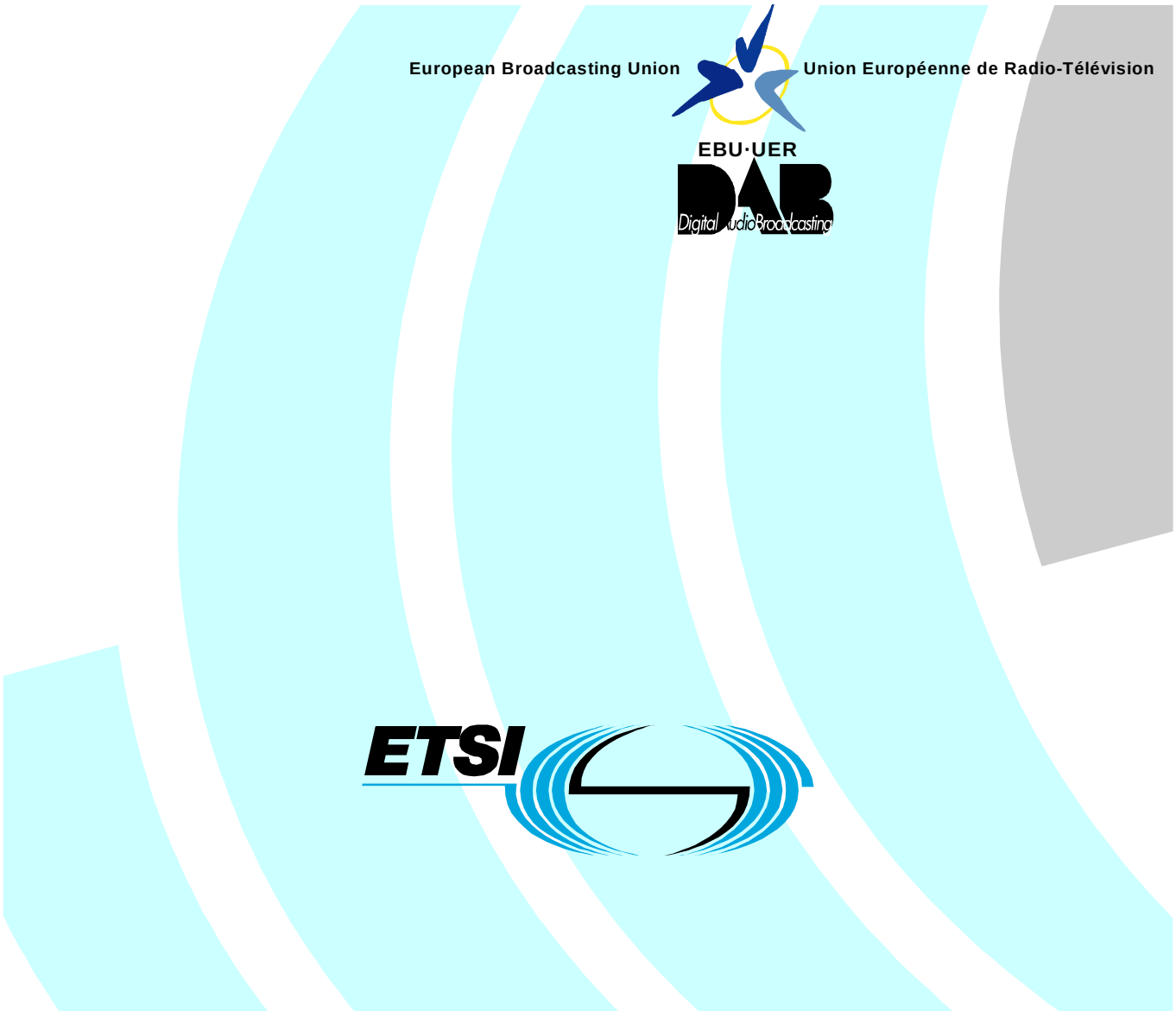
ETSI TS 102 367 V1.2.1 (2006-01)

Technical Specification

Digital Audio Broadcasting (DAB); Conditional access



Reference
RTS/JTC-DAB-46

Keywords
audio, broadcasting, DAB, data, digital

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE 1: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

The Eureka Project 147 was established in 1987, with funding from the European Commission, to develop a system for the broadcasting of audio and data to fixed, portable or mobile receivers. Their work resulted in the publication of European Standard, EN 300 401 [1], for DAB (see note 2) which now has worldwide acceptance. The members of the Eureka Project 147 were drawn from broadcasting organizations and telecommunication providers together with companies from the professional and consumer electronics industry. In 1995, the European DAB Forum (EuroDAB) was established to pursue the introduction of DAB services in a concerted manner world-wide, and it became the World DAB Forum (World DAB) in 1997.

NOTE 2: DAB is a registered trademark owned by one of the Eureka Project 147 partners.

1 Scope

Conditional Access systems provide DAB with the ability to deliver encrypted services, both audio and data. The present document specifies a standardized framework that defines how to signal and transport encrypted services within the Digital Audio Broadcasting (DAB) system. The framework is open to the integration of several different Conditional Access systems as well as to the integration of Shared Scrambler Systems.

2 References

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

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

[1] ETSI EN 300 401: "Radio Broadcasting Systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers".

[2] ETSI EN 301 234: "Digital Audio Broadcasting (DAB); Multimedia Object Transfer (MOT) protocol".

3 Definitions, abbreviations and conventions

3.1 Definitions

For the purposes of the present document, the terms and definitions given in EN 300 401 [1] and the following apply:

Conditional Access (CA): mechanism by which the user access to service components can be restricted

Control Word (CW): Key or part of the key that is used to encrypt and decrypt the content

PrefixDataField: body of Sub-channel Conditional Access Prefix

PrefixHeader: Header of Sub-channel Conditional Access Prefix

Shared Scrambler System (SSS): provides synchronization of the different CA Systems where one service is provided by more than one CA provider with common content scrambling

NOTE: Apart from the CA System decoder module, receivers contain a Common Descrambler.

3.2 Abbreviations

For the purposes of the present document, the abbreviations defined in EN 300 401 [1], EN 301 234 [2] and the following apply:

BWS	Broadcast WebSite
CA	Conditional Access
CACC	Conditional Access Communication Controller
CAFlag	Conditional Access Flag
CAId	Conditional Access Identifier

CAIndi	Conditional Access Indicator field
CAIntChar	Conditional Access system Internal Characteristics
CAIntMess	Conditional Access system Internal Messages
CAIntMessField	Conditional Access system Internal Messages Field
CAMode	Conditional Access Mode
CAOrg	Conditional Access Organization
CAOrgFlag	Conditional Access Organization Flag
CAOrgIndi	Conditional Access Organization Indicator field
CASyncParam	Conditional Access Synchronization Parameters
CASysId	Conditional Access System Identifier
CASysIdList	List of Conditional Access System Identifiers
Ch	Channel
CW	Control Word
CWT	Control Word Toggle bit
DGCAPrefix	Data Group Conditional Access Prefix
EEP	Equal Error Protection
FIDC	Fast Information Data Channel
GSM	Global System for Mobile communication
IP Tunnel.	Internet Protocol Tunnelling
LSb	Least Significant bit
MOT DirMod	MOT Directory Mode
MOT HdMode	MOT Header Mode
MOTCAPrefix	MOT Conditional Access Prefix
MPEG2-TS	MPEG2 Transportation Stream
MSb	Most Significant bit
MSC	Main Service Channel
PAD	Programme Associated Data
SharedFlag	Shared scrambler Flag field
ShortCASysId	Short Conditional Access System Identifier
SLS	SLide Show
SSS	Shared Scrambler System
SUBCAPrefix	SUB-channel Conditional Access Prefix
TDC	Transparent Data Channel

3.3 Conventions

Unless otherwise stated, the following notation, regarding the order of bits within each step of processing is used:

- in figures, the bit shown in the left hand position is considered to be first;
- in tables, the bit shown in the left hand position is considered to be first;
- in byte fields, the Most Significant bit (MSb) is considered to be first and denoted by the higher number. For example, the MSb of a single byte is denoted "b_7" and the Least Significant bit (LSb) is denoted "b_0";
- in vectors (mathematical expressions), the element with the lowest index is considered to be first.

NOTE: Due to time-interleaving, this order of bits is not the true transmission order.

A black triangle within figures indicates that the corresponding element is scrambled.

4 General description

4.1 DAB transport protocols and DAB transport channels

This clause gives an overview of the existing DAB transport protocols and DAB transport channels.

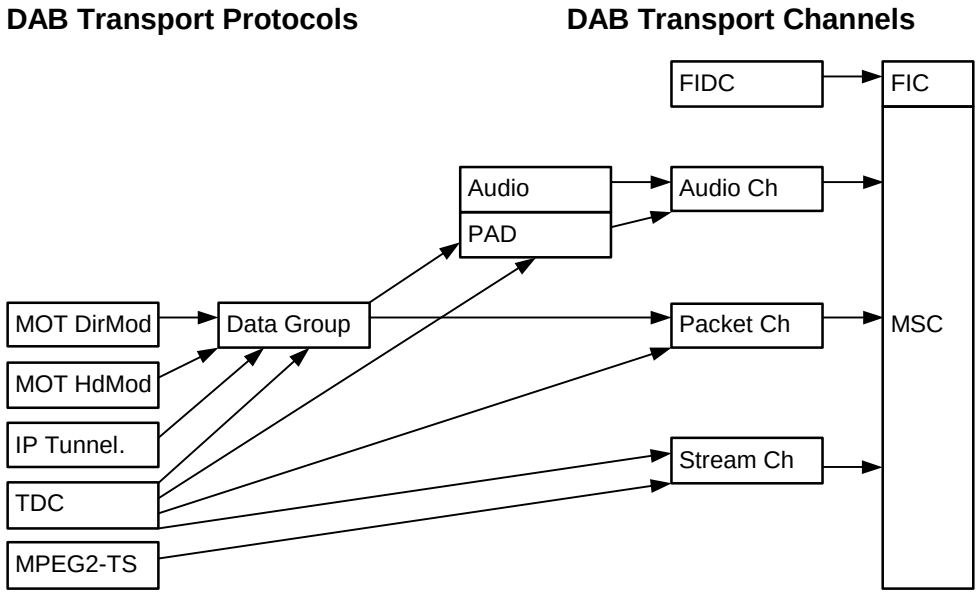


Figure 1: Overview of DAB transport protocols and DAB transport channels

4.2 Scrambling Modes

Conditional Access Systems might be applied on different levels. This clause describes the three DAB transport levels that are available for Conditional Access and illustrates the impact on the upper transport levels, marked with black triangles in figures 2, 3 and 4.

4.2.1 Scrambling DAB sub-channels

Sub-channel CA: Encrypts a complete sub-channel, for example an audio sub-channel including PAD or an entire packet mode sub-channel or a stream mode sub-channel.

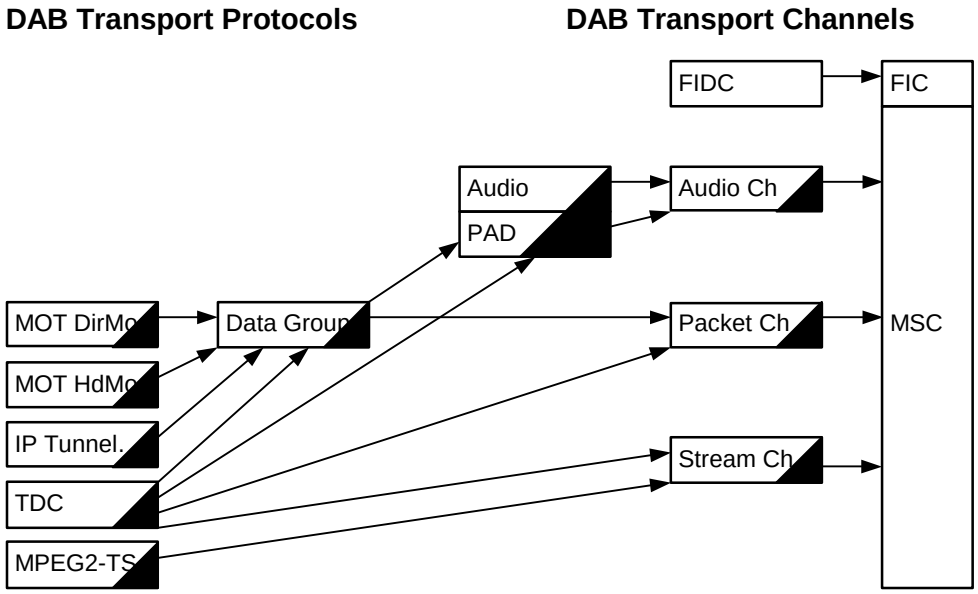


Figure 2: Application and impact of Sub-channel CA

4.2.2 Scrambling DAB Data Groups

Data Group CA: Permits the encryption of all DAB data transfer protocols that use MSC Data Groups, like IP Tunnelling, MOT, TDC, etc.

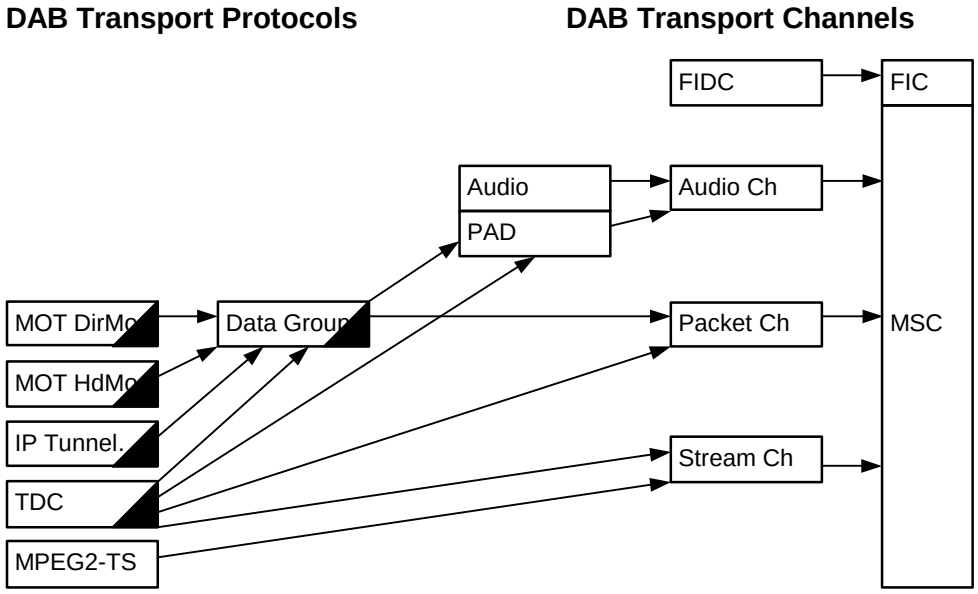


Figure 3: Application and impact of Data Group CA

4.2.3 Scrambling MOT Objects

MOT CA: Permits the encryption of files carried using MOT directory mode, for instance selected parts of a Broadcast WebSite (BWS).

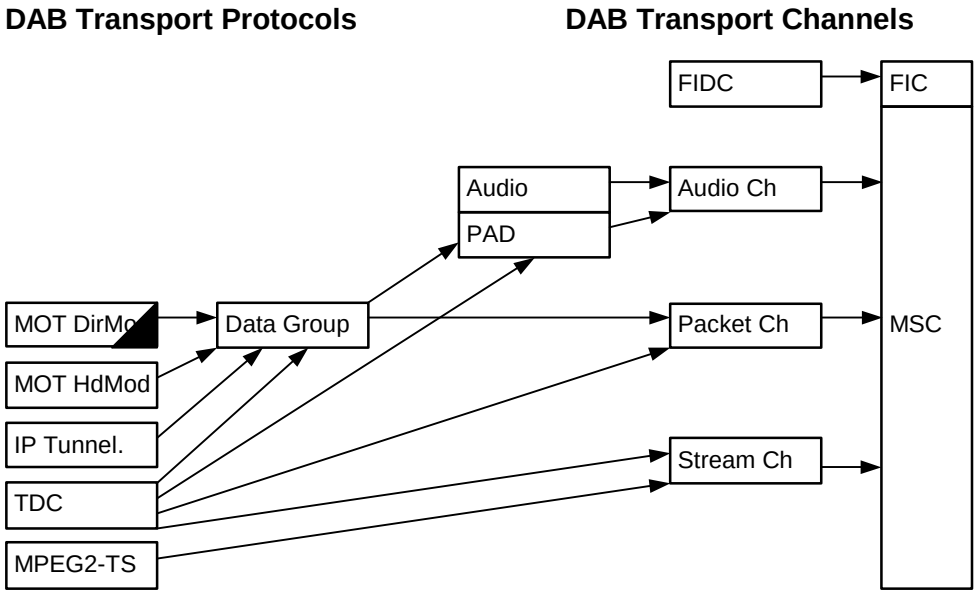


Figure 4: Application and impact of MOT CA

4.3 Concept of a Conditional Access System

The components of a Conditional Access System can be separated into "transmitter side CA system components" and "receiver side CA system components".

Transmitter side CA system components: CA Message Encoder, Control Word Generator, Synchronizer and Scrambler.
These components are provided with subscriber data, CA provider data and unscrambled content as input.

- **CA Message Encoder:** The CA Message Encoder generates messages that might for instance be management messages that concern the entitlements of a user, or messages that contain the current control word and activate an entitlement check on the receiver side. These messages and message formats are CA system specific; they differ from CA system to CA system. In the following they are called CA System Internal Messages (CAIntMess) and will not be detailed.
  The CA System Internal Messages can be carried in the multiplex.

NOTE: In hybrid systems CA system Internal Messages may also be carried in other channels (e.g. GSM).

- **Control Word Generator:** The Control Word Generator provides the control words that are needed by the Scrambler as well as by the CA Message Encoder and passes them to the Synchronizer.

- **Synchronizer:** The Synchronizer synchronizes the process of scrambling and control word dispatching and passes therefore the control words to the Scrambler and Message Encoder components, and generates in addition CA Synchronization Parameters (CASyncParam) that enable a receiver to decode the messages and descramble the content synchronously.
  The CA Synchronization Parameters form a part of the multiplex.

- **Scrambler:** The Scrambler scrambles the incoming content with the available control word.
  The scrambled content forms part of the multiplex.

Receiver side CA system components: CA Message Decoder, Synchronizer, Descrambler.

- **CA Message Decoder:** Interprets the CA System Internal Messages and starts the corresponding processes of entitlement management, entitlement checking and control word handling.

- **Synchronizer:** Interprets the CA Synchronization Parameters and passes the control word synchronously to the Descrambler.

- **Descrambler:** Descrambles the scrambled content using the control word.

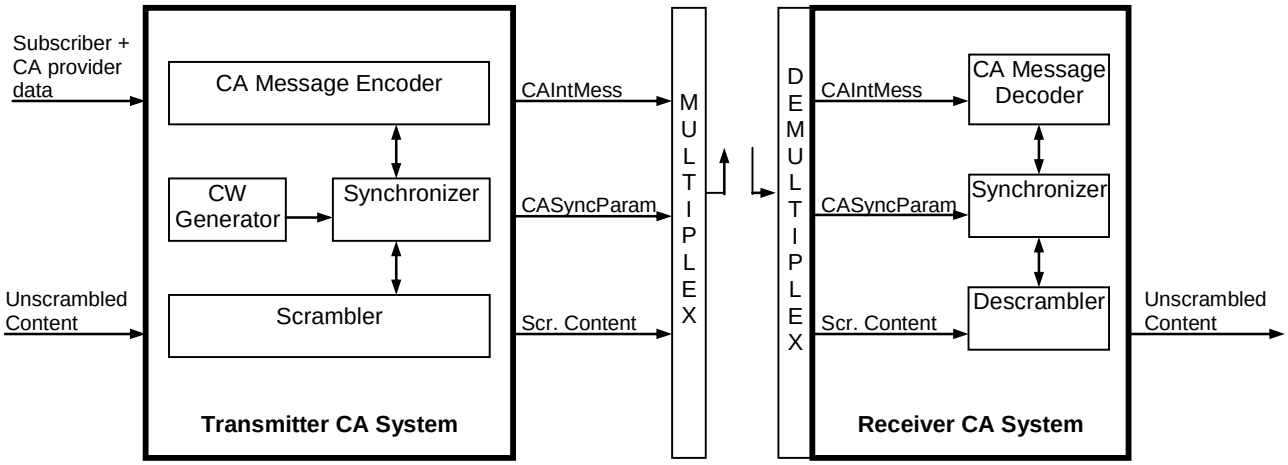


Figure 5: Concept of a Conditional Access System

4.4 Concept of a Shared Scrambler System (SSS)

Where there is more than one CA provider for the same service it should be feasible to transmit the content only once but at the same time to apply different CA Systems.

Therefore the CA providers may cooperate and scramble the content commonly using a shared scrambler. Each CA System has to generate its own CA System Internal Messages (CAIntMess) that have to be synchronized with the scrambling process. A terminal is only capable of interpreting the CA System Internal Messages of "its" CA system but it has a common descrambling module integrated to descramble the content.

The design of a Shared Scrambler System (SSS) is not within the scope of the present document. But the concepts are open to allow the attachment of an existing Shared Scrambler System on the transmitter side.

NOTE: A Shared Scrambler System (SSS) may be applied if a system update within one CA System is planned and both the old and new version should operate in parallel during a certain period.

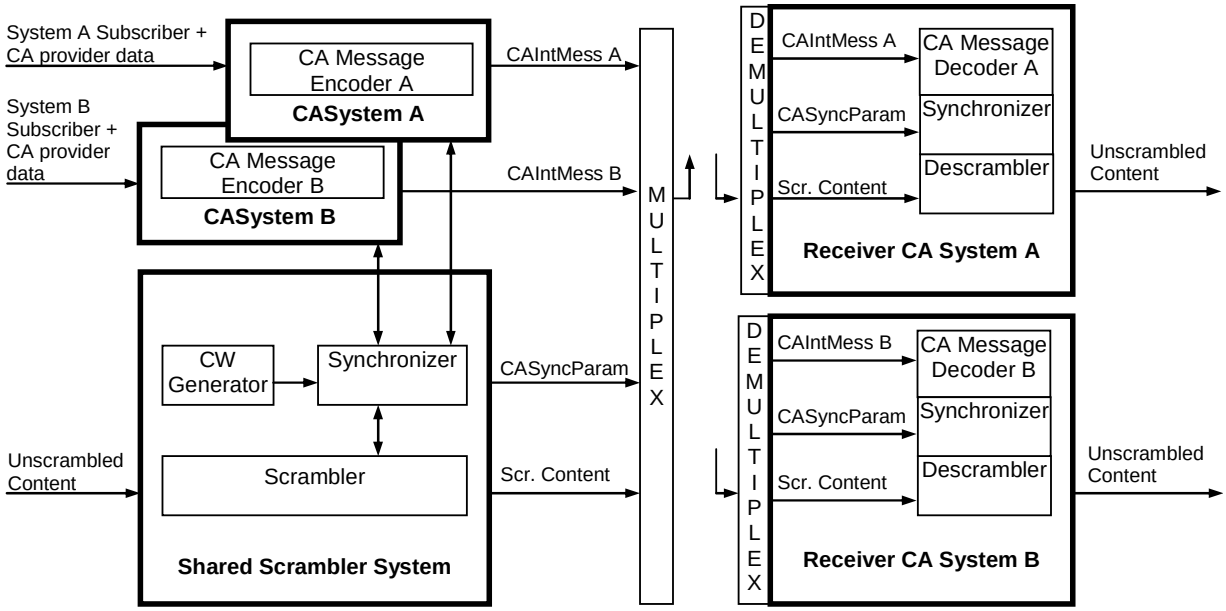


Figure 6: Concept of a Shared Scrambler System

The shared scrambler concept requires for each CAIntMess an indicator to determine to which CA System it belongs. This will be done with the Short CA System Identifier (ShortCASysId).
The shared scrambler concept requires for the scrambled content an indicator to determine all the CA Systems that have shared the scrambler. This will be done with the Shared Scrambler Flag Field (SharedFlag).

5 Parameters: format, coding and location

Several parameters are specified for signalling to the receiver whether the data is scrambled or not, which CA System and CA Mode is applied, where the CA System Internal Messages and additional internal characteristics are found, and how the descrambler is to be synchronized. These parameters are described below.

5.1 CA Identifier (CAId)

The CA Identifier (CAId) is signalled using FIG 0/2 (see [1] clause 6.3.1).

This 3-bit field shall indicate whether a Conditional Access system is used for any of the service components of a service. The interpretation of this field is as follows:

000: no access control for any components of the service.

001: reserved

010: reserved

011: reserved

100: reserved

101: reserved

110: reserved

111: at least one component of the service is scrambled. Scrambled components are signalled according to the present document.

NOTE: The CAId field can be used by a receiver to determine which version of the present document is in use by the service provider. The value "111" indicates to a receiver that the current version of the present document is used. The values "001" and "010" indicate that V1.1.1 of the present document is used.

5.2 CA System Identifier List (CASysIdList)

The CA System Identifier List (CASysIdList) describes the applied CA Systems; it contains identifiers and additional characteristics of the currently used CA systems.

Each list element of the CASysIdList is transported in one FIG type 6 data field. Therefore the length of one list element shall not exceed 29 bytes. The maximum length of the field CAIntChar is therefore defined by this limit. The length is indicated within the FIG header (see EN 300 401 [1], clause 5.2.2).

Parameters transported within the CASysIdList are of a static nature.

Each list element of the CASysIdList contains the identifier and the characteristics of one currently used CA system. Its format is shown in figure 7.

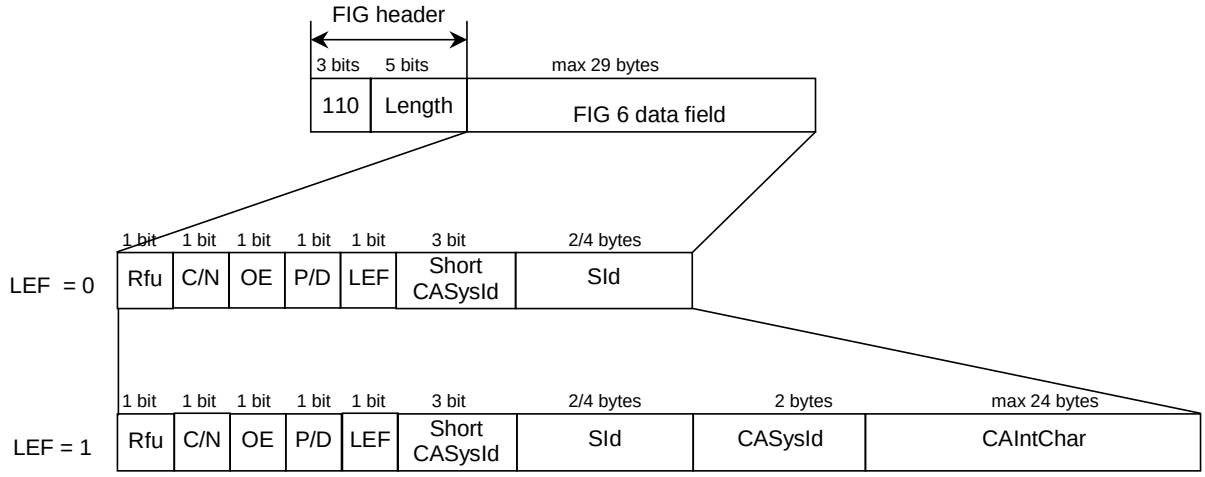


Figure 7: Structure of the FIG type 6 data field

The following definitions apply:

Rfu (Reserved for future use): this 1-bit field shall be reserved for future use of the remainder of the structure. The Rfu bit shall be set to zero for the currently specified definition.

C/N (Current/Next): this 1-bit field shall indicate the service information version (SIV), see EN 300 401 [1], clause 5.2.2.1, situation (b).

OE (Other Ensembles): this 1-bit flag shall indicate whether the information is related to this or another ensemble, see EN 300 401 [1], clause 5.2.2.1.

P/D: this 1-bit flag shall indicate whether the Service Identifier (SId) field is used for Programme services or Data services, see EN 300 401 [1], clause 5.2.2.1.

LEF (List Element Flag): this 1-bit flag shall indicate whether a database change is signalled (CEI) or an element of CASysIdList is contained, as follows:

0: database change, ShortCASysId shall be set to 000;

1: List element of CASysIdList present.

ShortCASysId: this 3-bit field shall carry the Short CA System Identifier, see clause 5.2.2.

SId (Service Identifier): this 2-byte or 4-byte field shall identify the service. Its length shall be signalled by the P/D flag.

CASysId: this 2-byte field shall identify the CA System, see clause 5.2.1.

CAIntChar: this field, of maximum length 24 bytes, shall carry the CA System Internal Characteristics, see clause 5.2.3.

This feature shall use the SIV signalling (see EN 300 401 [1], clause 5.2.2.1). The database shall be divided by use of a database key. Changes to the database shall be signalled using the CEI.

The database key comprises the **OE** and **P/D** flags and the **SId** field.

The change event indication (CEI) is signalled by the List Element Flag (**LEF**) = 0.

5.2.1 CA System Identifier (CASysId)

Every CA System is identified via a CA System Identifier (CASysId).

Format and Coding: The length is 2 bytes. The list of currently registered CA System Identifiers is given in annex F.

Location: The identifier of every CA System currently in use on the multiplex is transported within a list element of the CA System Identifier List (CASysIdList).

5.2.2 Short CA System Identifier (ShortCASysId)

For internal handling, the CA System Identifiers (CASysId) of the currently used CA systems and their parameters are mapped to temporary Short CA System Identifiers (ShortCASysId). These are then used to indicate the CA System to which each CAIntMess belongs while a Shared Scrambler System (SSS) is applied.

A ShortCASysId is unique and valid within one service. The applied CA system may differ from service component to service component.

NOTE: A scrambled component is accompanied by a flag field (SharedFlag) that identifies the CA System(s) applied, (see clause 5.4.2).

If multiple services share a service component then the mapping of CASysId to ShortCASysID shall be the same in all these services for all ShortCASysIds used within the shared service component.

Format and Coding: The length is 3 bits, so up to 8 different CA systems within one service can be specified.

Location: The Short CA System Identifier (ShortCASysId) of every CA System currently in use on the multiplex is transported within a list element of the CA System Identifier List (CASysIdList).
To associate a CAIntMess to a certain CA System, the Short CA System Identifier (ShortCASysId) is carried in the first bits of the CAIntMess.

5.2.3 CA System Internal Characteristics (CAIntChar)

CA System Internal Characteristics (CAIntChar) could be information like version, applied algorithm, system specific parameters, channel identifier, length indicator, content dedicated parameter, static prefixes, etc.

Format and Coding: The maximum length is 24 bytes. The organization of this field is a feature of the corresponding CA system and is not standardized in the present document.

Location: The CA System Internal Characteristics of every CA System currently in use on the multiplex are transported within a list element of the CA System Identifier List (CASysIdList).

5.3 CA Indication - CAFlag/CAIndi

Scrambled content or other CA related content is indicated either with a set CA Flag (CAFlag) or implicitly with the existence of a CA Indicator Field (CAIndi). The CA indication is mainly directed to non CA capable terminals, which should interpret it in the following way.

Format and Coding:

- **CA not indicated** - CAFlag is not set or CAIndi is not present:
  Part of the content or the whole of content is unscrambled.
  Part of the content may be scrambled.
  It makes sense for a non CA capable terminal to process the unscrambled content parts.
  A CA capable terminal will process the content anyway (unscrambled and scrambled content parts).

- **CA indicated** - CAFlag is set or CAIndi is present:
  The corresponding content is scrambled entirely.
  A non CA capable terminal shall not process it.
  A CA capable terminal will process the content.

Location: CA is indicated on different levels. Its location is detailed in the corresponding clauses.

5.4 CA Organization (CAOrg)

The CA Organization (CAOrg) contains generic information about how CA is applied. It contains the Scrambling Mode (CAMode) and the Shared Scrambler Flag Field (SharedFlag).

Format and Coding: The CAOrg field has a length of 16 bits; the following format is used:

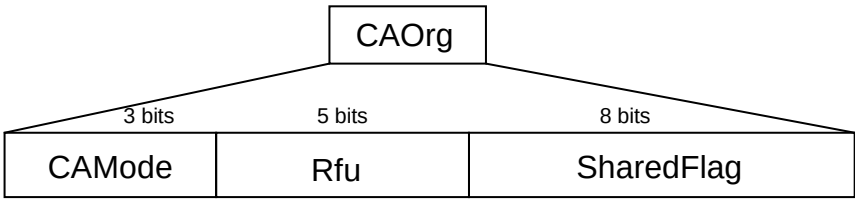


Figure 8: Format of the field CA Organization (CAOrg)

Location: The CA Organization might be transported in different locations as detailed in the corresponding clauses.

5.4.1 Conditional Access Mode (CAMode)

The currently applied scrambling mode is called CAMode.

Format and Coding: Its length is 3 bits.

000: Sub-channel CA

001: Data Group CA

010: MOT CA

011: proprietary CA

100: reserved

101: reserved

110: reserved

111: reserved

Location: It is transported within the CAOrg field.

5.4.2 Shared Scrambler Flag Field (SharedFlag)

The Shared Scrambler Flag Field (SharedFlag) refers to the scrambled content. It indicates the CA Systems that can be used to descramble the content.

Format and Coding: The Shared Scrambler Flag Field has a length of 8 bits. Therefore it can flag up to 8 different CA systems working in parallel within one service.

No flag is set:	Invalid
One flag is set:	No Shared Scrambler System (SSS) is applied; the corresponding CA System can be used to descramble the content.
Two to eight flags are set:	SSS is applied; two to eight different CA systems provide the scrambled content in common and any of them can be used to descramble the content.

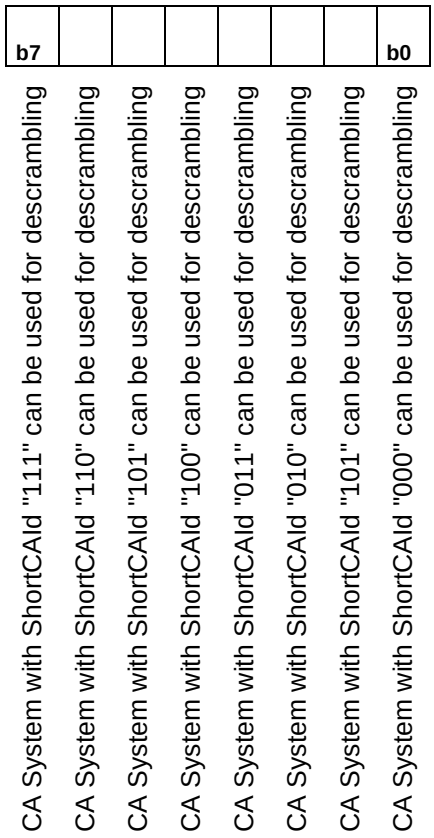


Figure 9: Coding of Shared Scrambler Flag Field

Location: It is transported within the CAOrg field.

5.5 CA Organization Indication - CAOrgFlag / CAOrgIndi

The CA Organization (CAOrg) contains generic information about the applied scrambling mode and CA Systems as described above. Its existence is either indicated with its own CAOrgFlag or it is indicated implicitly by the existence of a CA Organization Indicator Field (CAOrgIndi).

Format and Coding:

Each combination of the CA Indicator and the CA Organization Indicator has a meaning as listed below:

CA Indication CAFlag / CAIndi	**CA Organization Indication CAOrgFlag / CAOrgIndi**	
0 / not present	0 / not present	Nothing is scrambled
0 / not present	1 / present	Part of the corresponding content is scrambled
1 / present	0 / not present	Invalid
1 / present	1 / present	Corresponding content is scrambled entirely

Location: The location of the CA Organization Indication is detailed in the corresponding clauses.

5.6 CA Synchronization Parameters (CASyncParam)

In addition to the parameters introduced above that describe the applied CA systems and the applied CA mode, parameters to synchronize the descrambler are needed. They are CA System dependent and called CA Synchronization Parameters (CASyncParam).

The minimum might be a toggle flag that indicates a control word change. But other parameters like frame counter, initialization modifier etc. might be used. Annex E gives an overview of possible synchronization parameters.

Format and Coding: Their format and coding are not standardized in the present document.

Location: Depending on the different scrambling modes the CA Synchronization Parameters (CASyncParam) are transported in different locations as detailed in the corresponding clauses.

5.7 CA System Internal Messages (CAIntMess)

CA system specific messages that differ from CA system to CA system are called CA System Internal Messages (CAIntMess). Those messages might contain management information that concerns the entitlements of a user and transport keys, or they might contain the current control word and activate an entitlement check on the receiver side. Every transmitted CAIntMess starts with a ShortCASysId as first three bits. This makes a unique assignment to a CA System and therefore the application of a SSS possible.

Format and Coding: Their format and coding are not standardized in the present document.

Location: Depending on the different scrambling modes the CA System Internal Messages (CAIntMess) are transported in different locations as detailed in the corresponding clauses.

5.8 Overview of the parameter location

Table 1 gives an overview of where in the DAB multiplex the parameters introduced above will be placed in combination with the three CA Modes: Sub-channel CA, Data Group CA and MOT CA.

More detailed information of the coding and a full description of the CA Modes will be given in the following clauses. Examples for the parameter settings in the DAB multiplex are given in annex B.

Table 1: Overview of the location of the parameters described in this clause for each of the three scrambling modes: Sub-channel CA, Data Group CA and MOT CA

<table>
<tr><th></th><th colspan="3">Sub-channel CA</th><th colspan="2">Data Group CA</th><th colspan="2">MOT CA</th></tr>
<tr><th></th><th colspan="3"></th><th>Packet mode</th><th>PAD</th><th>Selected MOT Objects are scrambled</th><th>All MOT objects of an MOT Data Carousel are scrambled (see note)</th></tr>
<tr><td>CAId</td><td colspan="3">FIG 0/2</td><td>FIG 0/2</td><td>-</td><td>FIG 0/2 / -</td><td>FIG 0/2 / -</td></tr>
<tr><td>CASysIdList</td><td colspan="3">FIG 6</td><td>FIG 6</td><td>FIG 6</td><td>FIG 6</td><td>FIG 6</td></tr>
<tr><td rowspan="2">CA Indication</td><td colspan="3">CAFlag in FIG 0/2</td><td rowspan="2">CAFlag in FIG 0/2</td><td rowspan="2">CAFlag in FIG 0/13</td><td colspan="2">CAIndi: existence of MOT header parameter CAInfo in MOT directory</td></tr>
<tr><td>Packet Ch</td><td>Stream Ch</td><td>Audio Ch</td><td>CAFlag in FIG0/2 / FIG 0/13</td></tr>
<tr><td>CAOrg Indication</td><td>CAOrgFlag in FIG 0/3</td><td>CAOrgIndi: existence of FIG 0/4</td><td>CAOrgIndi: existence of FIG 0/4</td><td>CAOrgFlag in FIG 0/3</td><td>CAOrgFlag in FIG 0/13</td><td colspan="2">CAOrgIndi: existence of MOT header parameter CAInfo in MOT directory
CAOrgFlag in FIG 0/3 / FIG0/13</td></tr>
<tr><td>CAOrg
"permitted CAMode"</td><td>FIG 0/3
"Sub-channel CA" or "Proprietary CA"</td><td>FIG 0/4
"Sub-channel CA" or "Proprietary CA"</td><td>FIG 0/4
"Sub-channel CA" or "Proprietary CA"</td><td>FIG 0/3
"Data Group CA" or "Proprietary CA"</td><td>FIG 0/13
"Data Group CA" or "Proprietary CA"</td><td colspan="2">MOT header parameter CAInfo in MOT directory "MOT CA" or "Proprietary CA"
FIG 0/3 / FIG0/13
"MOT CA" or "Proprietary CA"</td></tr>
<tr><td>CASyncParam</td><td colspan="3">SUBCAPrefix</td><td colspan="2">DGCAPrefix</td><td colspan="2">MOTCAPrefix</td></tr>
<tr><td>CAIntMess</td><td colspan="3">SUBCAPrefix</td><td colspan="2">MSC Data Group type 1</td><td colspan="2">MOT body or MSC Data Group type 1</td></tr>
<tr><td colspan="8">NOTE: In the case where all MOT objects of an MOT Data Carousel (see EN 301 234 [2]) are scrambled it does not make sense for a non-CA capable terminal to rebuild the data carousel. This case can additionally be signalled on the same level as Data Group CA as follows:
o CAFlag in FIG0/2 or FIG 0/13 is set and
o CAOrgFlag in FIG0/3 or FIG 0/13 is set and
o CAOrg in FIG0/3 or FIG 0/13 contains a copy of the CAOrg transported in the MOT header CAInfo field in the MOT directory.</td></tr>
</table>

6 Sub-channel CA

Sub-channel CA provides the most universal scrambling mode and covers a large number of applications. Because it is located on transport level, it scrambles a complete MSC sub-channel (e.g. it is not possible to scramble PAD data without scrambling the corresponding audio programme).

As input stream it gets either:

- an Audio Sub-channel possibly including PAD data;
- a Packet Mode Sub-channel transporting Data Services in Packet mode; or
- a Stream Mode Sub-channel transporting Stream Mode Data.

NOTE: It seems to be reasonable to use Equal Error Protection (EEP) not only for packet and stream mode channels but also for audio sub-channels when Sub-channel CA is applied. This may cause an additional overhead.

6.1 Location of the CA System

The Location of a CA System that performs Sub-channel CA is shown in figure 10.

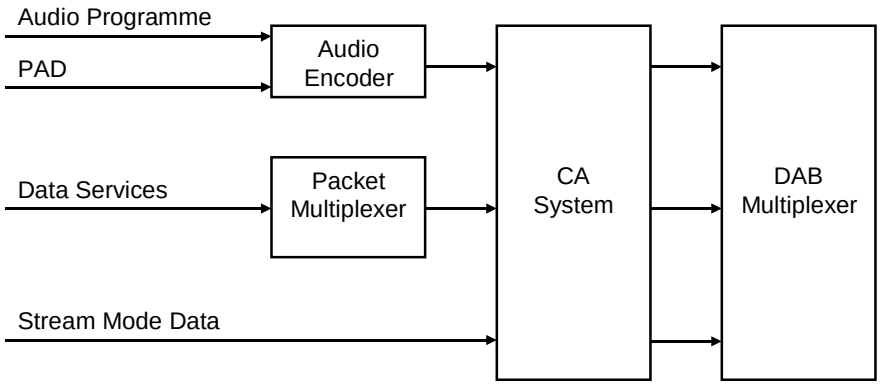


Figure 10: Location of the CA System for Sub-channel CA

6.2 Signalling of CA

The signalling of sub-channel CA is performed according to the concept described above. The signalling parameters are transported as follows:

CAId: see clause 5.1.

CASysIdList: see clause 5.2.

CAFlag: see clause 5.3.

In the "Service component definition" FIG 0/2 (see EN 300 401 [1]; clause 6.3.1) the CAFlag is set for each of the service components involved.

CAOrg Indicator, CAOrg: see clauses 5.4 and 5.5.

- **Packet mode sub-channel:** If the input stream is a packet mode sub-channel, then the CAOrgFlag and the CAOrg are transported in the "Service component in packet mode with or without Conditional Access" FIG 0/3 (see EN 300 401 [1]; clause 6.3.2).
- **Audio sub-channel or stream mode sub-channel:** If the input stream is an audio sub-channel or stream mode sub-channel, then the CAOrg is indicated implicitly by the existence of the CAOrgIndi: FIG 0/4. The CAOrg is transported in the "Service component with Conditional Access in stream mode" FIG 0/4 (see EN 300 401 [1]; clause 6.3.3).

CAMode: For the different transport mechanisms that are defined with the parameter TMId in the "Service component description" FIG 0/2 (see EN 300 401 [1]; clause 6.3.1) the permitted CA Modes are listed below.

Transport mechanism	TMId	CAMode	Meaning
MSC stream audio	00	000	Sub-channel CA
	00	011	proprietary CA
MSC stream data	01	000	Sub-channel CA
	01	011	proprietary CA
FIDC	10	not standardized in the present document	
MSC packet data	11	000	Sub-channel CA
	11	011	proprietary CA

Sub-channel CA does not cover the FIDC. However, a proprietary CA mechanism may be applied to the FIDC.

6.3 Transport of content and of CAIntMess

The CAIntMess are transported in the same sub-channel as the scrambled content. Therefore the CAIntMess and scrambled content can easily be synchronized, which is essential for the announcement of a control word change.

The CAIntMess are transported within a field that prefixes the scrambled content. This field is called the Sub-channel CA Prefix (SubCAPrefix). The length of the SubCAPrefix field is flexible, leading to an increased sub-channel bitrate:

- If the capacity of the sub-channel is a multiple of 8 kbit/sec as it is for a packet mode sub-channel or an audio sub-channel, then it can only be increased by 8 kbps or a multiple of 8 kbps, giving an additional frame length of 24 bytes or a multiple of 24 bytes.

- If a stream mode user application does not require a multiple of 8 kbps, then the prefixed information could be less than a multiple of 8 kbps; provided that the sum of the bitrate for the prefixed information and the scrambled content is a multiple of 8 kbps.

NOTE: For a typical capacity for an audio channel of 160 kbit/sec an addition of 8 kbit/sec leads to a CA overhead of 5 % and to a length of the SUBCAPrefix of 24 bytes. For a typical capacity of a data channel of 64 kbit/sec an addition of 8 kbit/sec leads to a CA overhead of 12,5 % and to a length of the SUBCAPrefix of 24 bytes.

CAIntMess will be partitioned into packets and transported within the SubCAPrefix in consecutive frames. Therefore a packet header has to be defined.

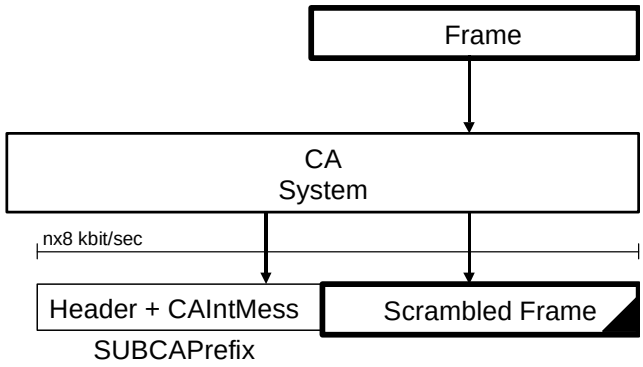


Figure 11: Transport of content and of CAIntMess

6.4 Coding of the SUBCAPrefix

The structure and coding of the SUBCAPrefix is not standardized in the present document. A recommendation is given in annex G. CA providers that apply an SSS shall use a uniform structure of SUBCAPrefix.

7 Data Group CA

Data groups are either transported in a packet mode sub-channel or in PAD.

Data groups transported in a packet mode sub-channel:

Data group CA scrambles:

- either every data group from a service component. That implies that also data groups that contain MOT management data (MOT directory, MOT header) are scrambled.
  A non-CA receiver cannot present anything from the corresponding service component.

- or it scrambles only some data groups of a service component.
  A non-CA receiver can process the unscrambled data groups from the corresponding service component (e.g. within an IP insertion application the video could be scrambled but the audio could be unscrambled).

Data groups transported in PAD:

Data group CA scrambles:

- either all data of a User Application.
  A non-CA receiver cannot present anything from the user application.

- or it scrambles only part of the data of the User Application (e.g. some slides of a slide show (SLS) are scrambled while others remain unscrambled).
  A non-CA receiver can present the unscrambled parts of the user application.

7.1 Location of the CA system

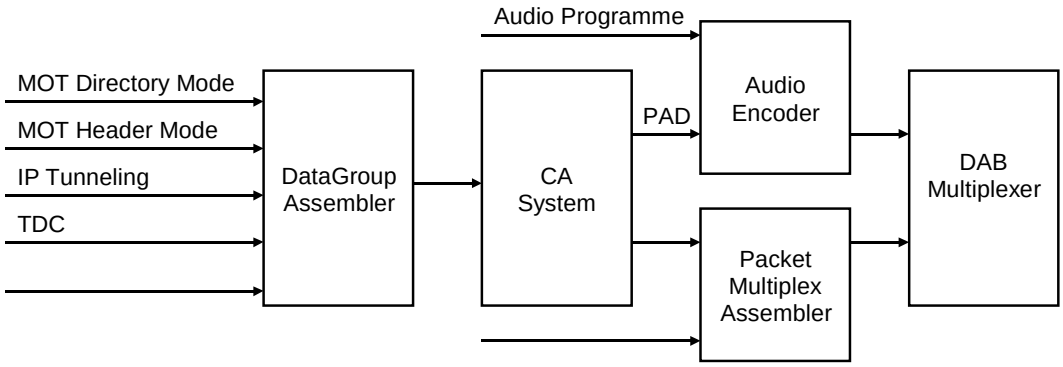


Figure 12: Location of the CA system for Data Group CA

7.2 Signalling of CA

The signalling of Data Group CA is performed according to the above described concept. We have to distinguish between data groups transported in a packet mode sub-channel and data groups transported in PAD. The signalling parameters will be transported in different ways in each case.

7.2.1 Signalling of data groups transported in a packet mode sub-channel

CAId: see clause 5.1.

CASysIdList: see clause 5.2.

CAFlag: see clauses 5.3 and 6.2. The CAFlag is set in the "Service component definition" FIG 0/2 (see EN 300 401 [1]; clause 6.3.1) for each entirely scrambled service component.

CAOrg Indicator, CAOrg: see clauses 5.4, 5.5 and 6.2. The CAOrgFlag and the CAOrg are transported in the "Service component in packet mode with or without Conditional Access" FIG 0/3 (see EN 300 401 [1]; clause 6.3.2).

CAMode: The permitted CAMode shall be as follows:

001: Data group CA

011: proprietary CA

Packet header - Command: As explained in EN 300 401 [1]; clause 5.3.2, a command flag within the packet header indicates the meaning of the packet. It identifies whether the packet is used for unscrambled data groups or for scrambled data groups and data groups containing CA System Internal Messages (CAIntMess) respectively, as follows:

0: Not a CA related packet:
can be processed as usual;

1: CA related packet (scrambled data groups or data groups containing CA System Internal Messages):
must be processed through the CA subsystem.

The command flag is set for packets containing scrambled MSC data groups as well as for packets containing the CAIntMess (MSC data group type 1).

A non-CA terminal ignores packets with a command flag set to 1.

7.2.2 Signalling of data groups transported in PAD

CAId: does not exist. When Data groups are transported in PAD, FIG 0/2 is not in use.

CASysIdList: see clause 5.2.

CAFlag, CAOrgFlag, CAOrg: see clauses 5.3, 5.4 and 5.5.

The CAFlag, CAOrgFlag and CAOrg are transported in the "User application data" field for PAD FIG 0/13 (see EN 300 401 [1]; clause 8.1.20).

CAMode: The permitted CAMode shall be as follows:

001: Data group CA;

011: proprietary CA.

X-PAD Application Type: The offset of the X-PAD Application Type of FIG 0/13 indicates the meaning of the packet. It identifies whether the packet is used for unscrambled data groups or for scrambled data groups and data groups containing CA System Internal Messages (CAIntMess) respectively, as follows:

Offset	Meaning
+0, +1	Not CA related packet: can be processed as usual
+2, +3	CA related packet (scrambled data groups or data groups containing CA System Internal Messages): must pass through the CA subsystem

The offset is +2 or +3 and the CAOrg Field is present for packets containing scrambled MSC data groups as well as for packets containing the CAIntMess (MSC data group type 1).

A non-CA terminal only processes packets indicated with an offset +0 or +1.
A CA terminal processes packets indicated with an offset +0 and +1 as well as packets indicated with an offset +2 or +3.

EXAMPLE: Slideshow:
If the lowest numbered application type used to transport this user application is 12, then unscrambled MOT header as well as the unscrambled MOT body use X-PAD Application Type 12/13. The scrambled MOT header, the scrambled MOT body as well as the CAIntMess use X-PAD Application Type 14/15.

7.3 Transport of content and of CAIntMess

The scrambling of an MSC data group comprises the scrambling of the session header together with the MSC data group data field. The initial MSC data group header remains unscrambled and is supplemented with the Data Group CA Prefix (DGCAPrefix). The new MSC data group CRC, if present, is calculated on the adjusted MSC data group header and the scrambled MSC data group.

The CA System Internal Messages (CAIntMess) and other system internal control and management information are also transported within "MSC Data Groups" (see EN 300 401 [1]; clause 5.3.3.1). Therefore the CAIntMess and scrambled content can easily be synchronized.

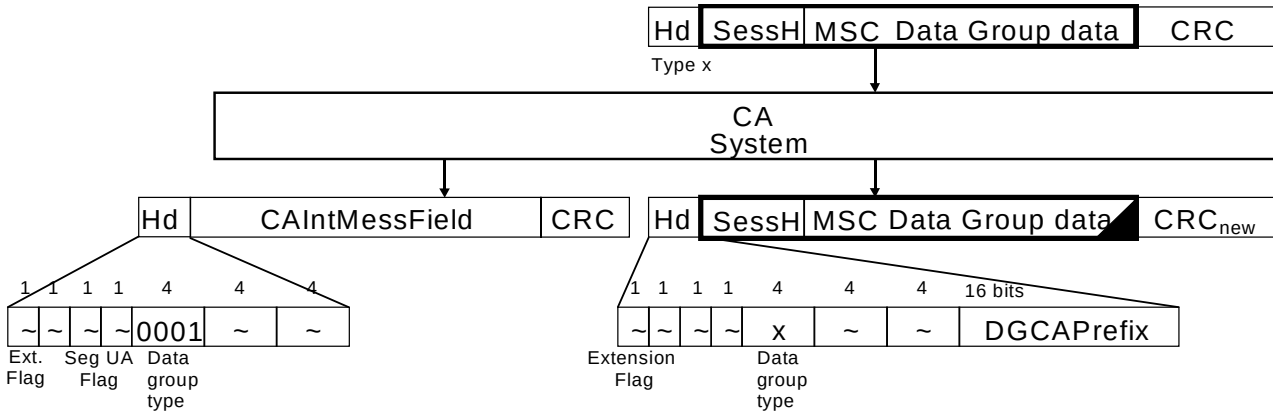


Figure 13: Transport of content and of CAIntMess

7.3.1 Transport of the Content

The MSC Data Group Session Header is scrambled together with the MSC Data Group data.
A field DGCAPrefix prefixes the scrambled content.

MSC data group header: The MSC data group header is defined as follows:

- **Extension Flag:** this 1-bit flag indicates the presence of the DGCAPrefix as follows:

  0: no DGCAPrefix present;

  1: DGCAPrefix present;

- **Data Group Type:** this 4-bit field contains the type of the transported MSC Data Group. The Data Group Type of the scrambled MSC Data Group is the same as the Data Group Type of the unscrambled MSC Data Group.

- **DGCAPrefix:** the use and coding of this 2-byte field is not normative.

The other fields in the MSC data group header are set as specified.

Session header: The MSC Data Group Session Header is scrambled together with the MSC Data Group data.

MSC data group data field: The MSC Data Group data is scrambled together with the MSC Data Group Session Header.

MSC data group CRC (if present)**:** The data group CRC of a scrambled MSC Data Group is calculated on the adjusted MSC data group header and the scrambled MSC data group (session header together with MSC data group data field) and is generated according the procedure defined in EN 300 401 [1], clause 5.3.3.3.

7.3.2 Transport of the CAIntMess

The CAIntMess are transported in the CAIntMessField that is located in an MSC data group data field of an MSC Data Group Type 1.
A CAIntMess can be transported as a whole.

MSC data group header: The MSC data group header is defined a follows:

- **Data Group Type:** this 4-bit field contains the type of the transported MSC Data Group. The Data Group Type is 0001.

The other fields in the MSC data group header are set as specified.

7.4 Coding of the CAIntMessField

The CAIntMessField contains the CA System Internal Messages (CAIntMess). The transport of one entire CAIntMess within one CAIntMessField is recommended.
The further use, structure and coding of the CAIntMessField are CA system specific and may differ from CA system to CA system.

Apart from the CA System Internal Messages (CAIntMess) the following parameters could be placed herein:

ShortCASysId: As explained in clauses 4.4 and 5.2.2: to realize the Shared Scrambler concept each CAIntMess transported in the CAIntMessField starts with a ShortCASysId.

CA Communication Controller: see annex E.

7.5 Coding of the DGCAPrefix

The presence of the DGCAPrefix is optional and indicated with an extension flag set to 1. Its use, structure and coding is not standardized in the present document.

CA Synchronization Parameters CASyncParam (see annex E) could be transported within the DGCAPrefix.

8 MOT CA

MOT CA relates to MOT Directory Mode. It provides the scrambling of some or all MOT objects within a directory structure. The MOT directory remains unscrambled.

A non-CA receiver can still process the unscrambled objects. (E.g.: within a BWS application unscrambled pages could be the entry page or contain information on how to subscribe to the service and some teaser data.)

The information as to which objects are scrambled and which remain unscrambled is contained in the MOT directory. The MOT directory remains unscrambled.

8.1 Location of the CA system

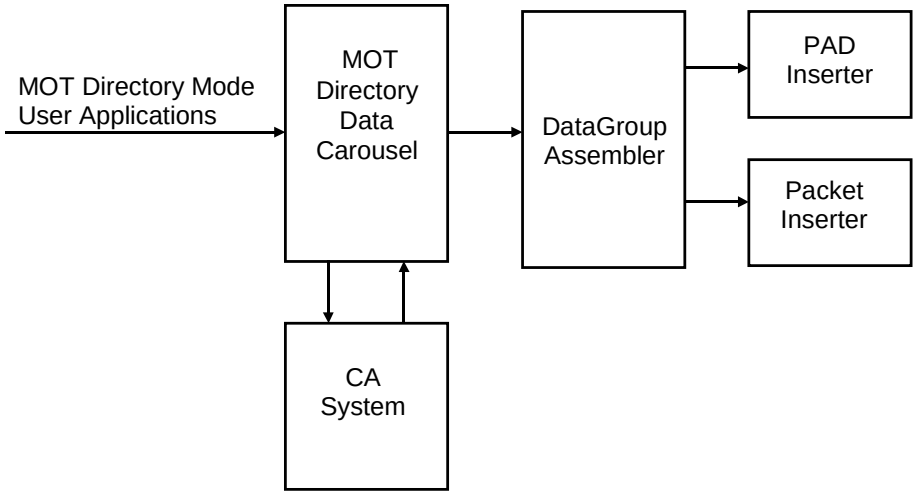


Figure 14: Location of the CA System for MOT CA

8.2 Signalling of CA

The signalling of MOT CA is performed according to the concept described above.

CAId: see clause 5.1.

CASysIdList: see clause 5.2.

CA Indication: No explicit CAFlag is used. CA Indication is given by CAIndi, which in the case of MOT CA is the existence of MOT parameter CAInfo (0x23) that is part of the MOT header information (see EN 301 234 [2]; clause 6.2.2.3.1). This parameter indicates that the body of the corresponding MOT object is scrambled.

Every MOT decoder shall check the existence of the MOT parameter CAInfo.

- A non-CA capable receiver does not have to evaluate the content of the MOT parameter CAInfo and will not process the corresponding MOT object.
- A CA capable receiver shall evaluate the content of the MOT parameter CAInfo to decide whether it is able to continue processing the corresponding MOT object.

CAOrg Indication, CAOrg: No explicit CAOrgFlag is used. CAOrg Indication is given by CAOrgIndi, which in the case of MOT CA is the existence of MOT Parameter CAInfo (0x23) that is part of the MOT header information (see EN 301 234 [2]; clause 6.2.2.3.1). This parameter indicates that CAOrg is present as one of the parameters within the MOT parameter CAInfo:

Every MOT decoder shall check the existence of the MOT parameter CAInfo:

- A non-CA capable receiver does not have to evaluate the content of the MOT parameter CAInfo and will not process the corresponding MOT object.
- A CA capable receiver shall evaluate the content of the MOT parameter CAInfo and interpret the CAOrg field to decide whether it is able to continue processing the corresponding MOT object.

If all MOT objects of an MOT data carousel are scrambled it does not make sense for a non-CA capable terminal to rebuild the MOT data carousel.
This case can additionally be signalled as follows:

- CAFlag in FIG0/2 or FIG 0/13 is set; and
- CAOrgFlag in FIG0/3 or FIG 0/13 is set; and
- CAOrg in FIG0/3 or FIG 0/13 contains a copy of the CAOrg transported in MOT parameter CAInfo field in the MOT directory.

CAMode: The permitted CAMode shall be as follows:

010: MOT CA;

011: proprietary CA.

Identification:

Data groups transported in a packet mode sub-channel:

- Packets containing MSC data groups with scrambled MOT body segments (MSC data group type 5) as well as packets containing the CAIntMess and transported in an MSC data group type 1 are indicated with a set command flag in the packet header.
- The MOT directory, that contains the unscrambled MOT headers and that is transported within MSC data group type 6 or 7 as well as MSC data groups type 4 with unscrambled content are transported in packets where the command flag is set to 0.

A non-CA terminal ignores packets indicated with the command flag being set.

Data groups transported in PAD:

- X-PAD subfields containing MSC data groups with scrambled MOT body segments (MSC data group type 5) as well as X-PAD subfields containing the CAIntMess and transported in an MSC data group type 1 are indicated with an X-PAD Application Type offset of +2 or +3.
- The MOT directory, which contains the unscrambled MOT headers and which is transported within an MSC data group type 6 or 7 as well as other MSC data groups type 4 with unscrambled content are transported in X-PAD subfields where the X-PAD Application Type offset is +0 or +1.

A non-CA terminal processes only X-PAD subfields indicated with an offset +0 or +1.
A CA terminal processes X-PAD subfields indicated with an offset +0 and +1 as well as X-PAD subfields indicated with an offset +2 or +3.

EXAMPLE: Broadcast Website:
If the lowest numbered application type used to transport this user application is 12, then unscrambled MOT directory, as well as the unscrambled MOT bodies use X-PAD Application Type 12/13. The scrambled MOT bodies as well as the CAIntMess use X-PAD Application Type 14/15.

8.3 Transport of content and of CAIntMess

MOT CA relates to MOT Directory Mode. It enables the scrambling of selected MOT bodies within a directory structure.

The information as to which objects are scrambled and which remain unscrambled is contained in the MOT directory. The MOT directory remains unscrambled.

The CAIntMess can either be transported within an MOT body or within the MSC data group data field of an MSC Data group type 1.

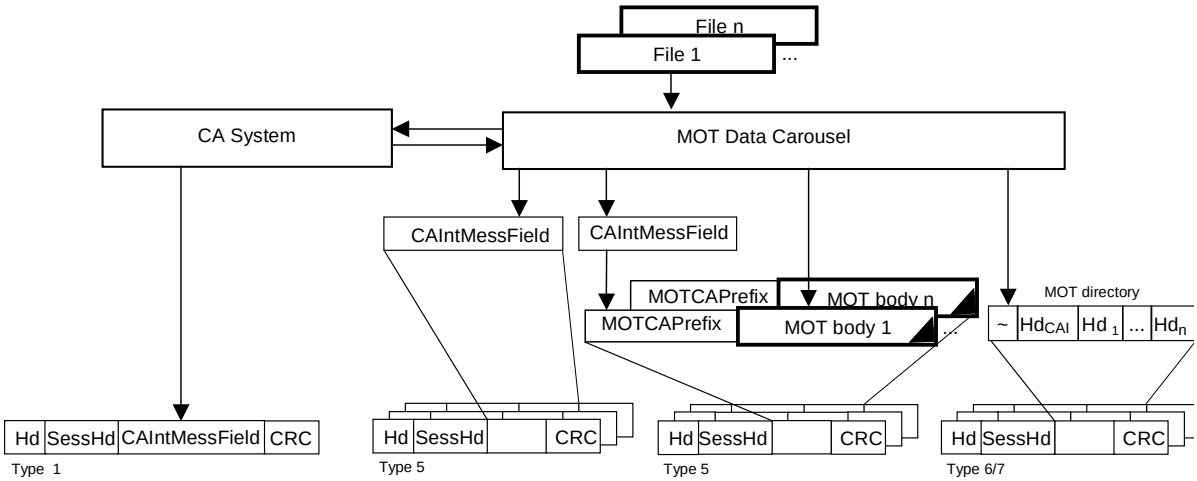


Figure 15: Transport of content and of CAIntMess

8.3.1 Transport of the content

The MOT directory, which also contains the MOT headers, remains unscrambled. It is transported within an MSC Data group type 6 or 7.

The scrambled MOT bodies are transported within MSC data group data fields of an MSC Data group type 5.

Each scrambled MOT body might be preceded with a field MOTCAPrefix. The MOTCAPrefix is then part of the MOT body.

8.3.2 Transport of the CAIntMess

The CAIntMess are transported in the CAIntMessField.

- This field can either be situated within one MSC data group data field of an MSC Data group type 1.
- Or it is situated within an MOT body and will therefore be transported within MSC data group data fields of MSC Data group type 5.
- Or it is situated within the MOTCAPrefix that prefixes the scrambled content and is part of an MOT body, that is transported within MSC data group data fields of MSC Data group type 5.

Which of the three possibilities of CAIntMess transport is chosen depends strongly on the following aspects:

Transport of the CAIntMess in an MSC data group type 1: The MSC data group data field of an MSC Data group type 1 can be used for CAIntMess containing CA management information that is not content related.

Transport of the CAIntMess in an MOT body within MSC data group type 5: A CAIntMess in an MOT body is part of the MOT Directory Data Carousel. Those CAIntMess are considered part of the content.
The MOT header information of the MOT object that contains the CAIntMessField is transported together with the other MOT header information in the MOT directory.

- If the CAIntMess refers to more than one scrambled MOT body, it is transported in an MOT body on its own.
- If the CAIntMess refers to only one scrambled MOT body, it can be transported in MOTCAPrefix.

8.4 Coding of the CAIntMessField

The CAIntMessField contains the CA System Internal Messages (CAIntMess). The transport of one entire CAIntMess within one CAIntMessField is recommended.
Its further use, structure and coding are CA system specific and may differ from CA system to CA system.

Apart from the CA System Internal Messages (CAIntMess) the following parameters could be placed herein:

ShortCASysId: As explained in clauses 4.4 and 5.2.2, to realize the Shared Scrambler concept each CAIntMess transported in the CAIntMessField starts with a ShortCASysId.

CA Communication Controller: see annex E.

8.5 Coding of the MOTCAPrefix

The presence of the MOTCAPrefix is optional. Its use, structure and coding are CA system specific and not standardized in the present document.

CAIntMess referring to one scrambled MOT body, as well as CA Synchronization Parameters CASyncParam (see annex E) could be transported in MOTCAPrefix.

Annex A (informative): Example for Shared Scrambler Concept

Five different CA systems A, B, C, D and E are used within one service. They get the ShortCASysId 0 to 4. The corresponding CASysIdList is shown below. It consists of five list elements.

ShortCASysId	CASysId	CAIntChar
000	CA System A	aaa
001	CA System B	bbb
010	CA System C	ccc
011	CA System D	ddd
100	CA System E	eee

In the example the four service components W, X, Y and Z will be treated in different ways:

- Service component W is scrambled with CA System A (ShortCASysId=0).
- Service component X is not scrambled at all.
- Service component Y is scrambled with System B (ShortCASysId=1).
- Service component Z is scrambled via a Shared Scrambler System, where the CA Systems B (ShortCASysId=1), D (ShortCASysId=3) and E (ShortCASysId=4) operate synchronized.

Each scrambled service component is described by its own CAOrg containing its own SharedFlag:

Service component W: **SharedFlag** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

flags System A (ShortCASysId =0)

Service component X: **No SharedFlag**

Service component Y: **SharedFlag** | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

flags System B (ShortCASysId =1)

Service component Z: **SharedFlag** | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |

flags System B (ShortCASysId =1)
flags System D (ShortCASysId =3)
flags System E (ShortCASysId =4)

Annex B (informative): Examples for Parameter Settings in the DAB Multiplex

The signalling is pointed up in the following nine examples and will be shown in the table below:

EXAMPLE:

1. Scrambled Audio Channel:
   Example for the scrambling of a complete sub-channel and therefore use of Sub-channel CA.

2. Partly scrambled Data Group Application transported in a Packet mode Sub-channel:
   Only some data groups of a service component are scrambled. Data Group CA is applied. This could be an IP insertion application, where the video is scrambled but the audio remains unscrambled.

3. Totally scrambled Data Group Application transported in a Packet mode Sub-channel:
   Every data group from a service component is scrambled. Data Group CA is applied.

4. MOT application where selected MOT objects are scrambled, transported in a Packet mode Sub-channel:
   This could be part of a Broadcast Web Site (BWS). MOT CA is applied.

5. MOT application where all MOT objects of an MOT Data Carousel are scrambled, transported in a Packet mode Sub-channel:
   In this case it does not make sense for a non-CA capable terminal to rebuild the data carousel. CA may therefore additionally be signalled on the same level as Data Group CA.

6. Partly scrambled Data Group Application transported in PAD:
   Part of the data of the User Application is scrambled. Data Group CA is applied. This could be some slides of a slide show (SLS), while others remain unscrambled.

7. Totally scrambled Data Group Application transported in PAD:
   All data of a User Application is scrambled with Data Group CA.

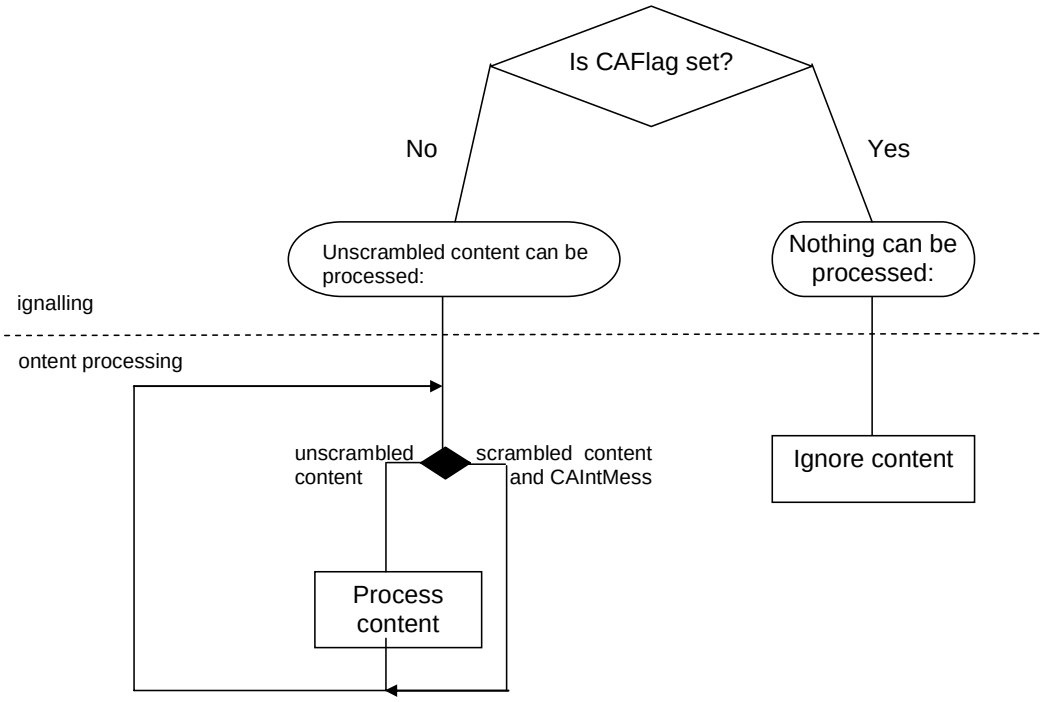
8. MOT application where selected MOT objects are scrambled, transported in PAD:
   This could be part of a Broadcast Web Site (BWS). MOT CA is applied.

9. MOT application where all MOT objects of an MOT Data Carousel are scrambled, transported in PAD:
   In this case it does not make sense for a non-CA capable terminal to rebuild the data carousel. CA may therefore additionally be signalled on the same level as Data Group CA.

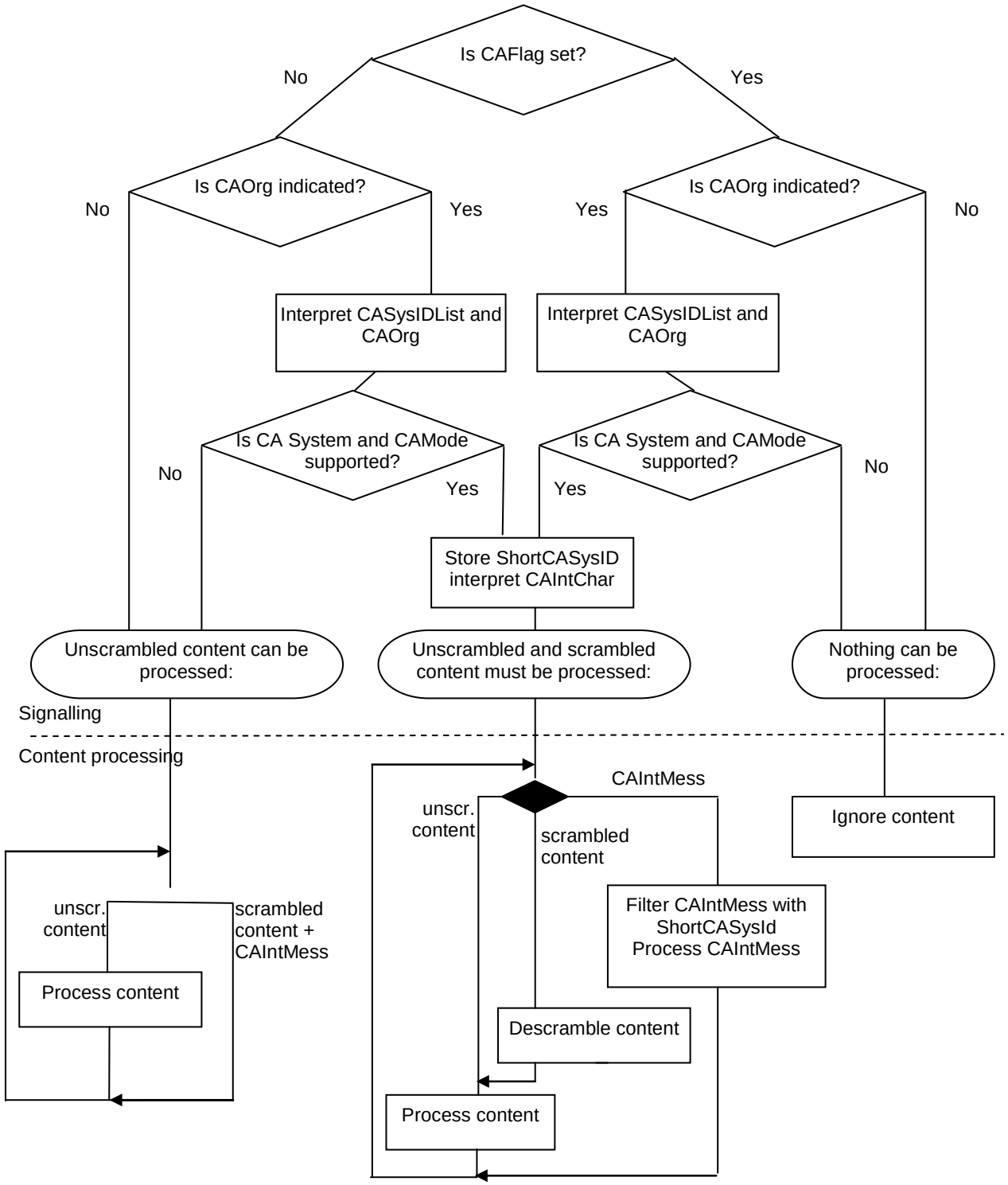
Example		1.	2.	3.	4.	5.	6.	7.	8.	9.
Transported in:		Audio subch.	Packet mode sub-channel				PAD			
Applied CA Mode:		Sub-channel CA	Data Group CA		MOT CA		Data Group CA		MOT CA	
Signalling of a:		scrambled Audio Channel	partly scr. Data Group Applica-tion	totally scr. Data Group Applica-tion	Selec-ted MOT objects are scramb-led	all MOT objects of an MOT Data Carou-sel are scr.	partly scr. Data Group Applica-tion	totally scr. Data Group Applica-tion	selected MOT objects are scramb-led	all MOT objects of an MOT Data Carousel are scr.
Para-meter	Parameter Location	Setting								
CA Id	FIG 0/2	111	111	111	111	111	-	-	-	-
CASys-IdList	Existence of FIG 6	yes	yes	yes	yes	yes	yes	yes	yes	yes
CA Indication	CAFlag in FIG 0/2	1	0	1	0	1*)	-	-	-	-
	CAFlag in FIG 0/13	-	-	-	-	-	0	1	0	1*)
	Existence of MOT header parameter CAInfo in MOT directory	-	-	-	yes	yes	-	-	yes	yes
CAOrg Indication	CAOrgFlag in FIG 0/3	-	1	1	0	1*)	-	-	-	-
	Existence of FIG 0/4	yes	-	-	-	-	-	-	-	-
	CAOrgFlag in FIG 0/13	-	-	-	-	-	1	1	0	1*)
	Existence of MOT header parameter CAInfo in MOT directory	-	-	-	yes	yes	-	-	yes	yes
CAOrg (CAMode)	FIG 0/3	-	"Data Group CA"	"Data Group CA"	-	"MOT CA"*)	-	-	-	-
	FIG 0/4	"Sub-channel CA"	-	-	-	-	-	-	-	-
	FIG 0/13	-	-	-	-	-	"Data Group CA"	"Data Group CA"	-	"MOT CA"*)
	MOT header parameter CAInfo in MOT directory	-	-	-	"MOT CA"	"MOT CA"	-	-	"MOT CA"	"MOT CA"
Additional Parameter Settings:					**Data group type of MOT bodies:** 5				**Data group type of MOT bodies:** 5	
			Packet header - Command flag: 1				**X-PAD Application Type Offset:** +2 or +3			

(-) Stands for "parameter location not present".
*) Recommended optional signalling, it prevents a non-CA capable terminal rebuilding the MOT Data Carouselonly to detect that all objects are scrambled.

Annex C (normative): Behaviour of non-CA capable Terminals



Annex D (normative): Behaviour of CA capable Terminals



Annex E (informative): Synchronization Parameters

In addition to the parameters introduced above that describe the applied CA systems and the applied CA modes, parameters to synchronize the descrambler are needed. They are called CA Synchronization Parameters (CASyncParam).

The minimum might be a toggle flag that indicates a control word change, but other parameters like frame counter, Initialization Modifier etc. might be used.

To give an idea, some possible parameters will be explained as follows. They will not be standardized in the present document. Their location depends on the applied scrambling mode.

Control Word Toggle: This bit signals the currently used control word. Its toggling indicates a control word change.

Control Word Change Countdown: In case of Sub-channel CA the control word change countdown indicates the number of the remaining frames until a control word change will occur. It has been introduced to signal the event of a control word change in advance. The handling and interpretation of the control word change countdown increases the system stability.

Frame Counter: The Frame Counter is incremented with each transmitted frame. It could be useful to keep the descrambler in the terminal synchronous, but its interpretation is optional for the terminal.
The frame counter is incremented with each transmitted and scrambled frame. Its length has to be fixed. An overflow leads to a counter restart. There is no need to transport the frame counter in every frame but as often as it is needed to keep the descrambler in the terminal synchronous.

Initialization Modifier: In addition to a transmitted Control Word some descrambling algorithms need an initialization modifier IM to reinitialize the Pseudo Random Binary Sequence generator and to allow a fast synchronization of scrambler and descrambler.

CA Communication Controller: A multicrypt capable terminal contains a CA Communication Controller (CACC) as a permanent component and it provides the possibility to attach interchangeable modules (e.g. smart cards, SIM cards) of different CA systems. The CA Communication Controller is CA system independent.
Additional CACC message headers must be defined. The CACC headers prefix the CAIntMess, but prefix also the scrambled content. They contain information as to how the terminal should handle the scrambled content or CAIntMess. The CACC interprets these headers.

Annex F (normative): Conditional Access System Identifier (CASysID)

The table contains values for the Conditional Access System Identifier (CASysID). See clause 5.2.1.

CASysId (hexadecimal)	Conditional Access System	Reference
0x8ECA	HECA High Efficient Conditional Access	Fraunhofer IIS http://www.iis.fraunhofer.de/dab/projects/heca/index.html T-Systems

Annex G (informative): Recommended coding of the SUBCAPrefix

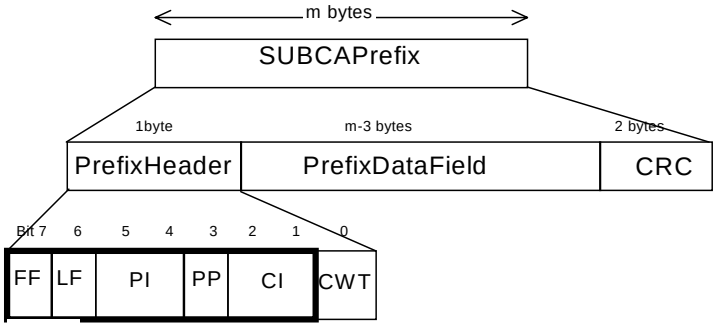


Figure G.1: Coding of the SUBCAPrefix

Length of the SUBCAPrefix: As explained above, the length of the SUBCAPrefix depends on the allocated bitrate available for the CA System Internal Messages. For the following remarks a length of m bytes is assumed.

Structure of the SUBCAPrefix:

- PrefixHeader 1 byte
- PrefixDataField m-3 bytes
- CRC 2 bytes Cyclic Redundancy Check

(Example: If m=24, then a CAIntMess with a length of 32 bytes will be transported within two frames.)

G.1 PrefixHeader

To transport the CAIntMess within the PrefixDataField they have to be split into packets. Bits 1 to 7 of the PrefixHeader contain information about the identifier and position of the packet. With the help of these bits the original messages can be reassembled in the terminal. Bit 0 is toggled to indicate a control word change.

- First flag (FF) 1 bit
- Last flag (LF) 1 bit
- Packet Id (PI) 2 bits
- Padded packet indicator (PP) 1 bit
- Continuity index (CI) 2 bits
- Control word toggle (CWT) 1 bit

First Flag (FF), Last Flag (LF): These flags are used to identify particular packets which form a succession of packets as follows:

FF	LF	This packet ist:
0	0	An intermediate packet
0	1	The last packet of a message
1	0	The first packet of a message
1	1	The one and only packet of a message

Packet ID (PId): This field indicates the packet identifier of a packet. By this means up to four logical transport channels can be realized in parallel (e.g. urgent CAIntMess can interrupt and overtake other messages).

PId	This packet belongs to the:
00	logical transport channel 0
01	logical transport channel 1
10	logical transport channel 2
11	logical transport channel 3

Padded Packet indicator (PP): This bit indicates whether all bytes in the PrefixDataField are used. If for instance the last packet of a succession needs only n bytes of the field (n<256), then the first byte of the PrefixDataField contains the number of used bytes. Its value does not include the count byte itself.

PId	Meaning
00	No padding present: all data bytes in the PrefixDataField are used
01	Padding is present: the first byte, coded as an unsigned integer, in the PrefixDataField gives the number of useful bytes in the PrefixDataField, unused bytes are set to 0x00

NOTE: Since the n is coded as an unsigned 8 bit integer, n must be less than 256. For n=255 the SUBCAPrefix has a length of 259 bytes, which corresponds to more than 80 kbit/sec.

Continuity Index (CI): This 2 bit field is incremented by 1 modulo 4 for each packet with the same packet identifier, making it possible to detect the loss of packets.

Control Word Toggle (CWT): This bit is not linked to the following PrefixDataField but to the Scrambled Frame that follows the SUBCAPrefix. It signals the currently used control word. Its toggling indicates a control word change. CWT is a CA Synchronization Parameter (see annex E).

G.2 PrefixDataField

The PrefixDataField transports the CA System Internal Messages (CAIntMess). Its further use, structure and coding are CA system specific and differ from CA system to CA system.

When the Padded Packet Indicator (PP) is set to 1 in the PrefixHeader, the first byte in the PrefixDataField gives the number of useful bytes in the PrefixDataField; unused bytes are set to 0x00.

Apart from the CA System Internal Messages (CAIntMess) the following parameters could be placed herein:

ShortCASysId: as explained in clauses 4.4 and 5.2.2, to realize the Shared Scrambler concept each CAIntMess transported in the PrefixDataField starts with the three bit parameter ShortCASysId.

Frame Counter: see annex E.
The frame counter helps to keep the descrambler in the terminal synchronous.

Control Word Change Countdown: see annex E.
The control word change countdown indicates the number of remaining frames until a control word change will occur.

Initialization Modifier: see annex E.

CA Communication Controller: see annex E.

G.3 CRC

The 16-bit Cyclic Redundancy Check shall be calculated on the PrefixHeader and the PrefixDataField. It is generated according the procedure defined in EN 300 401 [1], clause 5.3.3.3.

History

Document history		
V1.1.1	January 2005	Publication
V1.2.1	January 2006	Publication