB and CXPBX commands).

Remark that F23 can also be set when A- and B-party belong to a different Centrex group.

- **F24**

This flag is only relevant for Centrex subscribers. It indicates that the call is an Inter Charging Site call, i.e. a call between two members of a different Charging Site.

Remark that all Centrex members have a Charging Site assigned (if not explicitly administered, the Charging Site = 1). That means, as soon as a call is made between two Centrex members, either Flag F23 or F24 is set.

- **F25**

This flag is only relevant for Centrex subscribers. It indicates that the call is a Public Call. A Public Call is detected by an own access code (administered with the command CXCPY).

Remark that when F25 is set, none of the flags F21, F22, F23, F24, F26 can be set.

- **F26**

This flag is only relevant for Centrex subscribers. It indicates that the call is a Public Business Call. A Public Business Call is detected by an own access code (administered with the command CXCPY). Remark that this flag is independent of the destination, i.e. the destination can be a public or a Centrex subscriber.

Remark that when F26 is set, none of the flags F21, F22, F23, F24 can be set. Moreover, either F25 or F26 is set; not both of them.

The following figure illustrates the use of these flags :

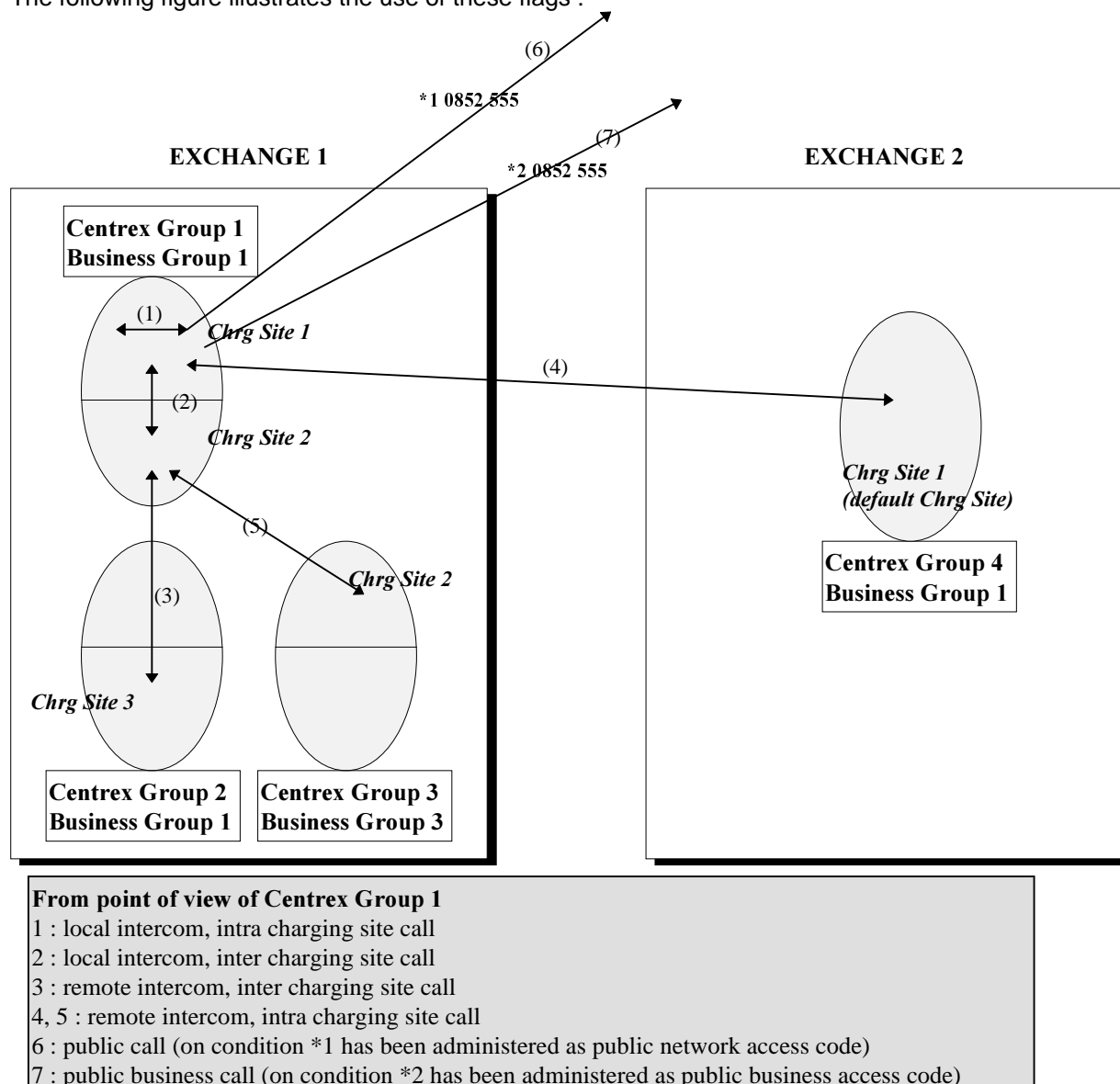


Figure 15 : illustration of Additional Call Indicators

C) Operating remarks

- The inclusion of package 154 in the IACAMA ticket is controlled via command
ENTR CDTDAT : EXTNSD = CALLIND;

D) Postprocessing remarks

- The total package length (in byte) is located at the second byte of the package.
- When the inclusion of package 154 is administered, the package will **always** be present, even when none of the flags is set.

5.1.2.25 Package 157 : OUTGOING PA-SLAVE IDENTIFICATION

A) Syntax description

Field No	Byte Pos	Length (bytes)	Field Name								Remarks
			7	6	5	4	3	2	1	0	
1	1	1	PACKAGE NUMBER (157)								binary; fixed value 157 (H'9D)
2	2	1	PACKAGE LENGTH								binary; total length of package
2	2	1	LAC LENGTH				NO OF DIGITS				binary
3	3	n	DIGIT (n - i)				DIGIT (n - i + 1)				packed digits, values (H'0:H'F)

B) Functional description

This package is provided in case the B-side of the call is a **PA-Slave**. It contains the Pilot Directory Number of this PA-Slave. The number of digits, indicated by the field NO OF DIGITS, includes the LAC digits.

Remarks :

- The **incoming** PA-Slave identification is **not** provided, since this identification is present as Record Owner, in case the call originates on a PA-Slave. Remark that the digits in package 157 are stored in the same way (LAC + further digits) as in the field Record Owner in the fixed part of the ticket.
- The identification of the PA-Slave can **not** be taken from the package 101 (Partner Directory Number) for the following two reasons :
 - * The PBX Overflow feature permits the call to be routed to another PBX, in case the first PBX to which the call is offered is not able to accept the call. Package 101 contains only the number of the first PBX; the directory number of the second PBX to which the call is routed is **not** contained in the ticket.
 - * In a typical configuration, the Pilot Directory Number of the PA-Slave is **not** dialled by the calling party, and hence not contained in package 101 (subscribers can be prevented to dial the PDN of a PA Slave, by adding hexadecimal digits to this PDN). The PDN will be put in front of the dialled digits by **digit conversion** (parameter CONVCODE in the CR DEST command) in the switch. With these converted digits, the call is routed to the PA-Slave. The parameter SSDI (Start Sending Digits) in the CR PBX command for the PA-Slave can be used to cut off the PDN in the called party digits which are sent over the PA-Slave to the other network.

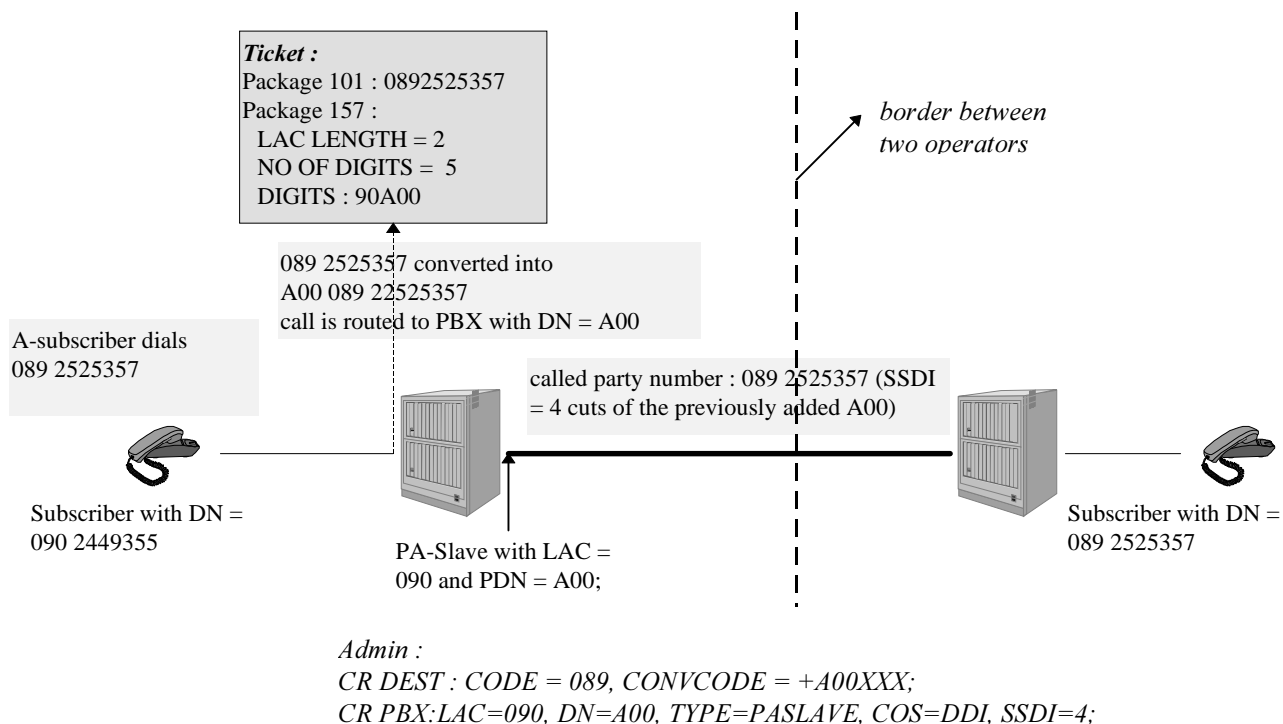


Figure 16 : Example of the use of package 157

C) Operating remarks

- The package is not administrable.

D) Postprocessing remarks

- The total package length (in byte) is located at the second byte of the package.
- PA-Slave is typically used for the inter-connection between two exchanges of different network operators via subscriber interface (refer to chapter 1). When PA-Slave is only used for this purpose, the presence of this package can be used to distinct calls which are routed to another operator.

5.2 Date Time Change Record

A) Syntax description

Field No.	Byte Pos	Length (bytes)	Field Name	Remarks
1	1	1	RECORD IDENTIFIER	binary; fixed value 1
2	2 - 7	6	DATE TIME	binary
3	8	1	CORRECTION REASON	binary
4	9	1	CORRECTION SIGN	binary
5	10	1	DELTA SECONDS	binary
6	11	1	DELTA MINUTES	binary
7	12 - 32	21	RESERVED	

Description of Field No 1 : RECORD IDENTIFIER

This field identifies the type of IACAMA Record.
In case of "Date-Time Change Record", the value is 1.

Description of Field No 2 : DATE/TIME

This field contains the new date/time.

Bit Position		Meaning			
Byte 1	Bit 7	time secure	(0 = yes, 1=no)		
Byte 1	Bits 0-6	year	right aligned	binary	value range 0-99
Byte 2	Bits 0-7	month	right aligned	binary	value range 1-12
Byte 3	Bits 0-7	day	right aligned	binary	value range 1-31
Byte 4	Bits 0-7	hour	right aligned	binary	value range 0-23
Byte 5	Bits 0-7	minutes	right aligned	binary	value range 0-59
Byte 6	Bits 0-7	seconds	right aligned	binary	value range 0-59

Description of Field No 3 : CORRECTION REASON

This field specifies the reason of the date-time correction.

Value	Meaning
0	Correct time Adjust time of the EWSD processor clock
1	Correct season Switch from/to summer or winter time
2	Enter time Set the EWSD processor clock to an initial value

Description of Field No 4 : CORRECTION SIGN

This field specifies the direction of a date/time correction

Value	Meaning
0	none
1	plus (forward)
2	minus (backward)

Description of Field No 5 : DELTA SECONDS

Value	Meaning
0-59	adjustment in seconds of the EWSD processor clock after a "correct time" command (MML: CORR TIME). The value is zero if the correction reason is "enter time" or "correct season"

Description of Field No 6 : DELTA MINUTES

Value	Meaning
0-120	adjustment in minutes of the EWSD processor clock after a "correct season" command (MML: CORR SEASON). The value is zero if the correction reason is "enter time" or "correct season"

B) Functional description

This record is generated when the time in EWSD is changed due to time correction, or daylight savings time correction.

C) Operating remarks

- The inclusion of this package in the AMA file is controlled via command
ACT IAOPT : SPREC = DTC;

D) Postprocessing remarks

- This package has a fixed length of 32 bytes. The package length is **not** included in the package itself.

5.3 Overflow Record

A) Syntax description

Field No.	Byte Pos	Length (bytes)	Field Name	Remarks
1	1	1	RECORD IDENTIFIER	binary; fixed value 3
2	2 - 7	6	DATE TIME OF OVERFLOW START	binary
3	8 - 13	6	DATE TIME OF OVERFLOW END	binary
4	14 - 17	4	NUMBER OF LOST RECORDS	binary
5	18 - 32	15	RESERVED	binary

Description of Field No 1 : RECORD IDENTIFIER

This field identifies the type of IACAMA Record.
In case of "Overflow Record", the value is 3.

Description of Field No 2 and 3 : DATE/TIME OF OVERFLOW

Refer to section 5.2 "Date Time Change Record", description of Field No 2.

Description of Field No 4 : NUMBER OF LOST RECORDS

Value	Meaning
0 - H'7FFFFFFF	In case of IACAMA buffer overflow new IACAMA records can no longer be stored in memory and are lost. Each time an IACAMA record is lost because of lack of buffer memory, the counter "Number of lost Records" is incremented.

B) Functional description

This record is generated in case generated AMA records can not be buffered because the buffers are full (e.g. because the disk is not available). As soon as the buffers become available again, the generated Overflow record is buffered itself.

C) Operating remarks

- The inclusion of this package in the AMA file is controlled via command
ACT IAOPT : SPREC = OVF;

D) Postprocessing remarks

- This package has a fixed length of 32 bytes. The package length is **not** included in the package itself.

5.4 Breakdown Record

A) Syntax description

Field No.	Byte Pos	Length (bytes)	Field Name	Remarks
1	1	1	RECORD IDENTIFIER	binary; fixed value 2
2	2 - 7	6	RESERVED	binary
3	8 - 13	6	DATE TIME OF RECOVERY	binary
4	14	1	RECOVERY LEVEL	binary
5	15 - 32	18	RESERVED	binary

Description of Field No 1 : RECORD IDENTIFIER

This field identifies the type of IACAMA Record.
In case of "Breakdown Record", the value is 2.

Description of Field No 3 : DATE/TIME OF RECOVERY

Refer to section 5.2 "Date Time Change Record", description of Field No 2.

Description of Field No 4 : RECOVERY LEVEL

Value	Meaning
0	not used
1	Recovery level 1_1
2	Recovery level 1_2
3	Recovery level 1_3
4	Recovery level 2_1
5	Recovery level 2_2

B) Functional description

This record is generated in case a recovery takes place in EWSD. .

C) Operating remarks

- The inclusion of this package in the AMA file is controlled via command
ACT IAOPT : SPREC = BRD;

D) Postprocessing remarks

- This package has a fixed length of 32 bytes. The package length is **not** included in the package itself.

5.5 Filler Records

Format of Filler Records to fill one or two bytes:

Field No.	Byte Pos	Length (bytes)	Field Name	Remarks
1	1	1	RECORD IDENTIFIER	binary; fixed value 128 = H'80

Format of Filler Records to fill more than two bytes:

Field No.	Byte Pos	Length (bytes)	Field Name	Remarks
1	1	1	RECORD IDENTIFIER	binary; fixed value 129 = H'81
2	2 - 3	2	RECORD LENGTH (n + 3)	binary
3	4	n	FILLER BYTES	value 0

Format of Filler Records to fill 32 Bytes:

Field No.	Byte Pos	Length (bytes)	Field Name	Remarks
1	1	1	RECORD IDENTIFIER	binary; fixed value 0 = H'00
2	2-32	31	FILLER BYTES	value 0

Description of Field No n : FILLER BYTES

In the Filler record to fill more than 2 bytes (RID=129) and to fill 32 bytes, the FILLER BYTES contain **the value 0**.

The maximum record length is the rest of 2, 4 or 8 Kbyte, according to the parameter FILRECSZ in the ACT IAFEAT command, because each data block contains at least one connection ticket or Date Time Change record or Overflow record or Breakdown record.

6 Operation Instructions

6.1 Operating instructions for activating the IACAMA feature

With the MML-command

```
ACT IAFEAT : TYPE = IARA, SIZE = <size of the cyclic file IA.ICIAR>, FILRECSZ = <size of file record>, NUMFILES = <specifies the number of files : 1 or 2>;
```

the IACAMA-feature is activated, i.e.

- buffers are created in main memory
- one or 2 cyclic files (IA.ICIAR1 and IA.ICIAR2) are created on disk

For a detailed description of buffer handling in EWSD, refer to the next chapter.

For details about cyclic file handling, refer to chapter 6.5.

With the MML-command

```
CAN IAFEAT : TYPE = IARA;
```

the IACAMA feature is deactivated, that means tickets are no longer buffered and the cyclic file IA.ICIAR is closed. Remark however that the cyclic file is **not** deleted. When the IACAMA feature is activated again, and the cyclic file IA.ICIAR is already present, the file is re-opened. Deactivation and re-activation do **not** affect the copy area (the area which has been transferred and not yet released, refer to chapter 6.5)

6.2 Buffer Handling in EWSD

When an IACAMA ticket is generated, it is not immediately written on EWSD disk, but it is first buffered in main memory. This buffering is used to minimize writing actions on disk. One buffer has a size of 56 Kbyte. The number of file records in one buffer depends on the size of the file record : up to 7 file records of 8192 byte, up to 14 file records of 4096 byte or up to 28 file records of 2048 byte. The numbers of buffers can be expanded from 3 to 5 with ODAGEN (Online Database Generation: method to resize the databases in EWSD).

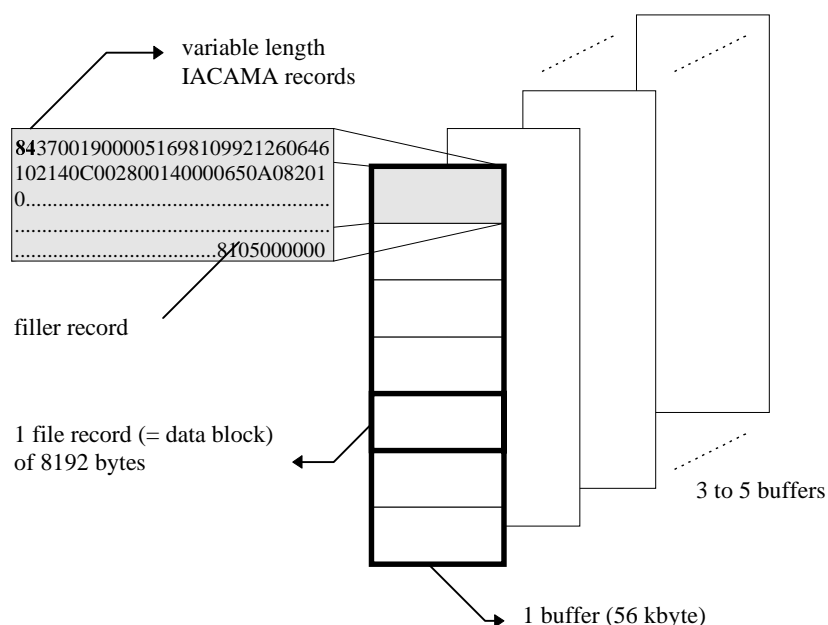


Figure 17 : Buffering of IACAMA tickets

In normal operation, buffers are written to disk as soon as they are completely filled. In case the disk is not available, tickets are written into the next buffer. At regular times, the disk is checked for availability. When the disk comes back into operation, all buffers which have been completely filled are written onto disk. It is ensured that the buffers are written to disk in the same sequence as they have been filled up with tickets.

At the other hand, measures are taken to prevent that tickets remain too long in the buffers before they are stored on disk. Each time a file record of 2, 4 or 8 Kbyte has been completely filled up, it is checked whether the previous writing of a buffer to disk was more than 2 minutes ago. If so, all the data blocks which have been written into the buffer so far, are written onto disk.

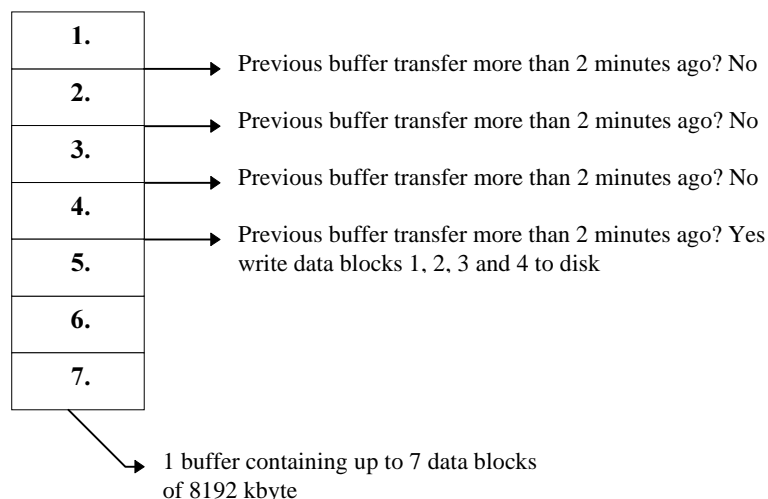


Figure 18 : Security measures for buffers

It is also possible to force a buffer transfer with the command

TRANS BUFFER : TYPE = IARA;

In this case, the current data block (i.e. the data block which is currently filled up with variable length IACAMA records), is completed with a filler record, and all data blocks which have been written into the buffer so far are written onto disk.

Remarks :

- when the disk is not available and all buffers have been filled up with tickets, newly generated tickets **are lost**. In this case, an overflow record is generated (if activated by the crafts person) and stored as soon as a buffer is available again.

6.3 Operating instructions for customizing the IACAMA ticket

With the MML-command

ENTR CDTDAT : REC = IACAMA, EXTNSD = <extra data>, ALTNSD = <alternative data>;

the contents of the IACAMA-ticket can be controlled (the control function of ENTR CDTDAT for other kinds of tickets is not considered).

• Extra Data

The parameter EXTNSD controls the inclusion of additional packages in the IACAMA ticket. The following table gives an overview of all possibilities. Remark however that in many cases, there are additional criteria for the presence of a package in the IACAMA ticket. Refer to the table in chapter 4.1 for an overview, and to chapter 5.1.2 for a detailed description of all additional criteria.

Parameter EXTNSD =	Package
CHARGES	package 103 (Charge Units for Connection)
CONNID	package 110 (Connection ID); is overruled by EXTNSD=GCR
NEWDEST	package 128 (New destination) and package 148 (CAC new destination) when additionally EXTNSD=CAC is set
TRAFQUAL	package 130 (Traffic quality data)
BMETPULS	package 138 (B Meter Pulses)
CLGDN	package 142 (Calling party number)
CAC	package 145 (CAC number) and package 148 (CAC new destination) when additionally EXTNSD=NEWDEST is set
THRDPRTY	package 146 (Third party number)
CALLTAR	package 147 (Tariff info)
GCR	package 149 (Global charge reference); overrules EXTNSD=CONNID
CHECKSUM	checksum in package 150 (Audit trail)
SEQUENCE	sequence number in package 150 (Audit trail)
CALLIND	package 154 (Additional call indicators)

CAC (Carrier Access Code), CONNID (Connection Identification) and TRAFQUAL (Traffic Quality Data) are set by **default**, i.e. **without explicit activation** of these packages with ENTR CDTDAT, these packages are **collected**.

- **Alternative Data**

The parameter ALTNSD controls the appearance of alternative data in the IACAMA ticket. There are following possibilities :

Parameter ALTNSD =	Data in IACAMA ticket
DURATCUM	cumulated call duration in package 100 (Date/Time/Duration) instead of call duration per intermediate time interval
RANSWTIM	Begin date/time in package 100 (Date/Time/Duration) instead of end date/time
RECTRCIC	packages 107 and 108 (Trunk identification Incoming and Outgoing in CIC Format) instead of packages 105 and 106 (Trunk identification Incoming and Outgoing)
TGN2	second trunk group identification (according to parameter TGN2 in the CR TGRP command) instead of first trunk group identification (according to parameter TGNO in the CR TGRP command) in the package 105, 106, 107 and 108 (Trunk Identification Incoming and Outgoing, in normal and in CIC format)

6.4 Operating instructions for controlling the generation of IACAMA-tickets

Following flags control the generation of IACAMA-tickets

- **“IACAMA for unsuccessful calls “**

This flag is set with the MML-command

ENTR CDTDAT : MARKS = UCALLIAC;

If this flag is set and charging has to be done with IACAMA, an IACAMA ticket will be generated, independent whether the call is answered or not.

6.5 Operating instructions for handling of the cyclic file IA.ICIAR(1)(2)

6.5.1 Cyclic file characteristics

After the cyclic file IA.ICIAR(1)(2) has been created during the activation of the AMA feature, file records (fixed length records with a length of 2, 4 or 8 Kbyte) are written continuously into the cyclic file. A Write Pointer in the header of the cyclic file points to the position where the next file record is to be written to.

When a transfer command is issued on the cyclic file (either local or remote; see further), the copy area (the area which is transferred) is marked in the header of the cyclic file with two additional pointers : a pointer to the begin of the copy area, and a pointer to the end of the copy area. After the first transfer, the copy area is frozen, that means, all subsequent transfer requests on the cyclic file will lead to the transfer of the same data again. The copy area has first to be released before new data can be transferred. The **release of the copy area** is either done via the MML-command REL CYCFILE, or via the FTAM Delete function (see further).

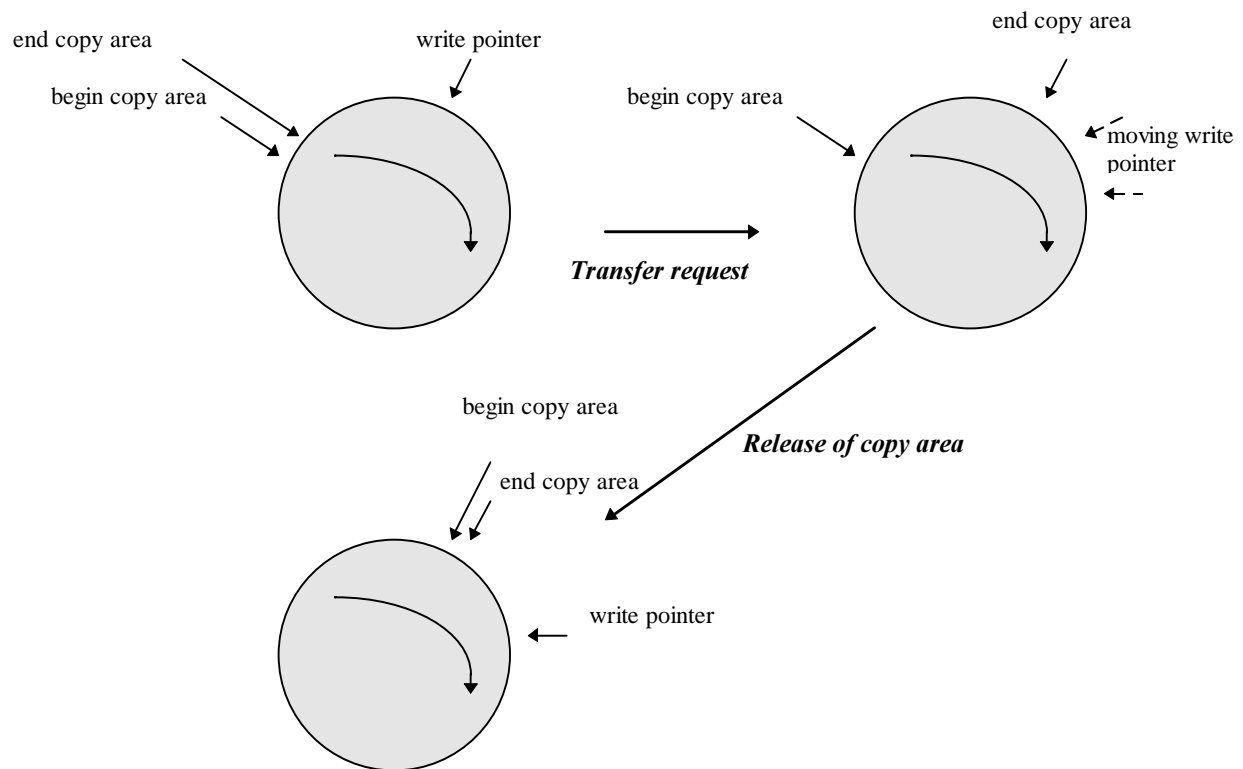


Figure 19: Handling of the Cyclic File

6.5.2 Local transfer of the cyclic file

A local transfer is a transfer from the cyclic file to another file on EWSD disk, or to MTD or MOD (Magnetical Optical Device). A local transfer is handled with the MML-command TRANS FILE.

6.5.3 Remote transfer of the cyclic file

There are two possibilities for remote transfer :

- with the FTINEA protocol, which is a Siemens specific transfer protocol.

In this case, EWSD can either be **initiator** or **responder** of the transfer, that means, the transfer request can be launched on the EWSD side or on the remote side.

In case EWSD is the initiator, the transfer is started with the MML-command TRANS FILE.

- with the FTAM protocol

FTAM is an international standard (ISO 8571) for remote file handling. EWSD complies to this standard with following **restrictions** :

- * EWSD can only be **responder**, i.e. the transfer can only be requested at the remote side.
- * it is not possible to transfer a **cyclic file** from the remote side to EWSD. Remark however that it is possible to transfer a **SAM** file from the remote side to EWSD (i.e. for SAM files, EWSD can be **sender** or **receiver**; EWSD can however not request the transfer itself).
- * as File Management actions from the remote side, "**Delete file**" and "**Read File Attributes**" are possible.

EWSD has implemented a special logic, when a cyclic file is opened with Exclusive Access as Concurrency control in the File Open during an FTAM transfer. The following figure gives an example of an FTAM flow, and indicates where the Exclusive Access has to be specified :

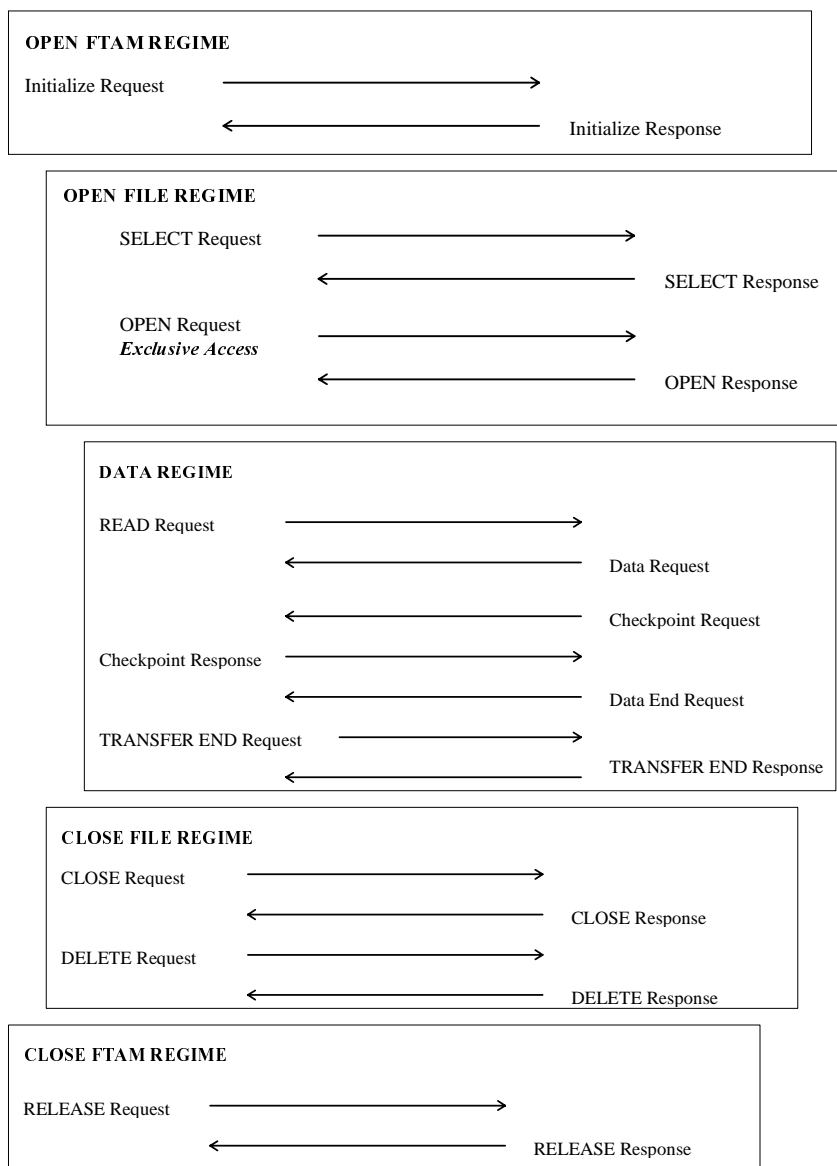


Figure 20 : Example of FTAM flow

The EWSD specific treatment of a cyclic file which is opened with the Exclusive File Access is the following :

- * a copy area which has been transferred once, **can not be transferred again**. Remark however, that a copy area which has been transferred once **can still be copied to tape**. When **first** a backup to tape is executed, then an **FTAM transfer is not possible anymore**.

This logic is implemented in the following way. When a cyclic file has at least once been successfully transferred, a so-called TRANSMITTED FLAG is set. The TRANSMITTED FLAG is set independent whether an FTAM transfer or a local transfer (disk to disk or disk to tape) was executed.

When an FTAM transfer request is launched with Exclusive Open Access, a check is made whether the TRANSMITTED FLAG is set. If so, the FTAM request is rejected.

The check on the TRANSMITTED FLAG is **not** carried out for the back up to tape, since for back up to tape, the Exclusive Open Access is never used.

- * when an FTAM transfer is aborted, **the copy area remains the same**. This is realized in the following way : when an FTAM transfer request is launched with Exclusive Open Access, an **established** copy area is taken over. A copy area is established when a file is opened for reading. Establishing the copy area means that the begin and the end of the copy area are stored in the disk catalog. This is illustrated in the first figure, where a transfer is requested with a copy area which is not yet established. The establishing

of the copy area is made at the begin of the transfer by storing the begin and end copy area (the end copy area pointer equals the actual write pointer).

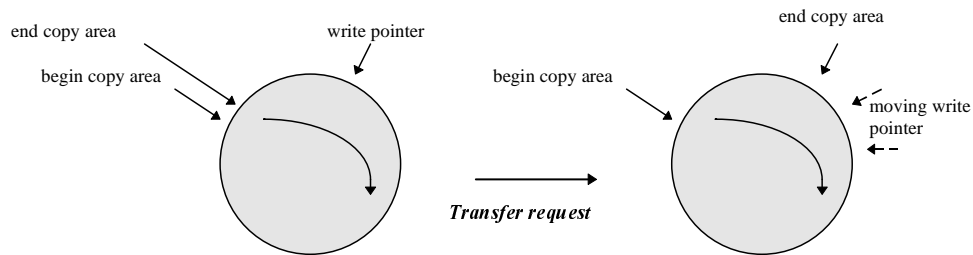


Figure 21 : Establishing of a copy area in a cyclic file

When the FTAM transfer would be aborted, and a new FTAM request with Exclusive Access is launched, the already established copy area will be taken over.

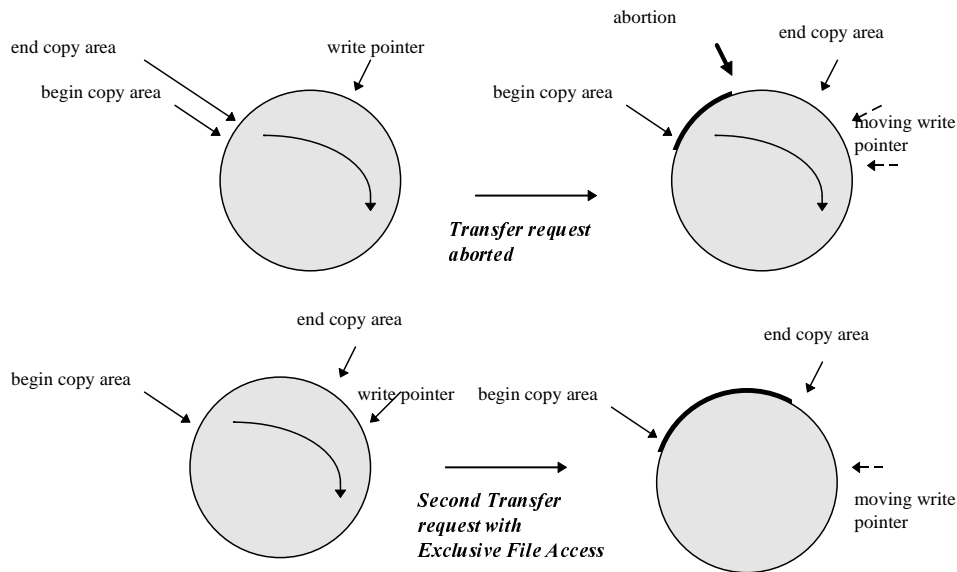


Figure 22 : Aborted FTAM transfer of a cyclic file

When the second FTAM transfer would be requested without Exclusive File Access, a new copy area would be established, whereby the end copy area pointer would be equal to the actual write pointer.

Remarks :

- * EWSD sustains the FTAM **Recovery function**. When a transfer is interrupted by a transient error, the transfer can be continued at a **fixpoint**, i.e. an intermediate point during the transfer which has been negotiated between sender and receiver.
- * The Exclusive Access can also be specified in the SELECT Request. In this case, the Exclusive Access has additionally to be specified in the OPEN Request, since the Concurrency control in the OPEN Request must be at least as severe as the Concurrency control in the SELECT request.
- * Remark the difference between **abortion** and **interruption** of FTAM. FTAM is **interrupted** when transient errors are detected (e.g. connection between EWSD and remote host is disturbed); FTAM is **aborted** when permanent errors are encountered (e.g. file not accessible) or on explicit request by the operator. An interrupted FTAM transfer will be recovered by the FTAM Recovery function. **When the FTAM Recovery function is not used, a transient error will lead to an abortion of FTAM.**
- * When more than one remote host for FTAM transfer is used, the second host can take over an **aborted** FTAM transfer from the first Host. The Exclusive File access prevents that the two hosts

will transfer the same data. **An interrupted FTAM transfer can however not be taken over by the second host** : the FTAM recovery function will continuously try to re-establish the previous connection, i.e. the **connection between EWSD and the first host**.

In order to allow the second host to take over the FTAM transfer in case a **transient error** is encountered, the **FTAM Recovery function must not be used**. The lack of the FTAM Recovery function will lead to the abortion of FTAM in case a transient error is encountered, which permits the second host to take over the FTAM transfer request.

- * It is possible to limit the max. number of file records to be transferred in one FTAM transfer request. This is done by administration in EWSD (parameter MAXCOPE in the MML command SET FGRPATT).
- * The FTAM Delete functions is used to release the copy area. This release of the copy area is executed, **independent of the number of safety copies**, administered via the parameter SAFCOP in the SET FGRPATT command.

6.6 Hints for the dimensioning of the IACAMA buffers and IACAMA file

In EWSD, the number of buffers can be changed from 3 to 5, and the size of the cyclic file can be specified at the moment the IACAMA feature is activated (parameter SIZE in the ACT IAFEAT command).

In order to have an idea of the required number of buffers and size of the cyclic file, following formulas can be used :

• amount of generated data

$$A = BHCA \cdot \beta \cdot L_{av} / 3600$$

- A : amount of generated data in bytes per second
 BHCA : Busy Hour Call Attempts (number of calls which are set up and released in one hour)
 β : value between 0 and 1, indicating the proportion of calls for which a ticket has to be made
 L_{av} : average length of a ticket

Example :

- BHCA : 500 000 /hour
 β : 0.3
 As example is taken :
 * 50% of the calls require a ticket
 * for these calls, only the successful ones get a ticket
 * 60% of the calls are successful
 L_{av} : 50 bytes

$$A = 2083 \text{ bytes / second}$$

• time interval between two buffer outputs on disk

$$T = 56 \text{ Kbyte} / A$$

- A : amount of generated data in bytes per second
 T : time interval between two buffer outputs
 56 Kbyte : size of one buffer

Example :

- A : 2083 bytes / second
 T = 27 seconds

- **maximum disk down time before tickets are lost**

$$O = T \cdot N_{\text{buf}}$$

O : maximum disk down time before tickets are lost
 T : time interval between two buffer outputs
 N_{buf} : number of buffers

Example :

T : 27 seconds
 N_{buf} : 3

O = 81 seconds

- **average occupation of the IACAMA file on disk**

$$S = A \cdot T_{\text{trans}}$$

S : average occupation of the IACAMA file
 A : amount of generated data in Kbyte per second
 T_{trans} : time between two transfers in seconds

Example :

A : 2083 bytes / second
 T_{trans} : 86400 (assumed is a transfer once a day, i.e. time between two transfers = 24 * 3600 = 86400)

S = 179971200 bytes = ca. 171 Mbyte

Remarks :

- assuming FTAM transfer via X.25, the average transfer duration can be calculated with the formula

$$D_{\text{trans}} = A \cdot T_{\text{trans}} / S_{\text{x.25}}$$

D_{trans} : average duration of FTAM transfer
 A : amount of generated data in Kbyte per second
 S_{x.25} : transfer speed over X.25 in Kbyte per second
 T_{trans} : time between two transfers in seconds

Example :

A : 2083 bytes / second
 S_{x.25} : 5825 bytes / second (remark that X.25 has a transfer speed of 64 kbit/s, but not all transferred bits are user data; in praxis, a transfer speed of ca. 20 Mbyte/hour is used)
 T_{trans} : 86400 (assumed is a transfer once a day, i.e. time between two transfers = 24 * 3600 = 86400)

D_{trans} = 30896 seconds = ca. 8,5 hour

6.7 Hints for the Postprocessing

6.7.1 Identifying the contents of the IA.ICIAR(1)(2) file

In EWSD V12, following records can be present in the IA.ICIAR file :

Record Id	Record Length	Description
0 (H'00)	fixed 32 bytes	Filler Record for 32 bytes
1 (H'01)	fixed 32 bytes, not stored in the record	Date/time Change Record
2 (H'02)	fixed 32 bytes, not stored in the record	Breakdown Record
3 (H'03)	fixed 32 bytes, not stored in the record	Overflow Record
128 (H'80)	fixed 1 byte	Filler Record one byte
129 (H'81)	variable, located at the second byte	Filler Record to store more than two bytes
132 (H'84)	variable, located at the second byte	variable length IACAMA records

Table 7 : Overview of Record Identifiers

6.7.2 Identifying the contents of one variable length IACAMA record

The treatment of Data Packages by the IACAMA Postprocessing must be designed in such a way that Data Packages which are not used in a project can be rejected as described below.

With every future EWSD version, new data packages may be introduced because of new IACAMA related features. New features may be implemented in EWSD common for all projects, although the feature is only required by one or a limited number of projects. Therefore, new data packages may be collected in IACAMA records in a project that does not evaluate these packages in the IACAMA Postprocessing.

To solve this problem of compatibility, new data packages that are project specific and of which the collection can not be activated by MML-commands, are always provided with a length indication stored in byte 2 of the package. In this way, Postprocessing can be designed to reject data packages with a package number not relevant for the project and address a further data package by adding the length stored in byte 2.

Remarks :

- Packages for which the collection is activated by MML are considered to be required in the project. The layout and length of such a package must be considered by postprocessing.
- Packages that form a standard part of the ticket (refer to the table in chapter 4.1) must also be considered by postprocessing.

Such packages can :

- * have no length indication, i.e. the package has a fixed length (which is indicated with the package description, section 5.1.2)
- * have the package length written on byte 2 of the package
- * have a length indication on byte 2. In such cases the package length must be calculated according to the formulas listed with the package description (section 5.1.2).

The table in chapter 4.1 indicates for each package :

- whether the package has a fixed or a variable length
- whether the package contains the package length on byte 2 or whether the package length must be calculated.

6.7.3 Average Record length

The average length of the IACAMA record depends on the operational settings for the layout of the IACAMA record.

The following table gives some typical examples

Conditions	Record length
registration of an analog call in a transit exchange; fixed numbering plan with 7 digits; 7 dialed digits; length of Local Area Code = 3; only standard data registered; no intermediate charging	Fixed Part: 13 bytes package 100 : 11 bytes package 101 6 bytes package 105 9 bytes package 106 9 bytes package 110 6 bytes package 130 5 bytes Total length 59 bytes
registration of an ISDN call in a transit exchange; fixed numbering plan with 7 digits, 10 dialed digits (area code included); length of Local Area Code = 3; Charge Units; no intermediate charging	Fixed Part: 13 bytes package 100 : 11 bytes package 101 7 bytes package 102 4 bytes package 103 4 bytes package 105 9 bytes package 106 9 bytes package 110 6 bytes package 130 5 bytes Total length 68 bytes

Table 8 : Examples for IACAMA Record length

7 Delta to Previous Versions

7.1.1 Version 6.2

New

- new datapackage 120 (Category)

7.1.2 Version 7.1

7.1.3 Version 8

7.1.4 Version 9

New

- new datapackage 130 (Traffic Quality Data)

Changes

- package 110 (Connection Id) : the fields Speech Channel and Processor Number are replaced by a 3 byte field without separate components (the total length remains the same)
- Call Failure Type is no longer built (**not upwards compatible change !**)

7.1.5 Version 10

New

- new datapackage 103 (Call charges)
- new datapackage 117 (Project specific data)
- new datapackage 134 (Durations before Answer)
- new datapackage 138 (B Meter Pulses)

Changes

- package 110 (Connection Id) : also for analog (before only for ISDN)

7.1.6 Version 11

New

- new datapackage 128 (New Destination)
- new datapackage 145 (CAC Number)
- new datapackage 146 (Third Party Number); this datapackage is only used for Call Transfer
- new datapackage 147 (Tariff Info)

Changes

- package 110 (Connection Id)
 - * field Connection Id is extended from 3 to 4 bytes

7.1.7 Version 12

New

- new datapackage 142 (Calling Party)
- new datapackage 148 (CAC New Destination)
- new data package 149 (Global Charge Reference)
- new data package 150 (Audit Trail)
- new data package 154 (Additional Call Indicators)
- new flags in header :
 - * "Analog Subscriber" (F24)
 - * "ISDN Subscriber" (F25)
 - * "Sequence Number present" (F37)
 - * "Checksum present" (F38)
- Description of field Sequence Number in package 150 (Audit Trail) changed : sentence 'least significant byte at the highest adress' is deleted
- Package 'New Destination' field no 5 NO_OF_DIGITS only 5 bits
- Package 'Calling Party Number' field no 5 NO_OF_DIGITS only 5 bits
- New CAC type : "provided and not used"

Changes

- package 130 (Traffic Quality Data) : Coding Standard added
- package 146 (Third Party Number) : Additional Info added
- package 147 (Tariff Info) : contents completely changes (**not upwards compatible change !**)
- package 100 (Duration) : Duration is set on 0 in case the call is not answered
- Connection Identification is the same on A- and B-side of the call
- Administration : following packages can be activated by the crafts person :
 - * package 130 (Traffic Quality Data)
 - * package 110 (Connection Id)
 - * package 145 (CAC)
 - * package 148 (CAC New Destination).
- Change of package Third Party Number to include Call Forwarding

8 Examples

This chapter contains an example of a binary file with IACAMA records.



IAC_V12W.BIN

The file IAC_V12W.BIN is a copy of the output of a file transfer from the EWSD disk file IA.ICIAR. As such it can be used as a “try out” input for the offline post processing or any other preprocessing on an offline system.



IAC_V12W.REC

The ICON IAC_V12W.REC represents a text file which is the output of an Siemens offline PC-tool. This tool converts the tickets with binary format into a text-file where the tickets are listed in a readable format.

The text file IAC_V12W.REC is the conversion which corresponds to the IAC_V12W.BIN.

The following WINWORD Version 6.0 instructions have to be carried out, in order to extract the binary and text files to a local disk

1. Point the mouse to the ICON for the binary file (IAC_V12W.BIN) and pop up the ICON-menu by clicking the right mouse button.
2. Select EDIT PACKAGE from this ICON-menu.
3. From the OBJECT PACKAGER window, pop up the FILE-menu from the toolbar
4. Select SAVE CONTENTS from this FILE-menu
5. In the SAVE CONTENTS window define the directory and the file name where the binary file has to be stored.