SDMX-EDI

SYNTAX AND DOCUMENTATION

(VERSION 2.0)

November 2005

1 SCOPE AND PURPOSE

The scope of this guide is to document the use of SDMX-EDI; this documentation covers the use of the whole UN/EDIFACT GESMES message and also the way the message should be used in order to serve specific requirements.

The purpose of this guide is to:

explain the SDMX-EDI functions;

define the syntax and rules for the various segments of the Edifact version of the GESMES message

give some guidelines for developing the necessary applications.

2 CONFORMANCE

Chapters 9 and 10 of this document are normative, providing rules for how the SDMX-EDI message must be used in order to be conformant with SDMX. All other chapters and appendixes are non-normative.

3 NORMATIVE REFERENCES

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

The GESMES message is defined in UN/EDIFACT UNTDID D.99B and D.02A (URL: http://www.unece.org/trade/untdid/), UN/ECE

4 BACKGROUND DOCUMENTATION

The reader has to become familiar with the SDMX Information Model.

The exchange of data and metadata, using SDMX-EDI, is based on a set of statistical structure definitions, statistical concepts and code lists from which the coded statistical concepts take their values. These structural definitions are made available by each centre administrating SDMX-EDI data exchanges. For example, in the framework of the data exchanges within the European System of Central Banks (ESCB), the European Central Bank (ECB) has been circulating to its partners SDMX-EDI files containing the *ECB structural definitions (i.e. key family definitions, list of statistical concepts and code lists*). Similarly, the Bank for International Settlements (BIS) has been disseminating to its partners the relevant structural definitions for their data exchanges. User communities of SDMX-EDI (e.g. BIS, ECB, Eurostat and all their partners) have agreed on using a basic set of common code lists. Thus, the exchanged SDMX-EDI messages are expected to look as similar as possible, not only from a syntactical point of view but also, to a great extent, semantically. For example, common code lists are used for the statistical concepts

"units", "unit multiplier", "observation status" (a piece of information "explaining" in a coded format the format of a single observations) and "observation confidentiality". Similarly, the identifier for the concept "frequency" is specified by most central institutions as "FREQ". Institutions using SDMX-EDI have to use this Guide in conjunction with the use of **a set of structural definitions**: either using structural definitions which have been already devised by a centre institution or -if new data flows are concerned- new (or additional) structural definitions need to be devised. When new structural definitions are created, it is strongly recommended first to go through the already existing structural definition files of central institutions in order to avoid creating completely new concepts or new code lists.

Though GESMES and EDIFACT provided the *syntax tools* to "describe" the contents of a message based on the SDMX-EDI data model, it is not necessary for the reader of this document to have a previous GESMES or EDIFACT background or experience. However, in order to fully understand the EDIFACT version of SDMX-EDI, **the knowledge of some basic EDIFACT principles** is a prerequisite: these are explained here in the corresponding Appendix. Readers who are interested in getting a more general and global knowledge of GESMES can refer to the official GESMES documentation (*GESMES Version 2.1: Guidance to Users* and *GESMES Version 2.1: Reference Guide*).

5 STRUCTURE OF THE GUIDE

This User Guide provides a technical reference explaining the features and the rules of SDMX-EDI. The contents are illustrated in several ways:
theory and examples showing the general structure of the message;
theory and examples in a segment-by-segment reference guide chapter; this shows all alternative variants (for each segment) which can appear in a message;
complete examples of messages oriented to practical implementation.

6 SDMX-EDI: THE STATISTICAL DATA AND METADATA EXCHANGE MESSAGE

SDMX-EDI is a message designed for the exchange of statistical information between organisations in a platform independent manner.

The message implements a **data exchange model** (SDMX INFORMATION Model) which provides for the exchange of time series identified through a multi-dimensional key and a variety of associated metadata. It employs an appropriate GESMES profile and, for the version described in this Guide, the EDIFACT syntax. Though GESMES is a generic statistical data model which affords sufficient flexibility to describe syntactically virtually any statistical data model, SDMX-EDI has a fixed syntax. This allows partner institutions to design and to build the applications needed to "read" and "write" SDMX-EDI messages, avoiding intermediate files and special translators; the design of the read/write applications is further simplified by eliminating genericity which is not needed when exchanging time series data. Due to the fixed syntax, in most cases, the rules used in SDMX-EDI are stronger and more restrictive than those in generic GESMES. However, the current design allows the possibility of future enhancements and progressive generalisation, if this is needed, upon agreement of the parties involved.

SDMX-EDI offers several features:

easy adaptation to any economic domain and flexible coverage of all types of economic - statistical data;
modern representation techniques: conceptually 'clean' multidimensional keys;
efficiency: it avoids the unnecessary repetition of information;
series keys without length restrictions;
easy introduction of new attributes;
attributes at various levels (observation, time series and 'higher');
exchange of 'rich' metadata, organised in a flexible and efficient manner;
suitability for two-way exchanging time series: i.e. for reporting and disseminating;

- a *paperless* dissemination of a whole statistical data base is possible: data, metadata, definitions, key structures and code lists can be electronically disseminated from the centre to other institutions;

suitability for stage-by-stage implementation: it could start with the essential parts (carrying the administrative and the numeric data) and later the whole message could be used;
easy implementation: no purchase of special software should be necessary;

- consistency with international standards (EDIFACT);
- wide user group: national central banks (NCBs) and national statistical institutes (NSIs), Bank for International Settlements (BIS), International Monetary Fund (IMF), OECD, EUROSTAT, European Central Bank (ECB);

consistency with the long-term goals of international institutions involved in statistical data exchange.

6.1 Generic GESMES and the UN/EDIFACT Standards

The goal of UN/EDIFACT-standardisation is to give to the exchanged data flow objects a structure whose elements can be understood and processed by software applications without human intervention. The first messages were created for commercial data exchange environments and concerned rather simple and static object types like invoices, purchase orders etc. Since then new sectors, such as statistics, with more complicated and varied data structures have joined the industry and it became evident that certain message types would have to be made generic to satisfy user requirements of those sectors without actually exploding the number of messages.

- GESMES is an acronym meaning Generic Statistical Message. It was developed by a group of European statistical organisations working within the international UN/EDIFACT standards body. GESMES has all the features required to exchange multi-dimensional arrays and time series data, including metadata (such as attributes and footnotes). The advantage of using GESMES, in preference to a proprietary data format, is that it is an internationally agreed standard which is both open and fully functional. It is not tied to the format and constraints of one particular application. In particular GESMES supports the exchange of:

metadata;
multi-dimensional arrays;

time series;
administration data.
In its generic form, it allows also the sending institution to use the data set structure it prefers.

GESMES was accepted as UN/EDIFACT Status 1 messages in 1995 and was first published in the UN/D95A directory. The statistical office of the European Union, EUROSTAT, who has lead the development of statistical UN/EDIFACT messages is implementing GESMES into the data flows between it and the Member States of the EEA (European Economic Area) and promoting the usage of the messages by other international organisations and by other sectors.

A description of the EDIFACT syntax is provided on the Appendix. For readers who are not familiar with the EDIFACT terminology and syntax, **it is strongly recommended that they familiarise themselves with the Appendix providing background on this subject.**

7 FUNCTIONALITY OF SDMX-EDI AND POSSIBLE DATA EXCHANGE ACTIVITIES

SDMX-EDI provides all different types of functionality required for 'two way' statistical data exchange; for example: between a national central bank (NCB) or a national statistical institute (NSI) and a central institution (e.g. BIS, IMF, Eurostat, ECB, OECD) or between any two institutions as long as they have made known to each other the semantics of the exchanged information (structural definitions). Of course, also a NCB or a NSI could act as a "centre" vis-à-vis other institutions by devising appropriate *structural definitions* (i.e. statistical concepts, code lists, key families) for the data exchanges it is going to administrate.

7.1 SDMX-EDI Functionality

The different levels of functionality allow partner institutions to implement SDMX-EDI in a phased approach:

- ***Data update:*** Exchange only observations (including the mandatory attribute "observation status").
- ***Data and attribute update:*** In addition to observations and observation status, the exchange could also comprise coded and uncoded attributes linked to the data on a data set, sibling group, time series and observation level.
- ***Deletions:*** send delete messages of observations and observation status;

in addition to observations and observation status, send delete messages for coded and uncoded attributes linked to the data on a data set, sibling group, time series and observation level.

and, in the longer-*term*:

- ***Exchange of structural definitions:*** Exchange of code lists, lists of statistical concepts and key family definitions.

7.2 Reporting and Dissemination

The *data update message* has been defined in order to identify the *minimum requirement for data reporting to a central institution (absolutely essential)*. However, it is expected that, over time, partner institutions develop the capability to also provide coded and uncoded attributes relating to the series they report, i.e. to provide the *data and attribute update message which is considered also as essential*. If they erroneously report observations (and attributes), partner institutions will also have to send a *data (and attribute) delete message* (the delete messages are in fact very similar to the ones used for updating).

Dissemination from a central institution may take the form of *complete databases*[1] (e.g. on a CD-ROM), or selected output, or the regular provision of *net updates and revisions*:

Complete Data bases:

On a complete database, the receiving institution will find in the SDMX-EDI format:

- ***data and attribute update message(s)*** with the observation values and the values of the coded and uncoded attributes;
- ***a structural definition message***, which gives information about the statistical concepts, code lists and key families used in the data message(s);

Delete messages cannot be included, as in this case a complete replacement of the previous copy of the database is provided.

Updates and Revisions from a Centre:

Updates and revisions disseminated by a centre enable receiving institutions to get only the information on observations and attributes that has changed in the database since a specific date. This also covers the dissemination of new series (or sibling groups) that have been added to the database, and deletions at the sibling group, series and observation level. In order to take full advantage of this option, receiving institutions have, of course, to be able to interpret *data and attribute update* and *delete messages*.

The use of the different messages[2] envisaged for SDMX-EDI is summarised in the following table, which also indicates the 'activity' (i.e. *write* or *read*) that partner institutions or the centre perform on the message for the different types of usage.

[1] A complete database: the whole ETS (including data, attributes, structural definitions) or a consistent subset of it.

[2] Actually, these are sub-messages of SDMX-EDI; each one of them reflects a selection of appropriate sections of SDMX-EDI in order to serve the desired activities and to support the corresponding functionalities. In this table the exchange of lists of data sets is not considered.

361 **Table 1.** Activities and message types

Activity	Message type	Partner	Centre
'Simple' reporting to a centre	• Data update message • Data delete message	W W	R R
'Advanced' reporting to a centre, including information on coded and uncoded attributes	• Data/attribute update message • Data/attribute delete message	W W	R R
Full data base dissemination from a centre	Structural definitions message Data/attribute update message	R R	W W
Update/revision output from a centre, ad hoc or scheduled	• Data/attribute update message • Data/attribute delete message	R R	W W

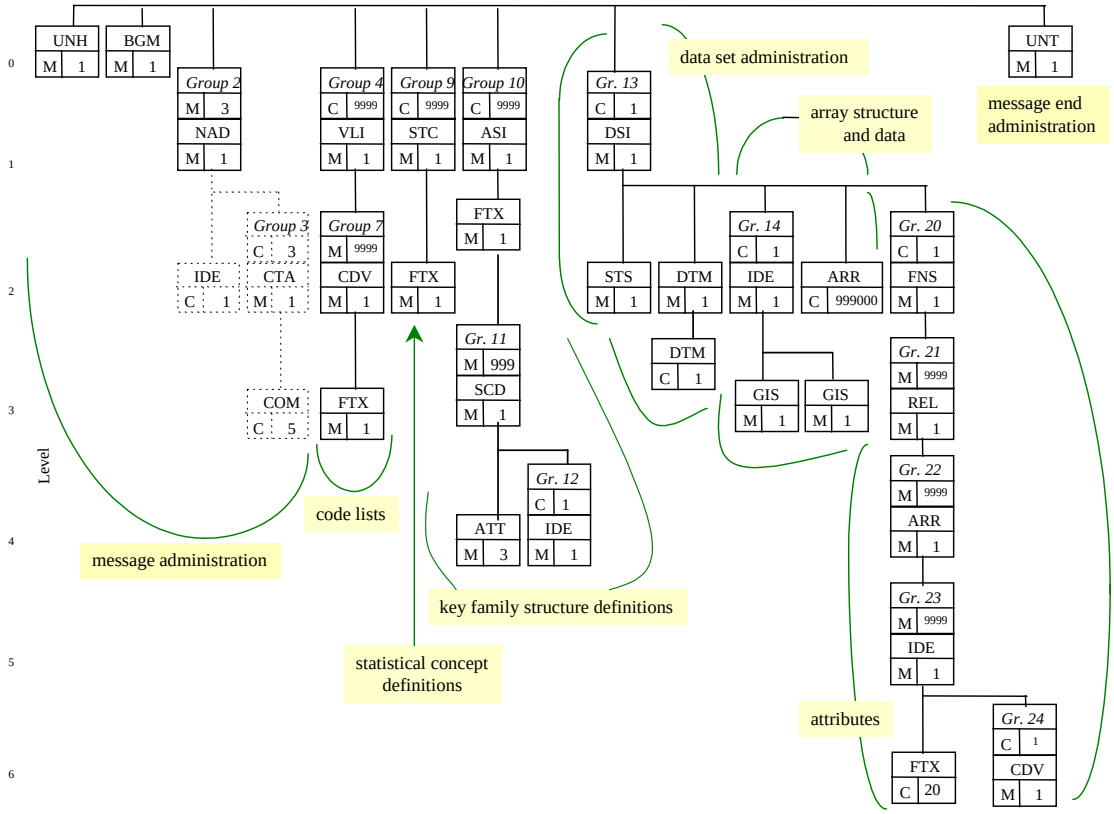
362

8 THE STRUCTURE OF SDMX-EDI

A SDMX-EDI interchange (file) may contain several messages; each one of those messages comprises a sequence of *segments*, reflecting each one of them a small logical section of the message. Each segment starts with a unique 3 character identifier (segment tag). Segment identifiers which begin with UN are called *service segments* and are defined as a part of the EDIFACT syntax (ISO 9735). Other segments are called *user data segments* and are defined in the Segments Directory of the UN Trade Data Interchange Directory (UNTDID)[1].

8.1 SDMX-EDI: The Branching Diagram

The most common way to present the contents of a GESMES message is to use its *branching diagram*. A branching diagram shows the segments which are included in the message (via their segment tags), their status (mandatory or conditional) and the maximum number of occurrences which are allowed; as the branching diagram describes a message and not the interchange file, the interchange header and trailer are not shown. The interpretation of a typical branching diagram is explained in the Appendix. Here is the branching diagram of a SDMX-EDI message including all its segments grouped in sections; each section reflects a functionality:



[1] http://www.unece.org/trade/untdid/d99b/trmd/gesmes_c.htm

The "conditional" status of the first segment of each section (e.g. VLI in Group 4, DSI in Group 13, FNS in Group 20) would turn into "mandatory" if the corresponding facility had to be used.

8.2 Functionality, Segment Sections and Step-by-Step Implementation

The segments of the previous diagram can be grouped in categories each one of them performing a logical task. In the list below the interchange header and trailer have been also included (numbered lines 1 and 10. The logical sections of the main body of a message are presented in the numbered lines 2-9:

Interchange administration (UNA and UNB segments)

Message administration (UNH, BGM, NAD+Z02, NAD+MR, NAD+MS segments)

Code lists (VLI, CDV, FTX segments)

Statistical concept definition (STC, FTX segments)

Key family definition (ASI, SCD, ATT, IDE segments)

Data set administration (DSI, STS, DTM segments)

Array structure (IDE, GIS, GIS segments)

Data (ARR segment)

Attributes (FNS, REL, ARR, IDE, FTX, CDV segments)

End of message administration (UNT segment)

<at this point, just after the "end of message administration", more *messages [repetitions of the sequence of items 2-9]* can be present in the same interchange>

End of interchange administration (UNZ segment)

The table below indicates which of the sections above are necessary (and their order) in a message, according to the desired function of the message.[1]

[1] In this table the exchange of lists of data sets is not considered.

Table 2. Data and Attribute Update Messages

Type of message/data exchange:	ABSOLUTELY ESSENTIAL Data update message	Attribute update message	Data and attribute update message
Section	(ordered) sections which must be present		
Interchange administration segments	Yes	Yes	Yes
Message administration segments	Yes	Yes	Yes
Code lists			
Statistical concepts			
Key families			
Data set administration segments	Yes	Yes	Yes
Array structure segments	Yes	Yes	Yes
Data segment(s)	Yes		Yes
Attributes		Yes	Yes
End of message administration	Yes	Yes	Yes
End of interchange administration	Yes	Yes	Yes

Example 1:

A **Data Update message** has to have the sections (1) Interchange administration, (2) Message administration segments, (3) Data set administration segments, (4) Array structure, (5) data segment(s), (6) End of message administration and (7) End of interchange administration.

Example 2:

A **Data and Attribute Update message** (as it contains attributes), has to include also the *attributes section* (just before the *end of message administration*).

When the purpose of a message is to indicate to the receiver institution to *delete* objects, then according to what should be deleted, the following sections should be used:

Table 3. Data and Attribute Delete Messages

Type of message/data exchange:	Data delete	Attribute delete	Data and Attribute delete message
Section	(ordered) sections which must be present		
Interchange administration segments	Yes	Yes	Yes
Message administration segments	Yes	Yes	Yes
Code lists			
Statistical concepts			
Key families			

Data set administration segments	Yes	Yes	Yes
Array structure	Yes	Yes	Yes
Data segment(s)	Yes		Yes
Attributes		Yes	Yes
End of message administration	Yes	Yes	Yes
End of interchange administration	Yes	Yes	Yes

The list of segments indicated in this table is in principle the same as the previous one. But when deleting attributes, not all segments included in the *attributes* section are used (for further information see the *Delete Messages* chapter).

The following table indicates which sections have to be used in order to prepare structural messages (only for dissemination from the centre to the other partners):

Table 4. Structural messages

Type of message/data exchange:	Code lists	statistical concept definition	key family definition
Section	(ordered) groups of segments which must be present		
Interchange administration segments	Yes	Yes	Yes
Message administration segments	Yes	Yes	Yes
Code lists	Yes		
Statistical concepts		Yes	
Key families			Yes
Data set administration segments			
Array structure			
Data segment(s)			
Attributes			
End of message administration	Yes	Yes	Yes
End of interchange administration	Yes	Yes	Yes

Remarks:

In fact the distinction into different types of messages is artificial, just to facilitate the presentation and the correspondence to logical functionalities: the type of a message (e.g. Data update, Attribute update, Data and attribute update, Data delete etc.) is exclusively determined by the presence or not of the specific sections (and the setting of some parameters).

In the next sections of the chapter each type of message (e.g. data and attribute update) is further illustrated by presenting the function of the segments used within each section.

The branching diagrams presented in the rest of this chapter describe specific message types and therefore the branching diagrams reflect sub-sets of the full message. Moreover, as the emphasis is on how to support desired activities and to serve specific functionalities, the

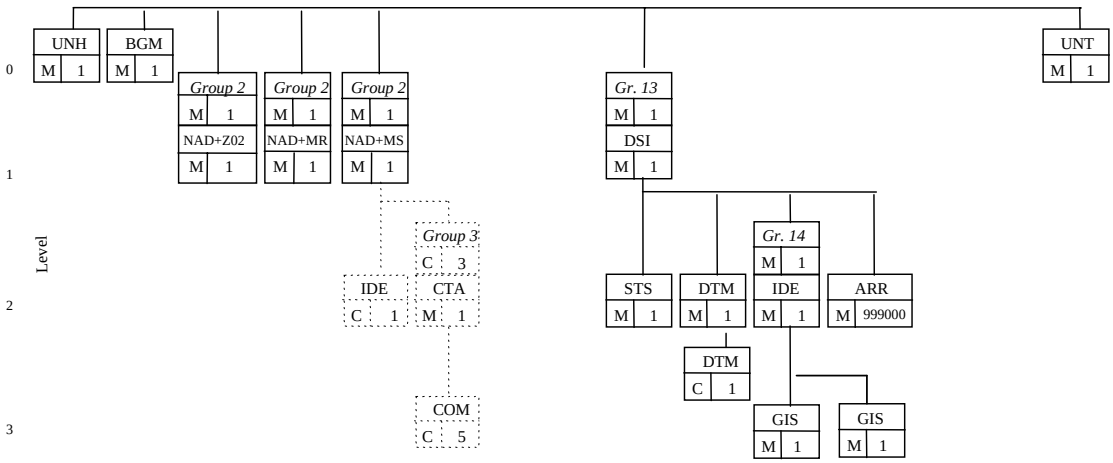
indicated segment status (mandatory/conditional) might be different here for some segments (vis-à-vis the full message) in order to indicate what is really needed per activity and desired functionality.

The reader going through the tables of the following sections of this chapter might find convenient also to refer (for each segment) to the analytical presentation of the *message sections* and *segments* of the next chapter; in this way, the details per segment presented in that chapter ("Segment by segment reference guide") will enlighten the very brief (but global) overview given in this chapter. This would be the most efficient way for studying the whole guide.

8.2.1 Data Update Message: Segment Structure and Purpose

When the goal is the exchange of *observations* (and their associated *array cell attributes*) only, then the segments belonging to the *structural section* (key family definition) and to the *attributes section* are not used.

Therefore, the complete branching diagram collapses to the following one:



This diagram describes the absolutely essential parts of the message that every institution should be able to manage (write and read) at the first implementation phase. *All segments (apart from the ones in dotted lines) are presented here as mandatory because, indeed, they are all necessary in order to serve this functionality.*

The three NAD segments (NAD+Z02, NAD+MR, NAD+MS) are presented here separately for reasons of clarity.

A brief explanation of the segments used (and needed) in this simple message is presented in the following table:

456 **Table 5. Data update message**

Sections and segments of the message	Purpose
interchange administration	
UNA*service-characters*	defines the service characters
UNB+*syntax-identifier*:*syntax-version*+*sender-identification*+*receiver-identification*+*date-of-preparation*:*time-of-preparation*+*interchange-reference*++*application-reference*(++++*test indicator*)'	interchange header
message administration	
UNH+*message-reference-number*+*message-type*:*message-type-version*:*message-type-release*:*controlling-agency*'	message identification
BGM+*message-function*'	message function (=74 for data/attr.)
NAD+*Z02*+*organisation-id*'	code list maintenance agency
NAD+*MR*+*organisation-id*' NAD+*MS*+*organisation-id*'	receiver identification sender identification
data set administration	
DSI+*data-set-identifier*'	data set identifier
STS+3+*status-code*'	update/replace (*status-code*=7)
DTM+*date-time-type*:*date-time*:*date-time-format*' DTM+*date-time-type*:*date-time*:*date-time-format*'	data set preparation date and time (and -conditional- reporting period)
array structure and data	
IDE+*5*+*identifier*'	key family identifier
GIS+*AR3*'	method to send data in the ARR segment
GIS+1:::*symbol-used-for-missing-values*'	symbol used for missing values
ARR++*key*:*period*:*date-format*:*observation*:*obs-status*:*obs-confid.*' ARR++*key*:*period*:*date-format*:*observation*: *obs-status*:*obs-confid.*' ... etc.	contains the numeric data; two methods can be used: - a single observation per ARR segment...
or: ARR++*key*:*start_date*-*end_date*:*time_range_format*:*observation*:*obs-status*:*obs-confid.*+ *observation*: *obs-status*:*obs-confid.*+...+*observation*: *obs-status*:*obs-confid.*'	- or a time range with the corresponding observations;
ARR++*key*:*start_date*-*end_date*:*time_range_format*:*observation*:*obs-status*:*obs-confid.*+ *observation*:*obs-status*:*obs-confid.*+... +*observation*: *obs-status*:*obs-confid.*' ...etc.	the ARR segment can appear up to 999000 times.
end of message administration (message trailer)	
UNT+*number-of-segments*+*message-reference-number*'	end of message control data
end of interchange administration (interchange trailer)	
UNZ+*number-of-messages*+*interchange-reference*'	end of interchange control data

Example:

Let's assume that the National Bank of Belgium is sending the time series M:BE:PROD:GN:NS and Q:BE:PROD:GN:NS (belonging to a key family called ECB_TESTPRICES) to the ECB.

For the monthly time series M:BE:PROD:GN:NS the following observations (together with their "status") are reported:

Sep95	Oct95	Nov95	Dec95	Jan96	Feb96	Mar96
99.10 A	98.10 A	98.40 A	99.50 A	100.00 A	99.20 A	99.80 E C

The flags A (="normal") and E (="estimate") are values for the Observation Status attribute which is attached next to each observation. The flag C (=confidential) is a value for the Observation Confidentiality attribute which can be attached next to the observation status to provide information about the confidentiality status of an observation. For more details about the usage of these code lists, please refer to the Appendix presenting the corresponding code lists.

For the quarterly series Q:BE:PROD:GN:NS the following data have to be sent:

95q4	96q1
98.67 A	99.67 A

Using SDMX-EDI, these data have to be sent by the central bank of Belgium in the following file:

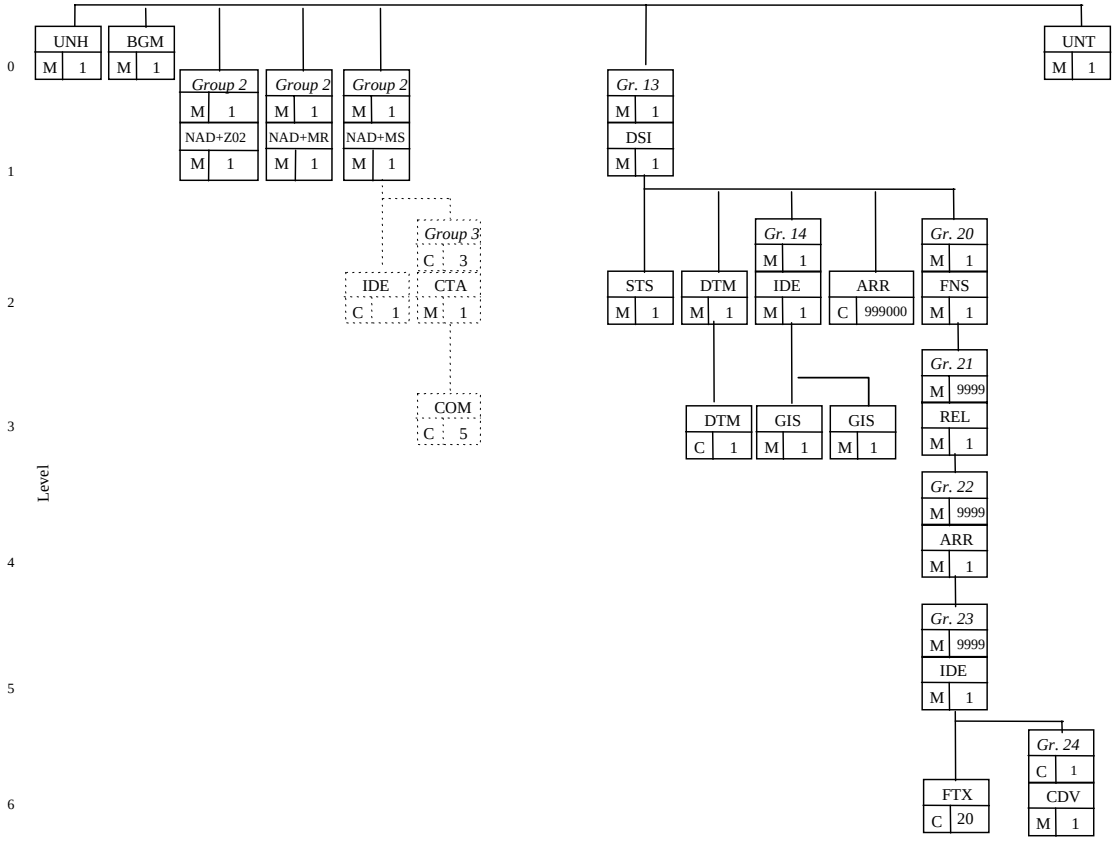
BE2=National Bank of Belgium, 4F0 = ECB *(example codes for organisations involved)*

```
UNA:+.? '
UNB+UNOC:3+BE2+4F0+970525:1539+IREF000001++SDMX-EDI'
UNH+MREF000001+GESMES:2:1:E6'
BGM+74'
NAD+Z02+ECB'
NAD+MR+4F0'
NAD+MS+BE2'
DSI+ECB_TESTPRICES'
STS+3+7'
DTM+242:199705251539:203'
IDE+5+ECB_TESTPRICES'
GIS+AR3'
GIS+1:::-'
ARR++M:BE:PROD:GN:NS:199509199603:710:99.10:A+98.10:A+98.40:A+99.50:A+100.00:A+
99.20:A+99.80:E:C'
ARR++Q:BE:PROD:GN:NS:1995419961:708:98.67:A+99.67:A'
UNT+14+MREF000001'
UNZ+1+IREF000001'
```

494 In this example the non-fixed elements are underlined and it is obvious that, apart from the
495 actual data, they provide mainly administrative information (e.g. BE2=central bank of Belgium,
496 4F0=ECB, ECB_TESTPRICES=key family identifier). For more explanations about the use of
497 these segments, refer to the reference guide chapters of this document.

8.2.2 Data and Attribute Update Message: Segment Structure and Purpose

An institution reporting data can send, apart from observations, also additional information which helps the users of the exchanged time series to better understand their nature or their special characteristics (title, methodology, etc.). These pieces of information are called attributes and they can be either coded or uncoded. They are exchanged using the FNS segment and the segments which follow it. The branching diagram presented in the previous section is now enlarged, including also the set of segments which carry the attributes:



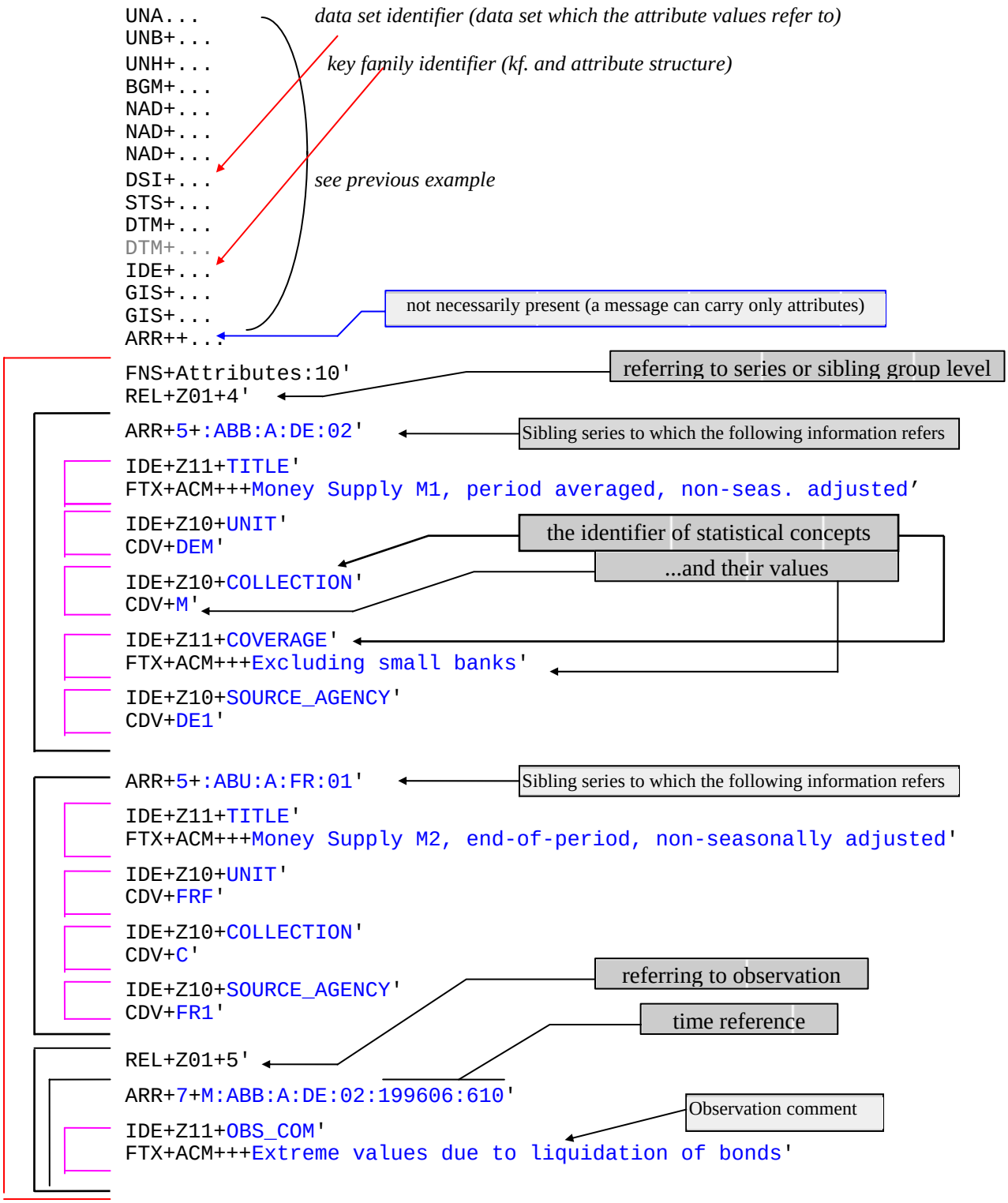
- Group 13 is presented here as mandatory: without this it is impossible to exchange observations and/or attributes;
- Group 14 is shown also as mandatory: for reporting *observations (with their associated observations flags) it is obvious that it is needed (it provides information about the structure in the ARR segment following this Group)*; also, even if the message is used to report only attributes, Group 14 should be used (only the ARR segment which follows immediately after Gr.14 would not be needed in this case).
- Group 20 is shown as mandatory, as it is the Group carrying the attributes.
- If there is a need to exchange *both observations and attributes then, apart from the essential Groups 13 and 14, the ARR segment which follows Group 14 and Group 20 must be present.*

The following table highlights the use of segments when both data and attributes are reported.

519 **Table 6. Data and attribute update message**

Sections and segments of the message	Purpose
UNA UNB+... UNH+... BGM+... NAD+... NAD+... NAD+... DSI+... STS+... DTM+... DTM+... IDE+... GIS+... GIS+... ARR++...	*update/replace*	*as described in the previous table (see DATA UPDATE MESSAGE)*
Attributes	
FNS+*general-attribute-identifier:identity-number-type*'	[segment introducing attributes]
REL+*Z01+array-scope*'	attribute scope
ARR+*last-dimension-position+key-dimension:key-dimension:key-dimension-key:dimension etc.*'	dimension/key pointer
IDE+*Z10+coded-attribute-identifier*' CDV+*coded-attribute-value*'	attribute identifier (Z10=coded attr.) attribute code value
IDE+*Z10+coded-attribute-identifier*' CDV+*coded-attribute-value*' etc.	attribute identifier attribute code value
IDE+*Z11+uncoded-attribute-identifier*' FTX+ACM+++*text*'	attribute identifier (Z11=uncoded) attribute text
IDE+*Z11+uncoded-attribute-identifier*' FTX+ACM+++*text*' etc.	attribute identifier attribute text
ARR+*last-dimension-position+key-dimension:key-dimension:key-dimension-key:dimension etc.*'	a new dimension/key pointer can follow
Etc.	new sets of segments 'IDE/CDV' and IDE/FTX can follow
UNT+... UNZ+...	*as described in the previous table*

Example:


UNA...
UNB+...
UNH+...
BGM+...
NAD+...
NAD+...
NAD+...
DSI+...
STS+...
DTM+...
DTM+...
IDE+...
GIS+...
GIS+...
ARR++...
FNS+Attributes:10'
REL+Z01+4'
ARR+5+:ABB:A:DE:02'
IDE+Z11+TITLE'
FTX+ACM+++Money Supply M1, period averaged, non-seas. adjusted'
IDE+Z10+UNIT'
CDV+DEM'
IDE+Z10+COLLECTION'
CDV+M'
IDE+Z11+COVERAGE'
FTX+ACM+++Excluding small banks'
IDE+Z10+SOURCE_AGENCY'
CDV+DE1'
ARR+5+:ABU:A:FR:01'
IDE+Z11+TITLE'
FTX+ACM+++Money Supply M2, end-of-period, non-seasonally adjusted'
IDE+Z10+UNIT'
CDV+FRF'
IDE+Z10+COLLECTION'
CDV+C'
IDE+Z10+SOURCE_AGENCY'
CDV+FR1'
REL+Z01+5'
ARR+7+M:ABB:A:DE:02:199606:610'
IDE+Z11+OBS_COM'
FTX+ACM+++Extreme values due to liquidation of bonds'
UNT+...
UNZ+...
data set identifier (data set which the attribute values refer to)
key family identifier (kf. and attribute structure)
see previous example
not necessarily present (a message can carry only attributes)
referring to series or sibling group level
Sibling series to which the following information refers
the identifier of statistical concepts
...and their values
Sibling series to which the following information refers
referring to observation
time reference
Observation comment

For further explanations about the use of the attribute related segments, the reader can refer to the segment-by-segment reference guide chapters.

8.2.3 Data and Attribute Delete Message

The table and the structure presented in the previous section could be used also, apart from sending data or attributes, to delete objects (which had been made already available to the receiving institution in the past). This is done by using exactly the same structure (as in the Data and Attribute Update message) and...

- **flagging appropriately the message** using a different parameter in the STS segment;

- **in order to delete data**:

  - the ARR segment which follows Group 14 should not contain observations: its references are used simply to point to (existing, previously reported, via an interchange in the past) observations which should be deleted.

Example 1:

The segments

```
...
STS+3+6'
...
ARR++M:BE:PROD:GN:NS:199509199603:710'
...
```

inform the receiving institution that it should delete the monthly observations of the series M:BE:PROD:GN:NS from Sep95 to Mar96.

Example 2:

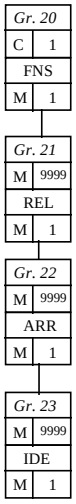
The segments

```
...
STS+3+6'
...
ARR++:BE:PROD:GN:NS'
...
```

inform the receiving institution that it should delete the sibling group (assuming that the second dimension is the frequency) :BE:PROD:GN:NS and all associated attributes at this and at lower levels.

- or, **in order to delete attributes**:

  - FTX and CDV after Group 23 are not used (see diagram on the right) and the ARR segment (Group 22) is used exclusively to point the attributes which have to be deleted.



  Example:
  The segments

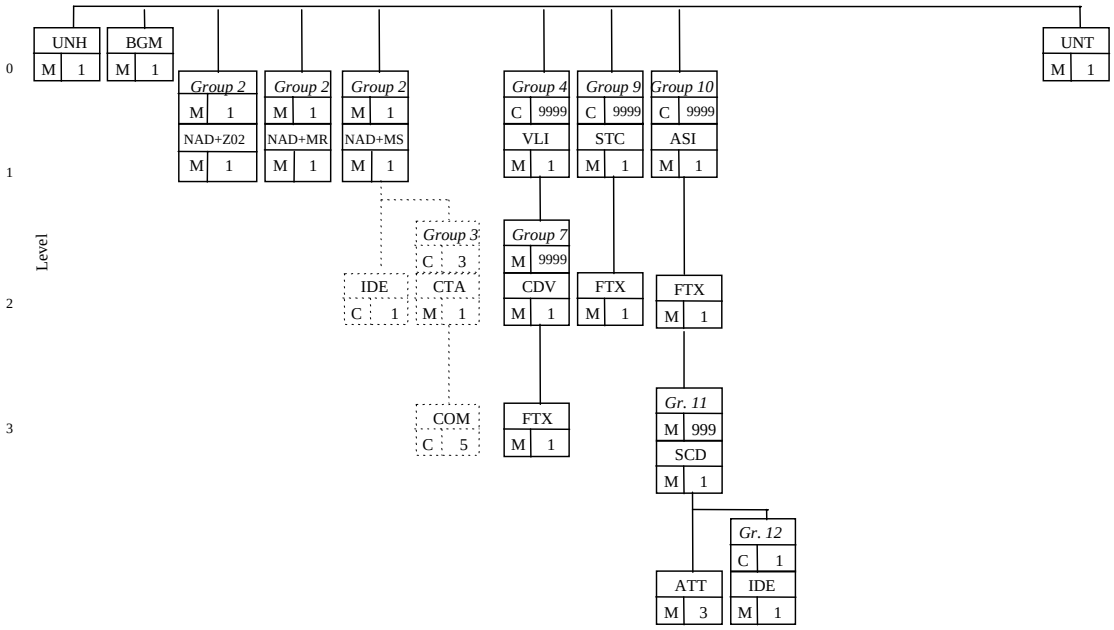
```
...
STS+3+6'
...
FNS+Attributes:10'
REL+Z01+4'
ARR+5+:ABB:A:DE:02'
IDE+Z11+COVERAGE'
```

  inform the receiving institution that it should delete the attribute referring to the *Coverage* of the sibling group :ABB:A:DE:02.

*A **more extensive and detailed discussion on issues related to deletions** is presented as a separate chapter (see page 136).*

8.2.4 Structural Definitions Exchange Message: Segment Structure and Purpose

The statistical data exchange system can become completely paperless, when the centre disseminates all structural data (concepts, key family definitions and code lists) using SDMX-EDI. In such a message, and in order to keep it clear, only structural (and administration) segments are included, as in the message below:



A structural message can contain one of the three sections (i.e. code lists, statistical concept definitions, key families), two of them or all three. Normally, all these three sections are used (in one or more messages) in order to provide statistical centres with the possibility to disseminate to their partners all structural definitions needed in an electronic form.

8.2.5 Code Lists

The segments relating to code lists (VLI, CDV, FTX) provide the facility to disseminate code lists for coded statistical concepts; they are used either as dimensions in a specific key family or as coded attributes.

8.2.6 Statistical Concept definition

The segments relating to the definition of statistical concepts (STC, FTX) are used to provide the link between a statistical concept identifier and its actual name.

8.2.7 Key Family Definition

The segments relating to the structure definition of a specific key family serve several purposes. They are used:

to assign a key family identifier and a description to a key family;

to define the statistical concepts used as dimensions in the key structure (information which is provided: position of each concept in the key, code value length and relevant code list);

to define the exact structure of the ARR segment (which contains the numeric values of the message) for this particular key family;

637 to define the coded and uncoded attributes used in the key family (concept identifier, field length,
638 usage status and attachment level; also, relevant code list in the case of coded attributes).

639

640 **Table 7. Structural data exchange message**

Sections and segments of the message	Purpose
UNA UNB+... UNH+... BGM+... NAD+... NAD+... NAD+...	BGM +73' *should be written now (no data/attr., only definitions)* *as before*
code lists	
VLI+*code-list-id*+++*code-list-name* '	Identifier of the code list
CDV+*code-value* ' FTX+ACM+++*code-value-description* '	code value text description
CDV+*code-value* ' FTX+ACM+++*code-value-description* ' *etc.*	code value text description
VLI+*code-list-id*+++*code-list-name* '	Identifier of next code list
CDV+*code-value* ' FTX+ACM+++*code-value-description* '	code value text description
CDV+*code-value* ' FTX+ACM+++*code-value-description* ' *etc.*	code value text description
(etc. ...for other code lists)	
statistical concept definition	
STC+*concept-identifier* ' FTX+ACM+++*concept-name* '	statistical concept identifier concept name
STC+*concept-identifier* ' FTX+ACM+++*concept-name* '	statistical concept identifier concept name
etc.	
key family definition	
ASI+*key-family-identifier* ' FTX+ACM+++*key-family-name* '	key family identifier key family name
(i) key structure (dimensions)	
SCD+*n*+*concept-identifier*++++:*dimension-position-in-the-key* '	*n*=13 if dimension is FREQ, *n* =4 otherwise
ATT+3+5+:::AN*field-length* '	
IDE+1+*code-list-id* '	corresponding code list
SCD+4+*concept-identifier*++++:*dimension-position-in-the-key* '	

ATT+3+5+:::AN*field-length*' IDE+1+*code-list-id*'	corresponding code list
etc. (for all key family dimensions)	
(ii) time, time format, observation value, coded attributes at obs. level	*(other ARR elements:)*
SCD+1+TIME_PERIOD++++:position-in-the-ARR-structure'	*time period* follows series key in the ARR structure
ATT+3+5+:::AN35'	then, *time format* follows
SCD+1+TIME_FORMAT++++:position-in-the-ARR-structure'	then, the observation follows
ATT+3+5+:::AN3'	
SCD+3+OBS_VALUE++++:position-in-the-ARR-structure'	the observation status follows and: - it is one character long - it is mandatory (=2) - attached at the obs. level (=5)
ATT+3+5+:::AN15'	
SCD+3+OBS_STATUS++++:position-in-the-ARR-structure'	
ATT+3+5+:::AN1' ATT+3+35+2:USS' ATT+3+32+5:ALV'	
(iii) key family attributes *[for coded attributes]*	**statistical concept identifier**
SCD+Z09+*concept-identifier*'ATT+3+5+:::AN*field-length*' (or: "AN..*max-field-length*" for variable length)	(max) length of the attribute value
ATT+3+35+*usage-status-code*:USS'	mandatory or conditional status
ATT+3+32+*attachment-level-code*:ALV'	attachment level
IDE+1+*code-list-id*'	corresponding code list
SCD+Z09+*concept-identifier*'ATT+3+5+:::AN*field-length*' (or: "AN..*max-field-length*" for variable length)	**statistical concept identifier** (max) length of the attribute value
ATT+3+35+*usage-status-code*:USS'	mandatory or conditional status
ATT+3+32+*attachment-level-code*:ALV'	attachment level
IDE+1+*code-list-id*'	corresponding code list
etc. (for all coded attributes of the key family)	
[for uncoded attributes]	
SCD+Z09+*concept-identifier*'	statistical concept identifier
ATT+3+5+:::AN..*max-field-length*'	max. length of the attribute value
ATT+3+35+*usage-status-code*:USS'	mandatory or conditional status
ATT+3+32+*attachment-level-code*:ALV'	attachment level
SCD+Z09+*concept-identifier*'	statistical concept identifier
ATT+3+5+:::AN..*max-field-length*'	max. length of the attribute value
ATT+3+35+*usage-status-code*:USS'	mandatory or conditional status
ATT+3+32+*attachment-level-code*:ALV'	attachment level
etc. (for all uncoded attributes of the key family)	
ASI+...etc. (key family definitions may follow for additional key families)	*(as discussed in the previous*

tables)

UNT+...
UNZ+...

641

Example:

```
UNA:+.? '
UNB+UNOC:3+4F0+BE2+970525:1539+IREF000001++SDMX-EDI'
UNH+MREF000001+GESMES:2:1:E6'
BGM+73'
NAD+Z02+ECB'
NAD+MR+BE2'
NAD+MS+4F0'

VLI+CL_FREQ+++Frequency'
CDV+A'
FTX+ACM+++Annual'
CDV+Q'
FTX+ACM+++Quarterly'
CDV+M'
FTX+ACM+++Monthly'
CDV+D'
FTX+ACM+++Daily'

VLI+CL _AREA+++Country or area'
CDV+BE'
FTX+ACM+++Belgium'
CDV+DK'
FTX+ACM+++Denmark'
...
CDV+US'
FTX+ACM+++United States'

VLI+CL_BOP_ITEM+++IMF Bal. of payments component'
CDV+100'
FTX+ACM+++Current account; goods'
CDV+200'
FTX+ACM+++Current account; services'
CDV+300'
FTX+ACM+++Factor income'
...
CDV+995'
FTX+ACM+++Financial account'
CDV+998'
FTX+ACM+++Errors and omissions'

VLI+CL_BOP_DATA_TYPE+++Type of data'
CDV+1'
FTX+ACM+++Stock at the beginning of the period'
CDV+2'
FTX+ACM+++Credit flow'
CDV+3'
FTX+ACM+++Debit flow'
CDV+4'
FTX+ACM+++Net flow'
CDV+5'
FTX+ACM+++Price valuation adjustment'
CDV+6'
FTX+ACM+++Exchange rate adjustment'
CDV+7'
FTX+ACM+++Reclassification, other adjustments and other revaluations'
CDV+8'
FTX+ACM+++Stock at the end of the period'
```

Disseminating code lists

(1= Stock at the beginning of the period)

```
701         VLI+CL_UNIT+++Unit'
702         CDV+BEF'
703         FTX+ACM+++Belgian franc'
704         CDV+DEM'
705         FTX+ACM+++Deutsche Mark'
706         ...
707         CDV+USD'
708         FTX+ACM+++US dollar'
709
710         VLI+CL_UNIT_MULT+++Unit multiplier'
711         CDV+6'
712         FTX+ACM+++Millions'
713         CDV+9'
714         FTX+ACM+++Billions'     etc.
```

Disseminating statistical concept definitions

```
715     STC+FREQ'
716     FTX+ACM+++Frequency'
717     STC+REF_AREA'
718     FTX+ACM+++Reference country or area'
719     STC+BOP_ITEM'
720     FTX+ACM+++IMF breakdown; balance of payments items (standard component)'
721     STC+DATA_TYPE'
722     FTX+ACM+++Type of data'
723     STC+TITLE'
724     FTX+ACM+++Title'
725     STC+UNIT'
726     FTX+ACM+++Unit'
727     STC+UNIT_MULT'
728     FTX+ACM+++Unit multiplier'
729     STC+TIME_PERIOD'
730     FTX+ACM+++Time period or time range'
731     STC+TIME_FORMAT'
732     FTX+ACM+++Time format'
733     STC+OBS_VALUE'
734     FTX+ACM+++Observation value'
735     STC+OBS_STATUS'
736     FTX+ACM+++Observation status'
737     STC+OBS_CONF'
738     FTX+ACM+++Observation confidentiality'
739
```

Disseminating a key family's definition

```
740
741     ASI+BAL_OF_PAYM_TEST'
742     FTX+ACM+++Bal. of payments k.f. for testing'
743
744     SCD+13+FREQ++++:1'          First dimension  is "frequency"
745     ATT+3+5+:::AN1'                and it is one character long;
746     IDE+1+CL_FREQ'                 corresponding code list: CL _FREQ
747     SCD+4+REF_AREA++++:2'       Second dimension of the series is "reference area"
748     ATT+3+5+:::AN2'                and it is two characters long;
749     IDE+1+CL_AREA'                 corresponding code list: CL_AREA
750     SCD+4+BOP_ITEM++++:3'    Third dimension is "IMF BoP standard component"
751     ATT+3+5+:::AN3'                and it is three characters long;
752     IDE+1+CL_BOP_ITEM'             corresponding code list: CL_BOP_ITEM
753     SCD+4+DATA_TYPE++++:4'      Fourth dimension is "Type of data/position of transaction"
754     ATT+3+5+:::AN1'                and it is one character long;
755     IDE+1+CL_BOP_DATA_TYPE'        corresponding code list: CL_BOP_DATA_TYPE
```

Other component elements in the ARR structure structure

```
SCD+1+TIME_PERIOD++++:5'          Fifth component element is time
ATT+3+5+:::AN..35'                and it is a descr. up to 35 char. long;
SCD+1+ TIME_FORMAT++++:6'         Sixth component element is the time format
ATT+3+5+:::AN3'                   and it is a number 3 char. long;

SCD+3+OBS_VALUE++++:7'            Seventh component element is the observation
ATT+3+5+:::N15'                   and it is a 15 char. long numeric field;
                                                    -- observation attributes --
SCD+3+OBS_STATUS++++:8'           Eighth component element is the observation status,
ATT+3+5+:::AN1'                   it is one character long,
ATT+3+35+2:USS'                   it is mandatory  (usage status=2),
ATT+3+32+5:ALV'                   it is attached at the observation level (attachment level=5) and
IDE+1+CL_OBS_STATUS'              it takes its values from the CL_OBS_STATUS code list.

SCD+3+OBS_CONF++++:9'             Ninth component element is the observation confidentiality flag
ATT+3+5+:::AN1'                   it is one character long,
ATT+3+35+1:USS'                   it is conditional  (usage status=1),
ATT+3+32+5:ALV'                   it is attached at the observation level (attachment level=5) and
IDE+1+CL_OBS_CONF'                it takes its values from the CL_OBS_CONF code list.
```

Disseminating same key family's definition of attributes

```
SCD+Z09+TITLE'                    "Title" is an attribute (=Z09) of the key family
ATT+3+5+:::AN..70'                  it can be up to 70 characters long
ATT+3+35+2:USS'                   it is mandatory  (usage status=2) and
ATT+3+32+9:ALV'                   it is attached at the sibling group level (attachment level=9)
SCD+Z09+UNIT'                     "Unit" is an attribute of this key family
ATT+3+5+:::AN..12'                  and it can be up to 12 characters long,
ATT+3+35+2:USS'                   it is mandatory  (usage status=2),
ATT+3+32+9:ALV'                   it is attached at the sibling group level (attachment level=9) and
IDE+1+CL_UNIT'                    it takes its values from the CL_UNIT code list

SCD+Z09+UNIT_MULT'                "Unit multiplier" is an attribute of the key family,
ATT+3+5+:::AN..2'                 it can be 1 or 2 positions long,
ATT+3+35+2:USS'                   it is mandatory  (usage status=2),
ATT+3+32+9:ALV'                   it is attached at the sibling group level (attachment level=9) and
IDE+1+CL_UNIT_MULT'               it takes its values from the CL_UNIT_MULT code list

UNT+135+MREF000001'

UNZ+1+IREF000001'
```

The example given above shows how a centre (e.g. the ECB, coded as 4F0) disseminates structural data to a central bank (e.g. to the National Bank of Belgium, coded as BE2). The dissemination of the concept definition *(segments starting with STC)* and of the code lists *(groups starting with VLI)* provide the receiver with some basic metadata; then, the dimensions are given *(first group of segments starting with ASI)* and the receiver can form the key structure of the example key family *(BAL_OF_PAYM_TEST)*:

	FREQ	REF_AREA	BOP_ITEM	DATA_TYPE
dimension value length in characters:	*1*	*2*	*3*	*1*

This is the structure of the key of the time series of the key family. E.g. a specific time series could be the monthly series (FREQ=*M*) referring to the *goods* (BOP_ITEM=*100*) *balance* (type:net=*4*) of Belgium (REF_AREA=*BE*); its key would be:

M:BE:100:4

Additional information is provided by the attributes TITLE (uncoded), UNIT (coded) and UNIT_MULT (coded) of the key family which for this series might take the values:

TITLE=*Merchandise trade balance*, UNIT=*BEF*, UNIT_MULT=*9*

9 SEGMENT BY SEGMENT REFERENCE GUIDE

9.1 Organisation of the Guide

The segments are presented within separate sections like the ones presented in

Table 2. As it was earlier discussed, each section reflects a type of functionality (e.g. message administration). All sections are described in the following way:

Scope

This defines the scope of the facility.

GESMES segments used

This shows the branching diagram of GESMES with the relevant segments highlighted.

Features and guidelines for usage

This explains the way the segments are used to implement the facility.

For each segment

- data element sequence

This shows the sequence of data elements in the segment.

- data element usage rules

This is a table which defines:

- the ***usage rules for the data element***, including its status:

mandatory: *the data element must be present;*

conditional: *the presence of the data element is conditional; the rules are explained in this table;*

optional: *the data element is not processed and, if sent, is for documentary purposes only.*

its ***format*** *(from the point of view of what the* ***generic GESMES syntax allows)*****:** the notation used is: an=alphanumeric, n=numeric, ".." = no more than; e.g. an4=four character (exact) alphanumeric, an..4=up to four characters alphanumeric. This is useful for syntax checking vis-à-vis the general GESMES rules.

the ***valid codes and their meaning (and/or concrete format specifications) in the SDMX-EDI context***; whenever there are references to code lists in the description of the segments, these refer to the code lists described in relevant documents or files provided by the data exchange centre institution. (see "Code lists" paragraph further below).

Content and general usage rules of the segment

This provides a description of the role of the segment and rules in the context of SDMX-EDI.

- dependencies: This describes dependencies with other segments, apart from the ones within-the-group.

Code lists

In some of the presented segments there are references to "code lists". Indeed, some "coded" elements need to take appropriate values from predefined code lists. These are given either as *concrete (mandatory for use)* code lists or as *recommendable* code lists or as *example* code lists.

850
851
852
853
854

9.2 Interchange Administration Section

9.2.1 Scope

The administration data at the level of the interchange comprises the identification of the syntax version and character set used, the sender and receiver identification and the date and time of the preparation of the interchange.

The Interchange Administration has to be present in any interchange SDMX-EDI file.

9.2.2 GESMES Segments Used

This comprises the UNA and UNB segments. These segments are not a part of the message structure, they consist instead of the envelope header for the interchange, which can contain many GESMES messages.

9.2.3 Features and Guidelines for Use

One or more GESMES messages can be sent in one interchange.

9.2.4 Interchange Administration: Segment Structure

UNA*service-characters*

UNB+*syntax-identifier*:*syntax-version*+*sender-identification*+ *receiver-identification*+*date-of-preparation*:*time-of-preparation* +*interchange-reference*++*application-reference*'

Segment Type: **UNA**

Segment Type Name: *Syntax Character Specification*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

UNA*syntax-characters*

Data Element Rules

local name	usage rules	format	code values
syntax-characters	mandatory	an6 *(including the segment terminator)*	*In SDMX-EDI these characters have to be:* :+.? '

Content and General Usage Rules

UNA contains the specification of the data element separator and segment terminator characters used. The character repertoire for SDMX-EDI is UNOC (see relevant discussion in the description of the next segment). In order to use the recommended service character set with the UNOC character repertoire it is necessary to specify the service characters in the UNA segment as detailed below. In SDMX-EDI the UNA segment has a fixed syntax and it is written always as: UNA:+.? '

(=UNA followed by colon, plus sign, dot, question mark, space and single quote; the single quote is the character corresponding to the ANSI/ASCII character 39). The UNA statement, as it is given above, means:

the *component separator* used is : (colon), the *data element separator* is + (plus sign), the *decimal sign* is . (dot), the *release indicator* is ? (question mark), space is reserved for possible future use and ' (single quote; ASCII/ANSI 39) is used as the *segment terminator*.

The reading and writing applications should have these characters parameterised (at least the service characters: colon, plus sign, release indicator). For the proper use of the release character see the appendix on the EDIFACT syntax and the special chapter on the text contained in uncoded attributes (the release character is not needed before the point, as the point is not an EDIFACT service character).

Example:

It has to be written always as: `UNA:+.? '`

ATTN!

(1) There is a space between ? and '.

(2) In SDMX-EDI only the point (.) is used for the decimal notation.

(3) The last character (') is part of the segment (and segment terminator by “coincidence”).

Segment Type: **UNB**

Segment Type Name: *Interchange header*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

UNB+*syntax-identifier*:*syntax-version*+*sender-identification*+ *receiver-identification*+*date-of-preparation*:*time-of-preparation* +*interchange-reference*++*application-reference*++++*test-indicator*'

Data Element Usage Rules

local name	usage rules	format	code values
syntax-identifier	mandatory this identifies the character set used in the interchange	an4	UNOC - (ISO 8859-1) this supports accented characters and the text can be in any European language except Greek
syntax-version	mandatory this is 3, which is the latest version of the syntax	n1	3
Sender-identification	mandatory it identifies the sender.	an..35	Central institution administrating the data exchange provides an appropriate code list for identifying partner organisations.
Receiver-identification	Mandatory it identifies the receiver	an..35	As above.
Date-of-preparation	Mandatory this is the date of preparation of the interchange the format is YYMMDD	n6	e.g. 970525 (Attn! when 2000, then YY will be 00)
Time-of-preparation	Mandatory this is the local time of preparation of the interchange the format is HHMM	n4	e.g. 0950
Interchange-reference-number	mandatory a unique reference which identifies the interchange, incl. a serial incremental number (see also usage rules below)	an..14	in SDMX-EDI the format is: IREFnnnnnn e.g. IREF000001
Application-reference	mandatory	an..14	SDMX-EDI
Test indicator	conditional if found, then the contents should not be used to update or modify databases, as the interchange serves	n1	1

	only testing purposes		

Content and General Usage Rules

This segment is the envelope header for one or more messages and contains administration and routing information which can be used by a message transfer service. *UNOC:3* is the syntax identifier referring to the character set used in the interchange (in SDMX-EDI it has been set equal to UNOC:3 which allows the use of upper and lower case characters in the message, including the accented Latin characters).

The *date-of-preparation* and *time-of-preparation* (fixed format for both together: YYMMDD:hhmm) refers to the date and the local time that the interchange file was produced by the system of the sender.

The *interchange reference number* is a serial (incremental by one) integer number produced by the sender. It is strongly recommended, for each institution sending a file to a specific receiver, to increment this counter by one (keeping in each sending institution different counters for each receiving institution); this is the only safe way to detect not only duplicate interchanges but also missing interchanges sent to a centre. The number part (full format: IREFnnnnnn) is six characters long, starting with 000001, 000002, ...etc.; the value IREF999999 will be succeeded by IREF000000 and then by IREF000001 (starting again from the beginning). A *test interchange* should also be flagged with an interchange reference number incremented by one vis-à-vis the previous interchange towards the same receiving institution.

Important remark: When a file contains live data, which are supposed to be used to update receiver's live databases, the UNB segment ends with the element "++SDMX-EDI". However, for files that should not be used by the receiver to update live databases (e.g. test files), the segment must be terminated with the test indicator component (++++1).

Dependencies

The same *interchange reference number* has to appear also at the end of the message, in the UNZ segment.

Example 1 - exchange of live data:

```
UNB+UNOC:3+BE2+4F0+950825:1236+IREF000006++SDMX-EDI'
```

In this example the values BE2 (=National Bank of Belgium) and 4F0 (=European Central Bank) have been taken from a code list called CL_ORGANISATION (commonly agreed and administrated by the ECB, Eurostat/BoP and the BIS; other data exchange circuits may need to define and use a different code list). The interchange file was produced on the 25th Aug. 1995 (at 12:36) and it is the 6th interchange of this institution via which it is sending data to the ECB.

Example 2 - exchange of test data:

```
UNB+UNOC:3+BE2+4F0+950825:1236+IREF000007++SDMX-EDI++++1'
```

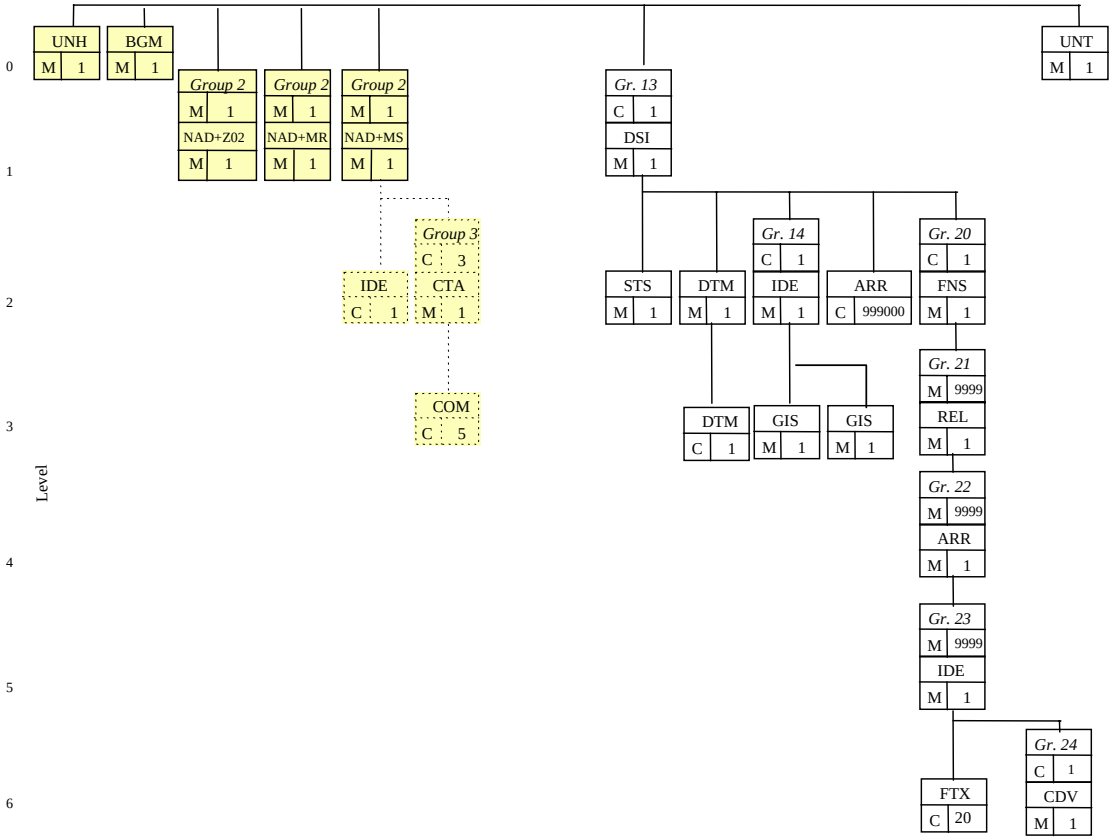
As before, but, due to presence of the test indicator (++++1), the contents of the interchange should not be used to update or to modify databases at the receiving end as this interchange serves only testing purposes.

9.3 Message Administration Section

9.3.1 Scope

This contains the administration data for the message (shaded boxes in the following branching diagram).

9.3.2 GESMES Segments Used



9.3.3 Features and Guidelines for Use

UNH identifies the message type and version.

BGM identifies the message purpose either as containing a data set (or references to a data set) or as containing structural data.

NAD three occurrences, one for the identification of the code list maintenance agency (NAD+Z02), one for the identification of the receiver (NAD+MR) and the third one for the identification of the sender (NAD+MS). The last one may trigger an IDE segment and/or Group 3 (CTA and possibly COM). The IDE segment provides a message identity (assigned by the sender) and Group 3, which can be repeated up to three times, provides detailed communication

information about the sender (if used by the sender, then the sending institution chooses its one, two or three most relevant contact people).

The Message Administration Group is necessary in all types of messages (the three NAD segments are mandatory).

An interchange can contain more than one messages; therefore, every message within the interchange should carry a Message Administration Group.

9.3.4 Message Administration Section: Segment Structure

UNH+*message-reference-number*+*message-type*:*message-type-version*:*message-type-release*:*controlling-agency*'

BGM+*message-name*'

NAD+Z02+*organisation-id*' (code list maintenance agency identifier)

NAD+MR+*organisation-id*' (receiver identifier)

NAD+MS+*organisation-id*' (sender identifier)

IDE+10+*message-id*' (message identification provided by the sender)

CTA+*contact-function*+*contact-id*:*contact-name*' (contact information)

COM+*communication-number*:*communication-channel*' (com. number and type of channel)

Segment Type: **UNH**

Segment Type Name: *Message Header*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

UNH+*message-reference-number*+*message-type*:*message-type-version*:*message-type-release*:*controlling-agency*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
message-reference-number	mandatory reference that must be unique within the interchange using a serial (incremental by one) integer number produced by the sender (see also rules below).	an..14	*format:* MREFnnnnnn
message-type	mandatory	an..6	GESMES
message-type-version	mandatory	an..3	2
message-type-release	mandatory	an..3	1
controlling agency	mandatory	an..2	E6

Content and General Usage Rules

This segment starts the message and contains data which identify the message type and a unique reference which can be used to validate the integrity of the message. In SDMX-EDI the only non-fixed element is the *message reference number*. The message reference number (format: MREFnnnnnn) is a serial (incremental by one) integer number produced by the sender for each message in the current interchange (an interchange can include more than one message). The part of the pure number (nnnnnn) is six characters long, starting with MREF000001, MREF000002, ...etc.; no more than 999,999 messages can be included in one interchange. The first message reference number should be 000001.

Dependencies

The same message reference number has to appear also at the end of the message, in the UNT segment.

Example:

UNH+MREF000001+GESMES:2:1:E6'

Segment Type: **BGM**

Segment Type Name: *Beginning of Message*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

BGM+*message-name*'

Data Element usage Rules

local name	usage rules	format	SDMX-EDI code values
message-name	mandatory **73** - statistical definitions this is used if the message contains only statistical definitions (e.g. structure definitions, code lists) and there are no data sets in the message (i.e. no occurrence of the DSI group) **74** - statistical data this is used to indicate that data and/or attributes are sent in the message or that the message contains delete references; therefore, the message contains the DSI group. **DSL** - data set list the message contains only a list of data set identifiers and the key family definitions that describe their structure.	an..3	73 - statistical definitions 74 - statistical data DSL - data set list

Content and General Usage Rules

This segment identifies the function of the message.

Dependencies

If the *message-name* has been set to 74, then the data set administration section has to be present and no structural sections (e.g. code lists, key family definition) should be included in the message. The reading applications should expect numeric data and/or attributes in this message. On the other hand, if the *message-name* has been set to 73, the data set administration section should not be present and the reading application should conclude that the message contains structural definitions (lists of concepts or code lists or key family definitions).

Example 1:

BGM+74'

This message contains or it refers to data (observations) and/or attributes.

Example 2:

BGM+73'

This message contains only code lists and/or statistical concept definitions and/or key family definitions.

Segment Group: **Group 2 (NAD)**

Max. Number of Occurrences: *3*

Status: *Mandatory*

Content and General Usage Rules

The NAD group is used to provide:

(1) the identity of the organisation maintaining the code lists and the key family definitions,

(2) the identity of the receiving organisation and

(3) the identity of the sending organisation (and, possibly, more information provided by the sender).

Each piece of information is provided by a simple NAD segment (NAD+Z02+..., NAD+MR+... and NAD+MS+... respectively). The third one (NAD+MS+...) might be followed by an IDE segment and/or up to three occurrences of Group 3 (CTA-COM) which give further information related to the sender.

Segment Type: **NAD** **First occurrence - ref. to code lists administration agency**

Segment Type Name: *Name and Address*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

NAD+*party-type*+*organisation-id*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
Party-type	mandatory Z02 indicates that this NAD segment refers to the party maintaining the default code lists	an..3	Z02 (fixed) Maintenance agency for code lists and k.f. definitions
Organisation-id	mandatory the code list maintenance agency id is given here	an..35	Examples: BIS ECB EUROSTAT IMF OECD

Content and General Rules

This is used to provide the identity of the organisation maintaining the code lists and key family definition used in the message. The sending institution chooses appropriately this parameter in accordance with the guidelines of the centre whose statistical concepts, code lists and structures are used in the message. For a more extensive discussion on the practical role and the use of the "maintenance agency" concept, the reader is advised to refer to the paragraph on central institutions.

Example 1 :

NAD+Z02+EUROSTAT'

This is how the segment should be written if the message uses/refers to structural definitions administrated by Eurostat.

Example 2 :

NAD+Z02+BIS'

Example 3:

NAD+Z02+ECB'

Segment Type: **NAD** **Second occurrence - reference to the message receiver**

Segment Type Name: *Name and Address*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

NAD+*party-type*+*organisation-id*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
Party-type	mandatory *party-type* is MR	an..3	MR - message receiver
Organisation-id	mandatory message receiver	an..35	A code value for identifying the receiving organisation; it is provided by the statistical centre administrating the data exchange.

Content and General Rules

This is used to provide the identity of the receiving organisation.

Example:

NAD+MR+4F0'

Here, the message is supposed to be sent to the European Central Bank (here coded as 4F0).

Segment Type: **NAD** **Third occurrence; reference to the message sender**

Segment Type Name: *Name and Address*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

NAD+*party-type*+*organisation-id*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
party-type	mandatory *party-type* is MS	an..3	MS - message sender
organisation-id	mandatory message sender	an..35	A code value for identifying the sending organisation; the underlying code list is provided by the statistical centre administrating the data exchange.

Content and General Rules

This is used to provide the identity of the sending organisation.

Example:

NAD+MS+BE2'

The message is sent by the National Bank of Belgium (here coded as BE2).

Segment Type: **IDE**

Segment Type Name: *Message identity*

Max. Number of Occurrences: *1*

Status: *Conditional*

Data Element Sequence

IDE+*object-type*+*message-identity*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
object-type	mandatory *object-type* is 10	an..3	10 - message context
message-identity	mandatory message identity assigned by the sender of the message (see also rules below)	an..35 (see also rules below)	free text *E.g. Regular daily update, Regular monthly reporting, etc.*

Content and General Rules

This message identity is provided by the sender. It is free text and in general it can be up to 35 characters long (e.g. “Daily update”, “Monthly tape”).

Example 1:

IDE+10+Quarterly BoP reporting'

The sender has identified this message as “Regular quarterly BoP reporting”.

Segment Group: **Group 3 (CTA-COM)**

Max. Number of Occurrences: *3*

Status: *Conditional*

Content and General Usage Rules

Each occurrence of this group is used to provide details about a contact that is relevant to the sending organisation identified in the preceding NAD+MS segment. These details include the name of the contact and, optionally, contact numbers such as telephone, fax, e-mail.

This Group is conditional. Certain data exchange circuits might find it useful and could agree to regularly use it in the exchanged messages.

As Group 3 can be repeated three times, if the sender made use of it, he could choose the (maximum) three most relevant people whose details could be useful to the receiver.

Segment Type: **CTA**

Segment Type Name: *Contact information*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

CTA+*contact-function*+*contact-id*:*contact-name*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
contact-function	mandatory It identifies the function of the person (on the sending side) whose name follows	an..3	CC - responsible person for information production CP - responsible person for computer data processing CF - Head of unit for information production CE - Head of unit for computer data processing
contact-id	optional the identity of the contact (dept. id) as known in sender's side	an..17	e.g. BoP, M&B, EDP, ICSD etc.
contact-name	mandatory the name of the contact person	an..35	e.g. John Smith

Content and General Rules

It identifies a contact name at the sending institution.

Example 1:

CTA+CC+:Mr John Smith'

Mr. John Smith is the person responsible for the compilation and production of the data at the sending institution.

Example 2:

CTA+CP+IS/BoP:Mr John Smith'

Mr. John Smith is the person responsible for data computer processing at the IS/BoP unit in the sending institution.

Segment Type: **COM**

Segment Type Name: *Communication number and type of channel*

Max. Number of Occurrences: *5*

Status: *Conditional*

Data Element Sequence

COM+*communication-number*:*communication-channel*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
communication-number	mandatory Telephone or fax number or e-mail address etc.	an..512	e.g. 0049 69 1344 0
communication-channel	mandatory	an..3	EM - e-mail TE - telephone FX - fax XF - X.400

Content and General Rules

It provides a contact number for the person indicated in the preceding CTA segment and it identifies the type of this number (communication channel).

Example 1:

COM+0049 69 13440:TE'

Example 2:

Combining with previous segments:

NAD+MS+4F0'
IDE+10+ECB-monthly disseminated aggr. data'
CTA+CC+Money and Banking:Mr John Smith'
COM+0049 69 13440:TE'
COM+0049 69 13446000:FX'
COM+jsmith@test.com:EM'
CTA+CP+IS/M&B:Mr Klaus Roberts'
COM+0049 69 1344888:TE'

This example shows the use of the whole (conditional) group following NAD+MS. The ECB (=4F0) provides info about the message contents (IDE segment) and people to contact (CTA/COM segments).

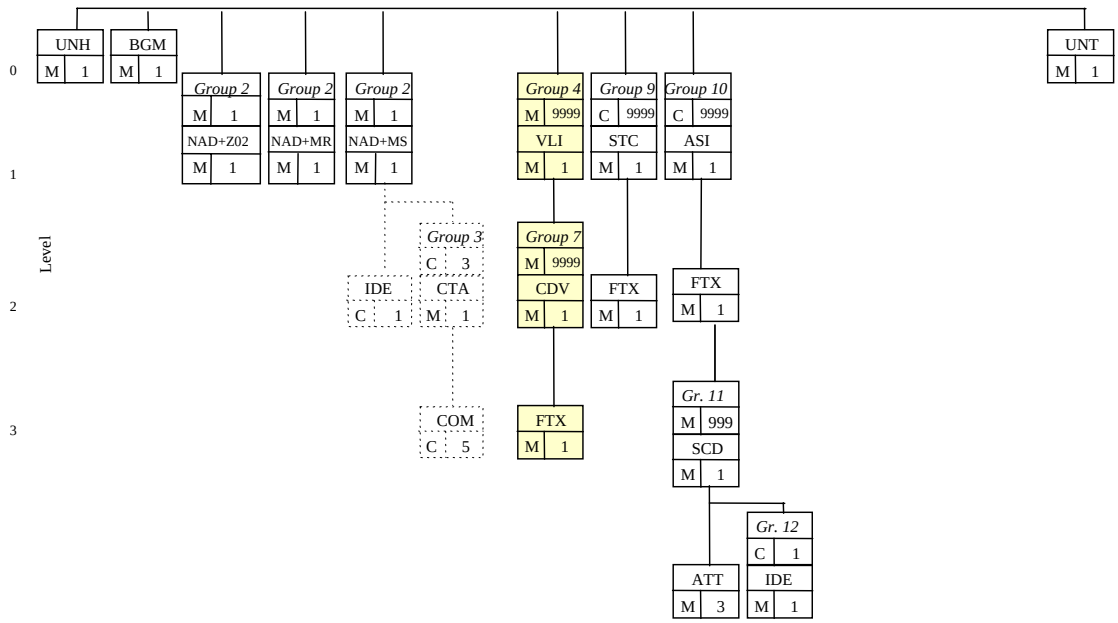
9.4 Code List Section

9.4.1 Scope

The dissemination of the code lists from the centre enables the receiving institutions to build tables with the possible values for all coded statistical concepts (dimensions used in key families and coded attributes). References to the code lists can be used by any of the data sets sent in other SDMX-EDI messages.

9.4.2 GESMES Segments Used

In SDMX-EDI data and attributes (DSI Group) cannot be mixed in the same message with the groups of segments carrying the structural definitions (i.e. code lists-VLI, statistical concepts-STC, key family definitions-ASI). Therefore, the following branching diagram presents a typical SDMX-EDI structural message carrying code lists, statistical concept definitions and key family definitions. The shaded boxes are the ones used for the exchange of code lists which is discussed in this section; that it is why Group 4 has been flagged here as mandatory.



9.4.3 Features and Guidelines for Use

The list is sent in Group 4. The list itself must be given a unique identifier and this is sent in the VLI segment. The individual code values are sent in the CDV segment (one code in each occurrence of a CDV segment) and their textual description is given in the FTX segment (one textual description for each occurrence of a CDV segment).

1206

1207 ### 9.4.4 Code List Section: Segment Structure

1208 VLI+*code-list-id*+++*code-list-name*'

1209 CDV+*code-value*'

1210 FTX+ACM+++*textual-description*'

1211

Segment Group: **Group 4 (VLI-Group 7)**

Max. Number of Occurrences: *9999*

Status: *Conditional*

Content and General Usage Rules

A group of segments containing a code list. It can be stored by the receiver on a metadata database (see also about alternative ways of organising structural definitions in Section 3.4.1.).

Segment Type: **VLI**

Segment Type Name: *Value List Identification*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

VLI+*code-list-id*+++*code-list-name*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
code-list-id	mandatory the identity of the code list	an..18	Code list identifier as provided by a centre institution, e.g.: CL_UNIT
code-list-name	mandatory the name of the code list	an..70	Short description (name) of a code list provided by a centre institution, e.g.: Units code list

Content and General Usage Rules

The segment is used to provide a unique identifier and a name for a code list.

Example 1:

VLI+CL_ADJUSTMENT+++Adjustment code list'

A code list called CL_ADJUSTMENT will be given via the segments which follow (CDV, FTX).

Example 2 :

VLI+CL_ORGANISATION+++Organisation code list'

A code list called CL_ORGANISATION will be given via the segments which follow (CDV, FTX).

Segment Group: **Group 7 (CDV-FTX)**

Max. Number of Occurrences: *9999*

Status: *Mandatory*

Content and General Usage Rules

A group of segments containing the code values in a code list and their descriptions.

The CDV segment occurs once for each code and the FTX segment once for each CDV.

Segment Type: **CDV**

Segment Type Name: *Code Value*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

CDV+*code-value*'

Data Element Usage Rules

local name	usage rules	Format	SDMX-EDI code values
code-value	mandatory the code value	an..18	A code value from a code list provided by a centre institution.

Content and General Rules

This is used to provide a code value.

Example:

CDV+DK2'

This is a code value from the CL_ORGANISATION code list (mentioned in a previous example when demonstrating VLI).

Segment Type: **FTX**

Segment Type Name: *Free Text (code value textual description)*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

FTX+*text-subject*+++*code-value-description*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
text-subject	Mandatory in SDMX-EDI: "statistical description"	an..3	ACM (=statistical description)
code-value-description	Mandatory the text of the code value	an..350	The textual description of the meaning of a code value. Expression for maximum length: text:text:text:text:text where each "text" can be up to 70 characters long.

Content and General Rules

This is used to provide the description of a code value.

Each occurrence of this segment is used to provide the description of a code value. It can be up to (total) 350 characters long. However, if the description is longer than 70 characters (upper limit for a single component element), then the *text* should be split in (the maximum) 5 component element-texts (each one of them not being longer than 70 characters and separated among them with a *component separator*). The reading applications should read the components of the received text and consider that they form a string which is the result of the added sub-strings. The writing applications should break the text into pieces of 70 characters or shorter; it does not matter whether a word is broken between two component elements or not (see also the Appendix on the EDIFACT syntax).

Briefly, the code value description can be given as *free-text*:*free-text*:*free-text*:*free-text*:*free-text*' where *free-text* can be up to 70 characters long and can repeat up to five times. See also section 6.9.9.

Example:

FTX+ACM+++Danmarks Nationalbank'

This is the description explaining the code value shown in the previous example demonstrating CDV.

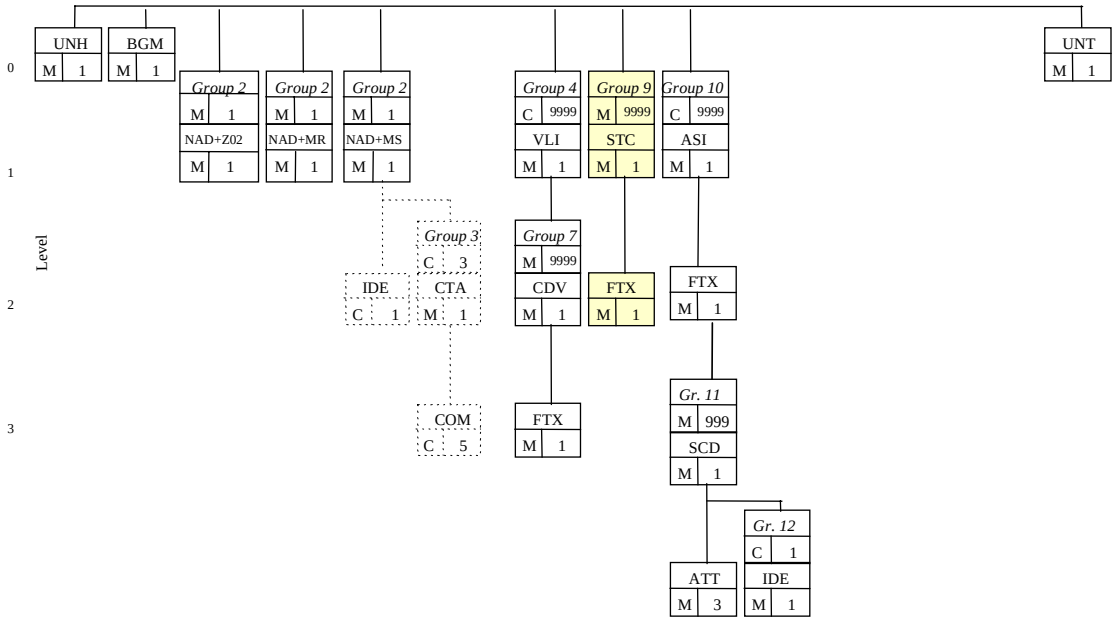
9.5 Statistical Concept Definition Section

9.5.1 Scope

This provides a list of the statistical concepts used by the partners together with their names.

9.5.2 GESMES Segments Used

In the following SDMX-EDI branching diagram the shaded boxes are used in the definition of the concepts. Group 9 is shown here as mandatory, as without it is not possible to disseminate statistical concept definitions.



9.5.3 Features and Guidelines for Use

A pair of segments is used; the first segment (STC) provides the identifier of the statistical concept and the second one (FTX) its name.

9.5.4 Statistical Concept Definition Section: Segment Structure

STC+*concept-identifier*'

FTX+ACM+++*concept name*' *repeats*

Segment Group: **Group 9 (STC-FTX)**

Max. Number of Occurrences: *9999*

Status: *Conditional*

Content and General Usage Rules

A pair of segments describing a statistical concept.

Segment Type: **STC**

Segment Type Name: *Statistical concept*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

STC+*concept-identifier*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
concept-identifier	mandatory the identifier of the statistical concept	an..18	A statistical concept identifier as provided by a centre institution. e.g.: UNIT

Content and General Usage Rules

The segment identifies the statistical concept.

Example:

STC+SOURCE_AGENCY'

The *name* of a concept, whose identifier is SOURCE_AGENCY, will be given in the FTX segment which follows.

Segment Type: **FTX**

Segment Type Name: *Free Text (name of the concept)*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

FTX+*text-subject*+++*code-label*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
text-subject	mandatory in SDMX-EDI: "statistical description"	an..3	ACM (=stat.description)
code-label	mandatory name of the statistical concept	an..70	A name for the concept specified on the previously given STC segment.

Content and General Rules

This is used to provide the name of the concept stated in the previous segment (STC).

Example:

FTX+ACM+++Source agency'

This is the name of the concept shown in the previous example (see description for STC).

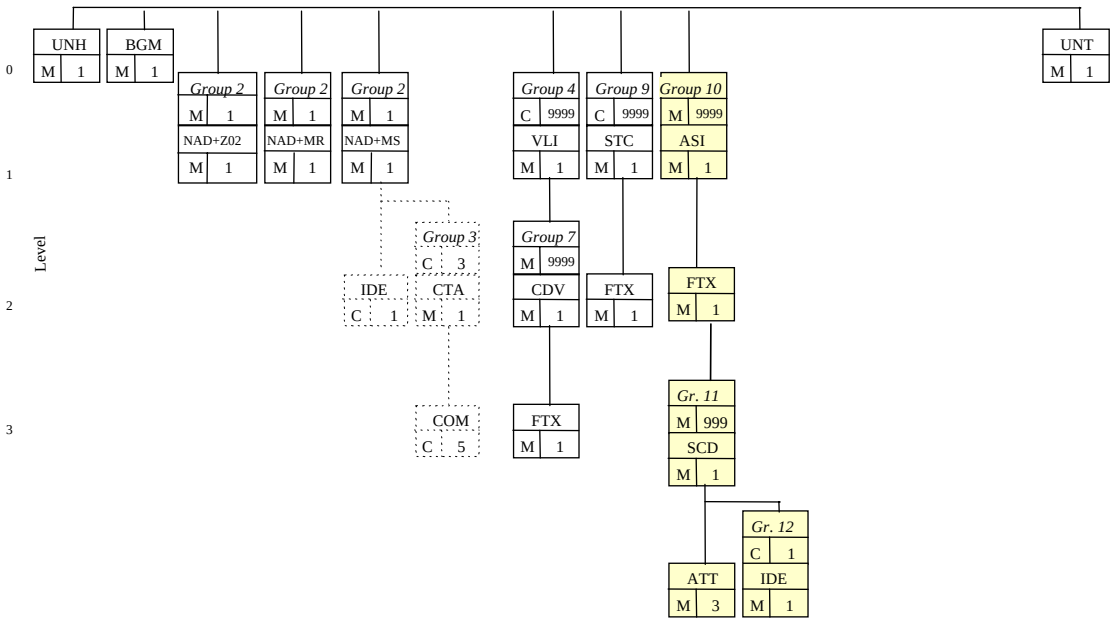
9.6 Key Family Definition Section

9.6.1 Scope

This section allows the organisation receiving the corresponding structural message to interpret key family definitions (which dimensions and at which positions) and the corresponding lists of relevant attributes. It is obvious that the key family definition could not contain structure components (i.e. a statistical concept) which have not been already described in the "exchanged" concepts (e.g. via Group 8): each concept id used in a key family definition should also become *known* to the partner institutions.

9.6.2 GESMES Segments Used

When the parties exchange key family definitions, they have to use Group 10 (the shaded boxes show the segments involved); that is why this group is shown here as mandatory:



9.6.3 Features and Guidelines for Use

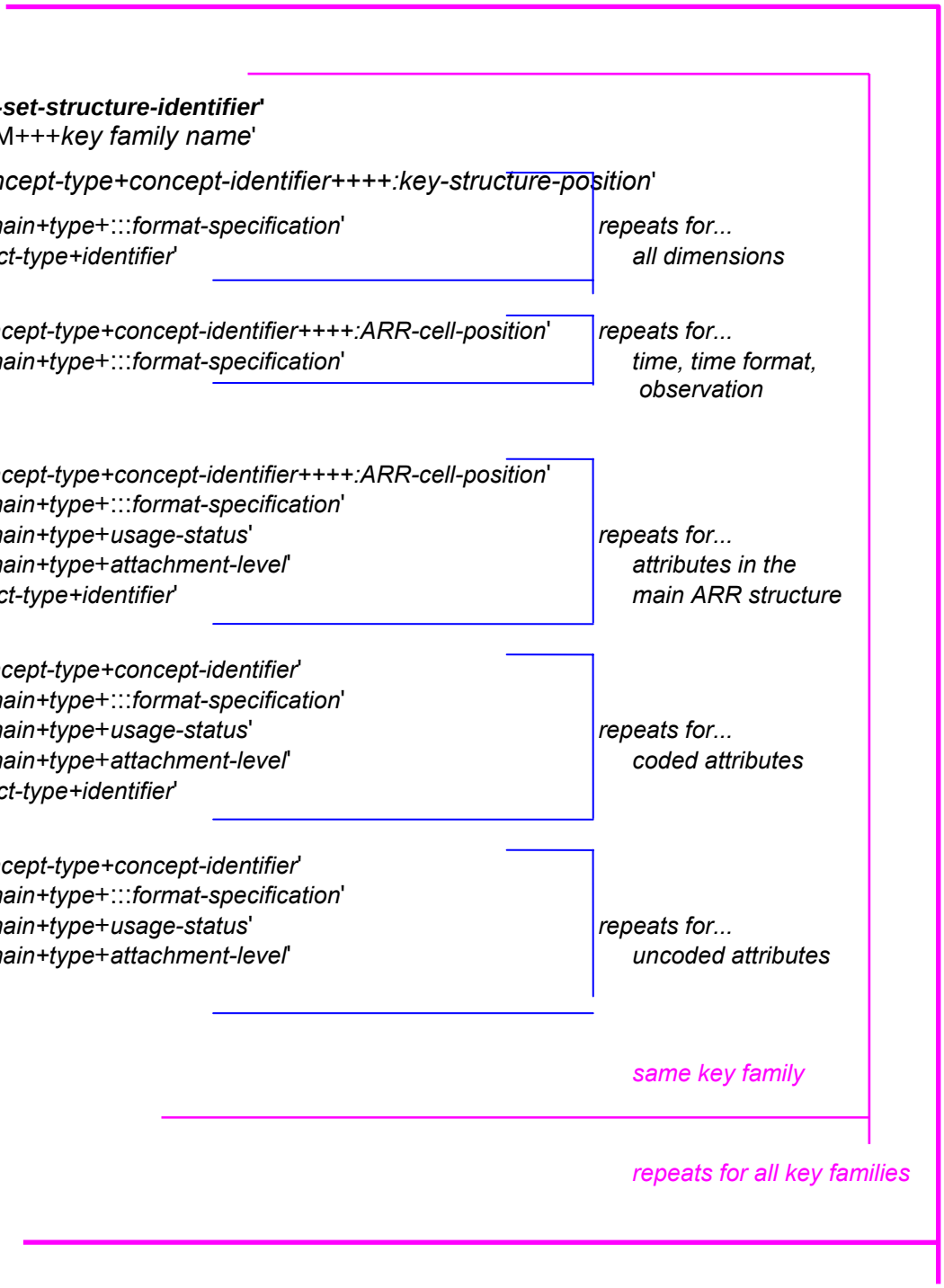
A key family definition is reflected in one occurrence of the ASI segment. A FTX segment provides the name of the key family. Each structure component is identified as an occurrence of the SCD segment, in terms of the statistical concept identifier (and the position of the dimension in the key structure if the concept is a dimension). In general, a complete key family definition comprises:

the identification of all dimensions of this key family and also of all the other relevant observation component elements (**this structural information is needed for the interpretation of the ARR segment following Group 14 - not shown here - which holds the data to be exchanged).**

the identification of all the attributes (mandatory and non-mandatory) which are in use within this key family (SCD segment), the definition of their status and their attachment level;
the description of the format of the values of the statistical concepts (ATT segment) and

the identification of the corresponding code list (IDE) if the statistical concept is coded.

9.6.4 Key Family Definition Section: Segment Structure



Segment Group: **Group 10 (ASI-Group 11)**

Max. Number of Occurrences: *9999*

Status: *Conditional*

Content and General Usage Rules

Each occurrence of this group is used to provide the definition of a key family (its dimensions and attributes). The key family may be new or it may already exist on the receiver's database, in which case the definition in this group should replace the definition in the existing metadata database.

Segment Type: **ASI**

Segment Type Name: *Array Structure Identification*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Content and General Usage Rules

This is used to identify the data set structure (i.e. array structure).

Data Element Sequence

ASI+*data-set-structure-identifier*'

Data Element Usage Rules

Local name	usage rules	format	SDMX-EDI code values
Data-set-structure-identifier	mandatory	an..18	the key family identifier

Content and General Rules

This is used to provide the structure of the data set: in SDMX-EDI the identification of the data set structure is provided by a "key family identifier".

Example:

ASI+ESCB_BAL_OF_PAYM01'

Segment Type: **FTX**

Segment Type Name: *Free Text (name of a key family)*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

FTX+*text-subject*+++*code-label*'

Data Element Usage Rules

local name	Usage rules	Format	SDMX-EDI code values
text-subject	Mandatory in SDMX-EDI: "statistical description"	an..3	ACM (=stat.description)
code-label	Mandatory name of the statistical concept	an..70	The name (short description) of the key family specified on the previously given ASI segment.

Content and General Rules

This is used to provide the name of the key family stated in the previously given segment (ASI).

Example:

FTX+ACM+++ECB kf for BoP data'

This is the name of the key family shown in the previous example (see description for ASI).

Segment Group: **Group 11 (SCD-ATT-Group 12)**

Max. Number of Occurrences: *999*

Status: *Conditional*

Content and General Usage Rules

This segment group identifies the statistical concepts of the key family and defines them as either dimensions or attributes. Dimensions must be coded, while attributes may be or not.
In addition, it provides information required for the interpretation of the ARR segment following group 14:

- time dimensions: "time period" ("To which periods do the data in the ARR segment relate?") and "time format" ("Do the data in the ARR segment relate to a single period or to a range and what is the frequency?")
- array cells: observation value, the observation level attribute observation status (mandatory) and the observation level attributes observation confidentiality and observation pre-break (both conditional).

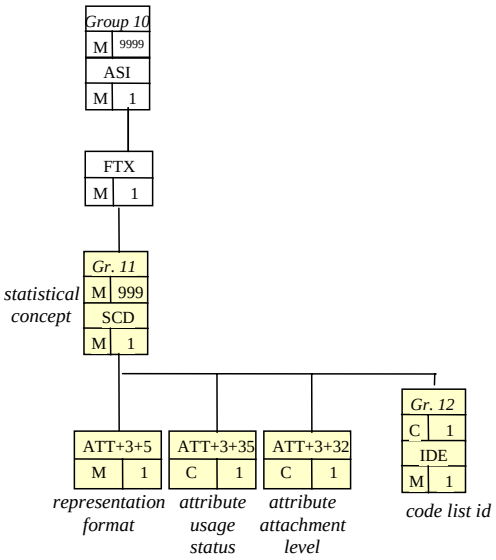
These categories of definitions are elaborated below:
Dimension: For each dimension of the key family, (1) its position in the key structure, (2) its representation (length of the code values and whether they are numeric or alphanumeric) and (3) the name of the relevant code list are provided.

Time period, time format, observation value and "array" attributes: These are defined by their representation and their position in the array structure. Moreover, for the three observation level attributes ("array attributes": observation status, observation confidentiality, observation pre-break value), also the attachment level, their usage status and (but not for the "observation pre-break") the relevant code lists are provided.

Attribute definitions: All attributes for the key family are defined with the information on

- whether they are coded or uncoded
- their usage status (mandatory or conditional)
- their attachment level (data set, sibling series, time series or observation)
- representation: field length (exact or maximum length) and the information whether they are numeric or alphanumeric
- relevant code list (for coded attributes) .

As the ATT segment is used for three different purposes (that it is why in the branching diagram it is shown that it can be repeated up to three times), for reasons of clarity in the following pages it is presented separately, for each one of its three usages. The diagram below provides a “zoom” view on the group:



Segment Structure

SCD+concept type+concept identifier++++:position in key (or array structure)'
ATT+domain+type+:::format specification' (*for the representation*)
ATT+domain+type+code:code list qualifier' (*for the usage status of attributes only*)
ATT+domain+type+code:code list qualifier' (*for the attachment level of attributes only*)
IDE+object type+identifier' (*code list for dimensions, coded attributes*)

Note

The examples for the various segments to be used in the key family definition section are derived from the example key family defined in the next paragraph. The examples are built up in such a way that the reader can with each added segment retain the overall view of the whole group. The complete section with required segments is then presented following the presentation of the IDE segment (group 12).

Segment Type: **SCD**

Segment Type Name: *Structure Component Definition*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

SCD+*concept-type+concept-identifier++++:position-in-key-structure*'

Data Element Usage Rules

local name	usage rules	Format	SDMX-EDI code values
concept-type	Mandatory	an..3	1 - time 3 - array cell 13 - dimension "frequency" in key family's structure 4 - dimension in key family's structure (other than "frequency") Z09 - attribute
concept-identifier	Mandatory the identifier of the statistical concept	an..18	Provided by centre institution.
position-in-key-structure	Conditional this gives the position of the statistical concept (as a dimension) in the key family structure (e.g. a value of 1 is the first position), if the concept is a dimension or element of the ARR structure.	an..6	1 - first position 2 - second position ... etc.

Content and General Rules

It is used to identify the statistical concepts used in the key family, their type (dimension or attribute) and in the case of dimensions to define their position in the key structure. In addition, apart from the dimensions, it defines the array cell position in the ARR segments of the supplementary elements, i.e. time dimension, time format, observation, observation status (and confidentiality and observation pre-break, if used in the key family).

Example:

Let's consider the definition of the following (example) key family:

Statistical Concept	Identifier	alphanum.(an)	concept role	Position in the key / *array struct. position*
DIMENSIONS:				
Frequency	FREQ	an1	dimension	**1**
Reporting/ref. country	REF_AREA	an2	dimension	**2**
IMF Bal.of Paym.	BOP_ITEM	an3	dimension	**3**
Type of data	DATA_TYPE	an1	dimension	**4**
Time period	TIME_PERIOD	an..35	*time dimension*	**5**
Time format	TIME_FORMAT	an3	*time dimension*	**6**
Observation	OBS_VALUE	an..15	*observation*	**7**
Observation status	OBS_STATUS	an1	*array cell*	**8**
Obs. confidentiality	OBS_CONF	an1	*array cell*	**9**
Pre-break value	OBS_PRE_BREAK	an..15	*array cell*	**10**
ATTRIBUTES:				
Title	TITLE	an..70	attribute	-
Unit	UNIT	an..12	attribute	-
Unit multiplier	UNIT_MULT	an..2	attribute	-

(In this key family it has been assumed that the observation confidentiality and observation pre-break have been included in the array structure of the key family; this might not be the case for other key families)

The structure definition segments for this key family would be:

SCD+**13**+FREQ++++:1'... *(corresponding ATT and IDE segments should follow)* ...

SCD+**4**+REF_AREA++++:2'

... *(corresponding ATT and IDE segments should follow)* ...

SCD+**4**+BOP_ITEM++++:3'

... *(corresponding ATT and IDE segments should follow)* ...

SCD+**4**+DATA_TYPE++++:4'

... *(corresponding ATT and IDE segments should follow)* ...

Other component elements (time dimension and array cells) present in the ARR segments for this particular key family:

SCD+**1**+TIME_PERIOD++++:5'

(corresponding ATT segment should follow)

SCD+**1**+TIME_FORMAT++++:6'

...*(corresponding ATT segment should follow)*...

SCD+**3**+OBS_VALUE++++:7'

...*(corresponding ATT segment should follow)*...

SCD+**3**+OBS_STATUS++++:8'

...*(corresponding ATT and IDE segments should follow)*...

SCD+**3**+OBS_CONF++++:9'

...*(corresponding ATT and IDE segments should follow)*...

SCD+**3**+OBS_PRE_BREAK++++:10'

...*(corresponding ATT segments should follow)*...

Attributes used in this key family (**note that only *concept-type* and *concept-identifier* are used here**):

SCD+Z09+TITLE'

...*(corresponding ATT segments should follow)*...

SCD+Z09+UNIT'

... *(corresponding ATT and IDE segments should follow)*...

SCD+Z09+UNIT_MULT'

...*(corresponding ATT and IDE segments should follow)*...

Segment Type: **ATT** **First occurrence - ref. to representation format**

Segment Type Name: *Attribute*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

ATT+*domain*+*type*+:::*format-specification*'

Data Element Usage Rules

Local name	usage rules	format	SDMX-EDI code values
domain	mandatory	an..3	3 - related to the key family definition
type	mandatory	an..3	5 - representation
format-specification	mandatory this gives the format (numeric or alphanumeric and the length) of the code values (AN*n*=exact, AN..*n*=up to *n* character long).	an..35	e.g. AN2 (=alphanumeric, exactly 2 char. long) AN..3 (=up to 3 characters long) etc.

Content and General Rules

It is used to inform about the format of the values taken by all the statistical concepts used in the key family.

Example:

Considering again the previous example:

Statistical Concept	Identifier	alphanum. (an) or numeric (n)	Concept role	Position in the key / *array struct. position*
Frequency	FREQ	an1	Dimension	1
Reporting/ref. country	REF_AREA	an2	Dimension	2
IMF Bal.of Paym.	BOP_ITEM	an3	Dimension	3
Type of data	DATA_TYPE	an1	Dimension	4
Time period	TIME_PERIOD	an..35	*time dimension*	5
Time format	TIME_FORMAT	an3	*time dimension*	6
Observation	OBS_VALUE	an..15	*observation*	7
Observation status	OBS_STATUS	an1	*array cell*	8
Obs. confidentiality	OBS_CONF	an1	*array cell*	9

Pre-break value	OBS_PRE_BREAK	an..15	*array cell*	10
Title	TITLE	an..70	Attribute	-
Units	UNIT	an..12	Attribute	-
Unit multiplier	UNIT_MULT	an..2	Attribute	-

1562

1563 The structure definition segments (SCD and the representation defintion in ATT) would now look
1564 as follows:
1565
1566

SCD+13+FREQ++++:1'
ATT+3+5+:::AN1'
....

IDE segment to follow

SCD+4+REF_AREA++++:2'
ATT+3+5+:::AN2'
....

IDE segment to follow

SCD+4+BOP_ITEM++++:3'
ATT+3+5+:::AN3'
....

IDE segment to follow

SCD+4+DATA_TYPE++++:4'
ATT+3+5+:::AN1'
....

IDE segment to follow

SCD+1+TIME_PERIOD++++:5'
ATT+3+5+:::AN..35'

No further segments relating to TIME_PERIOD

SCD+1+TIME_FORMAT++++:6'
ATT+3+5+:::AN3'

No further segments relating to TIME_FORMAT

SCD+3+OBS_VALUE++++:7'
ATT+3+5+:::AN..15'

No further segments relating to OBS_VALUE

SCD+3+OBS_STATUS++++:8'
ATT+3+5+:::AN1'
....

Other ATT and IDE segments to follow

SCD+3+OBS_CONF++++:9'
ATT+3+5+:::AN1'
....

Other ATT and IDE segments to follow

SCD+3+OBS_PRE_BREAK++++:10'
ATT+3+5+:::AN..15'
....

Other ATT segments to follow

SCD+Z09+TITLE'

ATT+3+5+:::AN..70'	
....	Other ATT segment to follow
SCD+Z09+UNIT'	
ATT+3+5+:::AN..12'	
....	Other ATT and IDE segments to follow
SCD+Z09+UNIT_MULT'	
ATT+3+5+:::AN..2'	
....	Other ATT and IDE segments to follow

1567

Segment Type: **ATT** **Second occurrence - ref. to usage status**

Segment Type Name: *Attribute*

Max. Number of Occurrences: *1*

Status: *Conditional*

Data Element Sequence

ATT+*domain+type+code*:*code-list-qualifier*'

Data Element Usage Rules

Local name	usage rules	format	SDMX-EDI code values
Domain	mandatory	an..3	3 - related to the key family definition
Type	mandatory	an..3	35 - usage status
Code	mandatory this gives the status of the attribute; if it is mandatory, then it should be expected to have always a value	an..3	1 - conditional 2 - mandatory
Code-list-qualifier	mandatory this gives the code list from which the previous values are taken	an..3	USS – usage status

Content and General Rules

It has to be used for all attributes (and only for the attributes). This applies to the attributes which are exchanged in the main ARR structure (attached next to the observation) and to the attributes which are exchanged using the FNS group. (So, it does not apply to the definition of the observation value, which, implicitly is mandatory)

Example:

Taking again the previous example, the structure definition segments (SCD and the representation and usage status defintion in ATT) relating to the attributes only would now look as follows:

SCD+3+OBS_STATUS++++:8'

ATT+3+5+:::AN1'

ATT+3+35+2:USS' Usage status for **OBS_STATUS** is mandatory

.... Other ATT and IDE segments to follow

SCD+3+OBS_CONF++++:9'

ATT+3+5+:::AN1'

ATT+3+35+1:USS'	Usage status for **OBS_CONF** is conditional
....	Other ATT and IDE segments to follow
SCD+3+OBS_PRE_BREAK++++:10'	
ATT+3+5+:::AN..15'	
ATT+3+35+1:USS'	Usage status for **OBS_PRE_BREAK** is conditional
....	Other ATT segment to follow
SCD+Z09+TITLE'	
ATT+3+5+:::AN..70'	
ATT+3+35+2:USS'	Usage status for TITLE is mandatory
....	Other ATT segment to follow
SCD+Z09+UNIT'	
ATT+3+5+:::AN..12'	
ATT+3+35+2:USS'	Usage status for UNIT is mandatory
....	Other ATT and IDE segments to follow
SCD+Z09+UNIT_MULT'	
ATT+3+5+:::AN..2'	
ATT+3+35+2:USS'	Usage status for UNIT_MULT is mandatory
....	Other ATT and IDE segments to follow

1585

1586

1587

Segment Type: **ATT** **Third occurrence - ref. to the attachment level**

Segment Type Name: *Attribute*

Max. Number of Occurrences: *1*

Status: *Conditional*

Data Element Sequence

ATT+*domain+type+code*:*code-list-qualifier*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
domain	Mandatory	an..3	3 - related to the key family definition
type	Mandatory	an..3	32 - object link type (attachment level)
code	Mandatory this gives the attachment level of the attribute	an..3	1 - data set 4 - time series 5 - observation 9 - sibling group
code-list-qualifier	Mandatory this gives the code list from which the previous values are taken	an..3	ALV – attachment level

Content and General Rules

It has to be used for all attributes (and only for the attributes). This applies to the attributes which are exchanged in the main ARR structure (attached next to the observation) and to the attributes which are exchanged using the FNS group. (So, it does not apply to the definition of the observation value, which, implicitly relates to the observation level)

Example:

Using again the previous example, we assume that in this key family:

- the “observation status”, the “observation confidentiality” and the “observation pre-break” are defined at the observation level;
- the “title” is defined at the sibling group level and
- the “unit” and “unit multiplier” at the time series level.

The structure definition segments (SCD and the representation, usage status and attachment level definition in ATT) relating to the attributes only would now look as follows (dimensions and the time dimensions do not carry an attachment level).

SCD+3+OBS_STATUS++++:8'

ATT+3+5+:::AN1'

ATT+3+35+2:USS'

ATT+3+32+5:ALV'	Attachment of OBS_STATUS is at the observation level
....	IDE segment (identifying relevant code list) to follow
SCD+3+OBS_CONF++++:9' ATT+3+5+:::AN1' ATT+3+35+1:USS' **ATT+3+32+5:ALV'**	Attachment of OBS_CONF is at the observation level
....	IDE segment to follow
SCD+3+OBS_PRE_BREAK++++:10' ATT+3+5+:::AN..15' ATT+3+35+1:USS' **ATT+3+32+5:ALV'**	Attachment of OBS_PRE_BREAK is at the obs. level
SCD+Z09+TITLE' ATT+3+5+:::AN..70' ATT+3+35+2:USS' **ATT+3+32+9:ALV'**	Attachment of TITLE is at the sibling level
SCD+Z09+UNIT' ATT+3+5+:::AN..4' ATT+3+35+2:USS' **ATT+3+32+4:ALV'**	Attachment of UNIT is at the time series level
....	IDE segment to follow
SCD+Z09+UNIT_MULT' ATT+3+5+:::AN..2' ATT+3+35+2:USS' **ATT+3+32+4:ALV'**	Attachment of UNIT_MULT is at the time series level
....	IDE segment to follow

Segment Group: **Group 12 (IDE)**

Max. Number of Occurrences: *1*

Status: *Conditional*

Content and General Usage Rules

This group is used to identify the code list used by a coded statistical concept.

Segment Type: **IDE**

Segment Type Name: *Identity*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

IDE+*object-type+identifier*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
object-type	mandatory	an..3	1 - code list
Identifier	mandatory the identifier of the code list	an..18	Provided by centre institution.

Content and General Rules

This contains the identifier of the code list of the statistical concept specified in the preceding SCD segment if this concept is coded.

Example 1:

IDE+1+CL_FREQ'

Note that this IDE should be preceded by a relevant SCD segment (and one or three ATT segments depending on whether the concept is a dimension or an attribute). For example, to introduce the previous segment, the following segments could precede:

SCD+13+FREQ++++:1'

ATT+3+5+:::AN1'

Obviously, the IDE segment will not be used if the related statistical concept is uncoded.

Example 2:

Referring to the example presented in the previous pages, the complete structure definition section (SCD and the representation, usage status and attachment level defintion in ATT and conditional IDE segment) for dimensions, the time dimension, array cells and attributes would now look as follows:

SCD+13+FREQ++++:1'
ATT+3+5+:::AN1'
IDE+1+CL_FREQ'

SCD+4+REF_AREA++++:2'
ATT+3+5+:::AN2'
IDE+1+CL_AREA_EE'

Concept: REF_ AREA (=reference area)

Code list CL_AREA_EE is used for REF_AREA

SCD+4+BOP_ITEM++++:3'
ATT+3+5+:::AN3'
IDE+1+CL_BOP_ITEM'
SCD+4+DATA_TYPE++++:4'
ATT+3+5+:::AN1'
IDE+1+CL_BOP_DATA_TYPE'
SCD+1+TIME_PERIOD++++:5'
ATT+3+5+:::AN..35'
SCD+1+TIME_FORMAT++++:6'
ATT+3+5+:::AN3'
SCD+3+OBS_VALUE++++:7'
ATT+3+5+:::AN..15'
SCD+3+OBS_STATUS++++:8'
ATT+3+5+:::AN1'
ATT+3+35+2:USS'
ATT+3+32+5:ALV'
IDE+1+CL_OBS_STATUS'
SCD+3+OBS_CONF++++:9'
ATT+3+5+:::AN1'
ATT+3+35+1:USS'
ATT+3+32+5:ALV'
IDE+1+CL_OBS_CONF'
SCD+3+OBS_PRE_BREAK++++:10'
ATT+3+5+:::AN..15'
ATT+3+35+1:USS'
ATT+3+32+5:ALV'
SCD+Z09+TITLE'
ATT+3+5+:::AN..70'
ATT+3+35+2:USS'

No IDE segment for TITLE, as it is uncoded

ATT+3+32+9:ALV'
SCD+Z09+UNIT'
ATT+3+5+:::AN..4'
ATT+3+35+2:USS'
ATT+3+32+4:ALV'
IDE+1+CL_UNIT'

SCD+Z09+UNIT_MULT'
ATT+3+5+:::AN..2'
ATT+3+35+2:USS'
ATT+3+32+4:ALV'
IDE+1+CL_UNIT_MULT'

1644

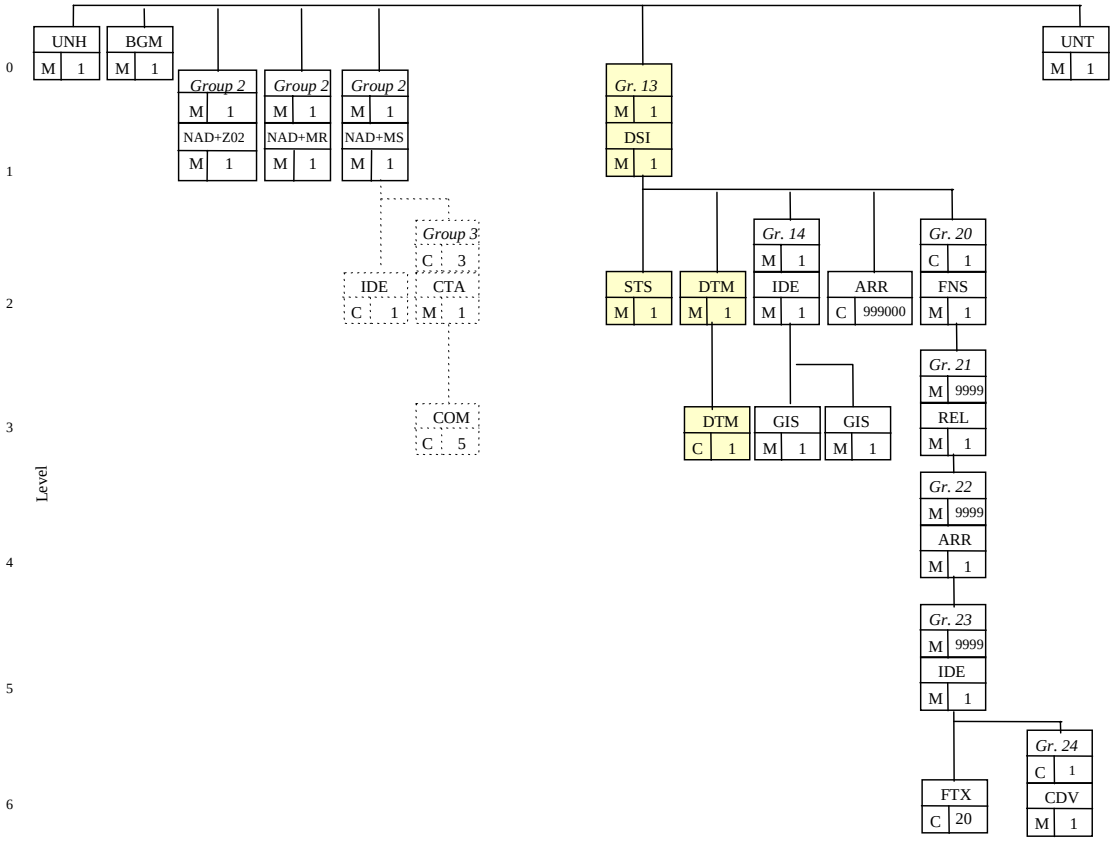
9.7 Data Set Administration Section

9.7.1 Scope

The Data set administration Section includes the identification of the data set, the action which has to be performed (update/replace or delete) and the included data and attribute values extraction date and time.

9.7.2 GESMES Segments Used

In this branching diagram the relevant boxes appear shaded and Group 13 is shown as mandatory.



9.7.3 Guidelines For Use of Segments

The data set identifier is indicated in the DSI segment. It is always present in messages carrying data and/or attributes (or containing instructions for deletions of data an/or attributes). This Group is not used in "structural messages" (containing e.g. key family definitions or code lists). Thus, if Group 13 is used by the sender, then the message should contain also data (or references to delete data) and/or attributes (or references for deleting attributes).

In the context of SDMX-EDI, it has been agreed that:

an update or delete message can contain data or references to **only one key family** and to **one data set**. Therefore, **the DSI segment can appear only once in a message**;

- **if a message contains Group 13, it should not contain any structural data** (Groups 4 to 12 should not be present).

9.7.4 Data Set Administration Section: Segment Structure

[If group 13 is used:]

DSI+*data-set-identifier*' (data set identification: *mandatory*)

STS+*status-type*+*status-code*' (status of the message: *mandatory*)

DTM+*date-time-type*:*date-time*:*date-time-format*'(1st DTM - data set extraction time stamp: *mandatory*)

DTM+*date-time-type*:*date-time*:*date-time-format*'(2nd DTM - reporting period: *conditional*)

Segment Group: **Group 13 (DSI-STS-DTM)**

Max. Number of Occurrences: *1*

Status: *Conditional*

Content and General Usage Rules

In SDMX-EDI only one occurrence of this group is allowed in a message; Group 13 comprises the segments:

- DSI (dataset identification)
- STS (status)
- DTM (first occurrence: date time of data set extraction)
- DTM (second occurrence: reporting period; conditional segment)

When this group is used in a message, this message should also contain:

- Sufficient information (i.e. group 14) for the ARR structure identification

and

- observations and/or attributes

or

in case of a “delete message”,

- references to observations and/or attributes

Segment Type: **DSI**

Segment Type Name: *Data Set Identification*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

DSI+*data-set-identifier*'

Data Element Usage Rules

local name	Usage rules	format	SDMX-EDI code values
Data-set-identifier	Mandatory	an..18	The data set identifier is specified and agreed between data exchange partners in the context of a data exchange agreement.

Content and General Usage Rules

The DSI identifier describes the specific statistical context which the included data and/or attribute values belong to. Guidelines concerning the use of the data set are provided by the centre administrating the data exchange (see also Box 1 on next page).

Implementation Remark

It is recommended to partner institutions to build their database systems and filters with "data set" intelligence, basing their design (when for example data files are received) on the DSI identifier of the received message. The same applies for the extraction systems: it is better to base the development of the "export" routines on the concept of *a data set (in general, a subset of series belonging to the same key family) which needs to be extracted* and not on the extraction of all series which belong to the same key family. One or more data sets (identification of the data set is provided by the DSI segment) can follow the structural definitions for the same key family (information provided by the key family identifier on the IDE segment). See also discussion in Box 1.

Example 1:

DSI+ECB_BOP1'

The reported data belong to the ECB_BOP1 data set.

Example 2:

DSI+BIS_MACRO'

The reported data in the message belong to the BIS_MACRO data set.

BOX 1. DATA SET IDENTIFICATION AND DISTINCTION OF SPECIAL REPORTS AND TARGET DATABASES

In SDMX-EDI the structure of the data is provided by the key family definition. A *data set* consists of series based on the same key family definition. And a data set *may* comprise *all* time series following the same key family definition; however, in general, data exchange arrangements between institutions may also stipulate the "grouping" of sibling groups and time series into different data sets as indicated in the examples below. Note that such different data sets or "groups" may be mutually exclusive or not (allowing in the latter case also the exchange of data and attribute vintages).

For reasons of simplicity and if no special distinction is needed, a common identifier might be used in the DSI segment (data set identifier) and the IDE segment (key family identifier in Group IDE-GIS-GIS), but this is not mandatory. In general, data exchange partners have to agree on the exact definition of each data set.

Here are some examples of exchanging multiple data sets (i.e. many data set identifiers involved) based on the same key family (i.e. using a single key family identifier):

- The receiver might prefer to receive data sets organised into tables or "reports" (a different message for each "report", e.g. the monthly report with monthly Balance of payments data and the quarterly report with quarterly Balance of payments data. This is an example for mutually exclusive data sets on the time series level.
- Two parties may agree to use the same key family for more than one data exchange projects: e.g. a key family definition may be relevant for two or three economic sub-domains and a "domain" distinction might be needed so that the receiver can forward the data sets to different data bases or production units if required.
- There may be a need to exchange vintages of time series and/or attributes compiled using different methodologies, timings or sources.
- A data flow based on a key family is regularly performed. However, based on the same key family definition, an experimental (pilot) data exchange project may be agreed for some time including additional sibling groups. In this case, most probably, the partners would prefer not to mix in the same receiving databases the data flows serving the new pilot data exchange with the ones belonging to the regular production arrangements.
- Quarterly and monthly data sets may be compiled and updated in different timings, leading therefore to non-comparable datasets that may need to be distinguished.

To illustrate the case, let us assume a key family called «MACRO_VARS_KF» which has been defined in such a way that it is used for both national accounts data and public finance data. The receiving institution has two separate production Units, one for national accounts and one for public finance data, and each one of them would like to receive only the message for which it is responsible. We assume that there is a group of reporting series needed to both areas and, thus, there is no way to split them into distinct groups basing the selection addressed to each unit on a "time series key" criterion (the two data sets are not mutually exclusive). The agreement could be to use two different messages as follows:

Message 1

```
…
DSI+MACRO_NAT_ACCS
'
…
IDE+5+MACRO_VARS_K
F'
```

Message 2

```
…
DSI+MACRO_PUB_FIN'
…
IDE+5+MACRO_VARS_KF
'
```

Thus, the receiving system should recognise and distinguish the incoming messages, forwarding them possibly to the database of the corresponding responsible Unit.

Segment Type: **STS**

Segment Type Name: *Status Report*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

STS+*status-type*+*status-code*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
status-type	mandatory this specifies the type of *status-code* used	an..3	3 - data contents
status-code	this specifies the status identified by *status-type*	an..3	7 - update or replace 6 - delete

Content and General Usage Rules

This segment specifies the action expected to be performed on the contents of the message. As in SDMX-EDI only one DSI (one reference to a data set) is allowed per message, **a message can be used to instruct the receiver either to perform deletions or to update a receiving database (these two actions cannot be mixed in the same message)**: the action declared by this status code affects the whole message:

if the segment has been written as **STS+3+7**' then the (update) message should contain either data (in the ARR segment following Group 14) or attributes (in Groups 20-24) or both;

if the segment has been written as **STS+3+6**' then the (delete) message should contain either references to delete data (in the ARR segment following Group 14) or references to delete attributes (in Groups 20 to 23) or both.

Examples:

- STS+3+7'

The contents of the message should be used to update the databases of the receiver and/or to replace previously existing values (assuming that the test indicator of the interchange was not used).

- STS+3+6'

The references in the ARR segments of the message are pointers for deletions which should be performed in the databases of the receiver (of course the receiving end could implement adequate security and control mechanisms, if desirable, in order to check or to prevent fully automated deletions).

Segment Type: **DTM (1st occurrence: data set extraction time stamp)**

Segment Type Name: *Date/Time/Period*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

DTM+*date-time-type:date-time:date-time-format*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
date-time-type	mandatory	an..3	242 - data set preparation date
date-time	mandatory date and local time of the extraction of the data set; its format is defined by *date-time-format*	an..35	format (in SDMX-EDI): CCYYMMDDhhmm
date-time-format	mandatory	an..3	203 (=format: CCYYMMDDhhmm)

Content and General Rules

In general, DTM is used to present dates/time and periods. Its first occurrence in SDMX-EDI is used to present the data set "preparation date/time" (=242). The date format used is always the same (CCYYMMDDhhmm which is coded as 203 in EDIFACT). The sending institutions should provide in this segment the date/time at which the data set reported in the message was extracted from the internal data base system (local date and time of the sender).

If the previously given STS segment indicated that the message will be used for deletions, then this date/time should be the date/time of the preparation of the message.

Example:

DTM+242:199702241345:203'

The reported data set was extracted from the database of the sender at 13:45 (local time) on the 24th February 1997.

Segment Type: **DTM (2nd occurrence: reporting period)**

Segment Type Name: *Date/Time/Period*

Max. Number of Occurrences: *1*

Status: *Conditional*

Data Element Sequence

DTM+*date-time-type*:*date-time*:*date-time-format*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
date-time-type	mandatory	an..3	Z02 - reporting period
date-time	mandatory	an..35	Period or range with format depending on the value of the following element.
date-time-format	mandatory	an..3	for specific periods: 102 - for CCYYMMDD 602 - for CCYY 604 - for CCYYS 608 - for CCYYQ 610 - for CCYYMM 616 - for CCYYWW for period ranges: 711 - for CCYYMMDD-CCYYMMDD 702 - for CCYY-CCYY 704 - for CCYYS-CCYYS 708 - for CCYYQ-CCYYQ 710 - for CCYYMM-CCYYMM 716 - for CCYYWW-CCYYWW where: CC - century YY - year S - half-year (1,2) Q - quarter (1,2,3,4) MM - month in range 01-12 WW - week within a calendar year in the range 1-53 DD - day in month 01-31

Content and General Rules

The second occurrence of DTM is conditional and is used to declare which is the reporting period (if relevant and/or requested by a centre institution). Obviously, if this segment is used, the period or period range given in this segment should be consistent with the contents (data) written in the message.

Example 1:

The following segment says that the message contains data for 1992 (single period):

DTM+Z02:1992:602'

Example 2:

The following segment says that the message contains data from 1992 to 1993 (period range):

DTM+Z02:19921993:702'

Example 3:

The following segment says that the message contains data from 1995q1 to 1995q4 (period range):

DTM+Z02:1995119954:708'

.

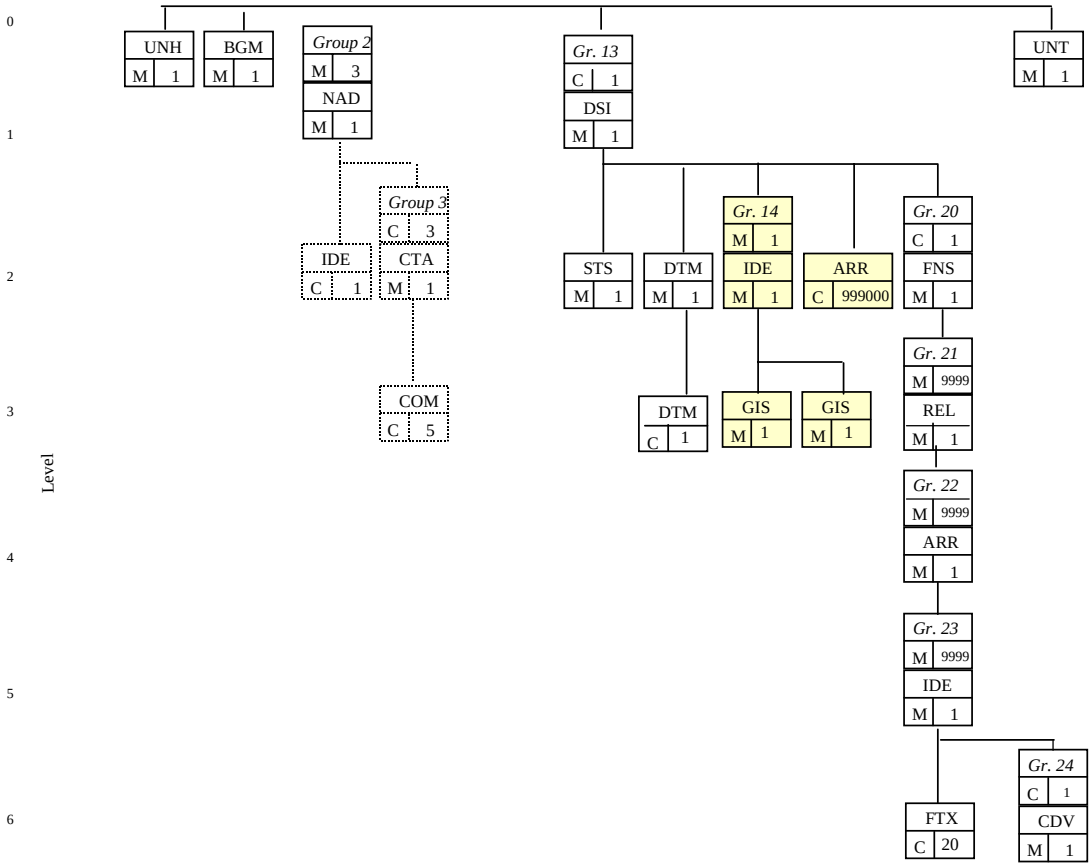
9.8 Array Structure and Data Section

9.8.1 Scope

The array data comprises the identification of the data set structure (IDE), the indication of the method used to place data values in the ARR segment (first occurrence of GIS), an indication about the character used for the missing values (second occurrence of GIS) and the data values (or references for deletions) in the ARR segment.

9.8.2 GESMES Segment Usage

In this branching diagram the shaded boxes (Group 14 and ARR) are the ones related to the *array structure* and the contained *data* (or references). Group 14 and the following ARR segment are shown here as mandatory to indicate that the *Array structure and data Section* is present.



9.8.3 Features and Guidelines for Use

The IDE segment provides the data set structure defining the expected sequence of data values in the ARR segment. In SDMX-EDI it is indicated by the key family identifier. The GIS segment (first occurrence) indicates which one of the four alternative ways of placing data in the ARR segment has been used (in SDMX-EDI it is always the same: AR3). The second occurrence of GIS is used to indicate the missing value character used. In SDMX-EDI the ARR segment contains the key, the numeric values and their corresponding status flags.

9.8.4 Array Structure and Data Section: Segment Structure

array structure

IDE+*object-type+identity-number*'

GIS+*processing-indicator:list-qualifier:code-maintenance-agency:processing-value*'

(GIS has to be present twice)

array data

ARR++*array-cell-data:array-cell-data:array-cell-data:array-cell-data:array-cell-data:array-cell-data:etc.*'

(the sequence just above is the generic expression; for further information, see p. 105 and the pages which follow)

Segment Group: **Group 14 (IDE-GIS-GIS)**

Max. Number of Occurrences: *1*

Status: *Mandatory*

Content and General Usage Rules

Group 14 comprises the segments:

- IDE - array structure identification
- GIS - message processing indicator
- GIS - missing value indicator

This group has to be present in all messages containing or referring to data and/or attributes (including the case of a pure attribute update or delete message; the ARR segment which follows Group 14 would not be present in this case).

Segment Type: **IDE**

Segment Type Name: *Identity*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

IDE+*object-type+identity-number*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
object-type	Mandatory	an..3	5 - data set structure
identity-number	Mandatory the identity of the data set structure	an..35	the key family identifier

Content and General Rules

This segment is used to indicate the structure of the data set (e.g. how many dimensions, which ones and in which order). This information is provided by the key family identifier.

.

Example 1:

IDE+5+ECB_BOP1'

Example 2:

IDE+5+BIS_MACRO'

Segment Type: **GIS (first occurrence)**

Segment Type Name: *General Indicator*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

GIS+*processing-indicator*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
processing indicator	mandatory	an..3	AR3 - ARR usage method 3

Content and General Usage Rules

This segment (first occurence) is used to define how the data elements are used in the ARR segment. In SDMX-EDI it is always AR3.

Example:

GIS+AR3'

In SDMX-EDI the segment has to be written in this fixed format.

Segment Type: **GIS (second occurence)**

Segment Type Name: *General Indicator*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

GIS+*processing-indicator*::: *processing-value*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
processing indicator	Mandatory	an..3	1 - no data available
processing-value	mandatory	an..17	- dash; the symbol for missing values in SDMX-EDI

Content and General Usage Rules

This segment (second occurence) is used in SDMX-EDI to define which is the character used for the missing values in the ARR segment. The reading applications should read this character, parametrise it and, based on this, they should recognise the missing values in the ARR segment.

Example:

GIS+1:::-'

In SDMX-EDI this segment is written in this fixed format.

Segment Type: **ARR**

Segment Type Name: *Array Information*

Max. Number of Occurrences: *999000*

Status: *Conditional*

Data Element Sequence in SDMX-EDI

"Single observation" technique:

ARR++series-key:period:time-format:observation:observation-status:obs.-confidentiality:obs-pre-break'

ARR++*series-key:period:time-format:observation:observation-status:obs.-confidentiality: obs-pre-break*'

(repeats)

"Time range" technique:

ARR++*series-key:period-range:time-format:observation:observation-status:obs.-confidentiality:obs-pre-break+*

observation:observation-status:obs.-confidentiality:obs-pre-break +
etc.
observation:observation-status:obs.-confidentiality:obs-pre-break'

ARR++*series-key:period-range:time-format:observation:observation-status:obs.-confidentiality:obs-pre-break+*

observation:observation-status:obs.-confidentiality:obs-pre-break+
etc.
observation:observation-status:obs.-confidentiality:obs-pre-break'

(repeats)

Data Element Usage Rules

Local name	usage rules	Format	SDMX-EDI code values
Series-key	mandatory the key of the reported time series; the rules are detailed below	component elements separated by component separators	dimension values of the time series key delimited by component element separators
Period (or period-range)	mandatory the period (or period range) to which the reported observation(s) in this ARR segment correspond; the rules are detailed below	an..35	period or range of periods written in one of the foreseen formats. See Table 8. period and period range format codes
Time-format	mandatory the rules are detailed below	an3	the code list for these values is presented in the table below
Observation	mandatory for updating [not used in "delete messages"] the rules are detailed below	an..15	a numeric observation or a missing value indicator
Observation-status	mandatory for updating [not used in "delete messages"] the rules are detailed below	an..35	A value taken from an observation status code list (e.g. CL_OBS_STATUS with format an1)
Observation-confidentiality	Conditional [not used in "delete messages"] the rules are detailed below	an..35	A value from an observation confidentiality code list (e.g. CL_OBS_CONF with format an1)
Pre-break-observation conditional	Conditional [not used in "delete messages"]	an..15	a numeric observation or a missing value indicator

The following rules and explanations refer only to reporting data (updating actions). The role and the use of the ARR segment in deletions are discussed later as a separate chapter.

Content and General Usage Rules When Reporting Data *(for deletions see p.136)*

- **Time series key**

A time series key is composed by values given to the (ordered) dimensions included in the key structure of a key family. The dimensions are reported separated by the component separator ":"(colon), e.g.:

Q:BE:100:4.

In each ARR segment only one time series key can be present (just after the ARR++ characters), regardless of the technique used (single observation or time range). Within one message all time series keys used in the ARR segment have to belong to the same key family.

- **Periods, period ranges, time format and time format code list**

Period and time format have always to be given as a "pair" separated by the component separator ":" (colon).

Example: *single monthly period:* 199505:610 (= May 1995)

period range (quarterly): 1994119962:708 (= 1994 Q1 to 1996 Q2)

The table below indicates the formats for period specifications and the time format codes.

Table 8. period and period range format codes

Types for dates and specific periods			
Code	format	explanation	dates and specific periods: examples
203	CCYYMMDDhhmm	(for series with freq. higher than daily) year/month/day/hours/minutes	199511210850 = 21 Nov. 1995 (08:50) date & format code: 199511210850:203
102	CCYYMMDD	(for daily and business series) year/month/day	19951121= 21 Nov. 1995 date & format code: 19951121:102
616	CCYYWW	(for weekly series) year/ week	199252= 52nd week of 1992 date & format code: 199252:616
610	CCYYMM	(for monthly series) year/month	199511= Nov. 1995 date & format code: 199511:610
608	CCYYQ	(for quarterly series) year/quarter	19953=1995q3 date & format code: 19953:608
604	CCYYS	(for half yearly series) year/semester	date & format code: 19951:604
602	CCYY	(for annual series) year	date & format code: 1995:602
Types for period ranges			
Code	format	explanation	period ranges: examples
711	CCYYMMDD CCYYMMDD	(for daily and business series) from year/month/day to year/month/day	1992110419930124=4Nov92-24Jan1993 date & format code: 1992110419930124:711
716	CCYYWW CCYYWW	(for weekly series) from year/week to year/week	199227199550= from week no.27 of 1992 to week no.50 of 1995 date & format code: 199227199550:716
710	CCYYMM CCYYMM	(for monthly series) from year/month to year/month	199208199511= from Aug92 to Nov95 date & format code: 199208199511:710
708	CCYYQCCYYQ	(for quarterly series) from year/quarter to year/quarter	1992319954= from 1992q3 to 1995q4 date & format code: 1992319954:708
704	CCYYSCCYYS	(for half yearly series) from half-year to half-year	1995219961= from 1995h2 to 1996h1 date & format code: 1995219961:704
702	CCYYCCYY	(for annual series) from year to year	19951996= from 1995 to 1996 date & format code: 19951996:702

Data elements: observation and observation level attributes

Following the time series key and the time period and format indication the actual observations are reported as data elements. Each data element consists of at least 2 components: the **observation** itself and the mandatory (observation level) attribute "**Observation status**". Depending on the key family a third and a fourth component element may be added: the

(observation level) attribute "**Observation confidentiality**" and "**Pre-break observation**". Within the data element the components are separated by the component separator ":" (colon).

Example: Data element with 2 components:

1234.5:A (only observation status)

Data element with 3 components:

1234.5:A:C (observation status + confidentiality)

Data element with 4 components:

1234.5:B::1230.5 (observation status, confidentiality not given, pre-break observation)

Data element with 4 components:

1234.5:B:C:1230.5 (observation status, confidentiality and pre-break observation)

The *time format indication* and the *first data element with an observation* have to be separated with a component separator ":". The *individual data elements* that follow are then separated by the data element separator "+" (plus sign), which follows each data element. Due to the GESMES truncation rules the data element separator, following the last data element in the segment, is omitted and replaced by the segment terminator sign (', single quote). The maximum number of data elements in an ARR segment is 9999, i.e. the overall maximum number of data element separators ("+") is 10,000 including the 2 immediately following the ARR statement. A time series with more than 9999 observations has to be split over more than 1 ARR segment. It is recommended that applications use the maximum number of data elements as a parameter.

Example: ...+1234.5:A+234.6:B+3456.7:A' (3 data elements followed by segment terminator)

Remarks, rules, special cases and recommendations concerning the handling of observation elements

In SDMX-EDI this **maximum length** of the component element holding the observation is 15 positions. As a consequence the maximum number of significant figures for an observation is:

⇒ 15 for a positive integer
⇒ 14 for a positive decimal or a negative integer
⇒ 13 for a negative decimal

- Exclusively the point (".") is used as **decimal separator**.

Scientific notation may be used for both the observation and the observation pre-break, if agreed by the parties involved. To express a number in scientific notation[1]: omit any plus signs, as "+" is a reserved character in GESMES and may be misinterpreted; use "E" before the exponent; the value of the number before the "E" should be between -10 and 10; to express numbers between minus one and one preface the exponent with a negative sign.

Examples: The number 1,230,000 is written 1.23E6. The number 0.000001 is written as 1.0E-6.

[1] For more information on the rules applying see http://www.ex.ac.uk/cimt/dictunit/enote.htm (section on "E-format").

Again, in SDMX-EDI, the total length of the field should not exceed 15 characters.

Pre-break observations do not have an observation status.

A **missing value** is indicated using a dash ("-") (e.g. ...+-:H+...). The explanation of the absence of a value is provided by the observation status which follows. (Applications should be parameterised for this "missing observation" character, taking its value from the second occurrence of the GIS segment.)

If there are **unchanged or non-reported values** within a reported time range, the corresponding separators have still to be written; therefore a sequence "++" means that the observation which would be expected between the two plus signs is considered either as not reported or as unchanged vis-à-vis the last reported value (see also examples below when discussing the time range technique).

Observations have always to be reported together with an observation status value. If an observation changes and the status is not changed, both components have to be reported. Likewise, if the observation status changes and the observation remains unchanged, both components would have to be re-reported; the same would apply also if the observation confidentiality status changed (relevant for the key families in which this attribute might be used): all three elements have to be re-reported. **Similarly, if the observation confidentiality status or the pre-break observation changed, all component elements would have to be re-reported.** By not reporting, for a certain observation, neither an observation confidentiality attribute nor an observation pre-break value, the receiver should assume that for this period values (for the confidentiality status and pre-break) never existed or they cease to exist from now on.

Example 1:

On 10 March 1999 the segment ARR++M:YY:ZZ:199902:610:-7.9:E:C' was reported.

On 15 March 1999 the segment ARR++M:YY:ZZ:199902:610:-7.9:A:F' was reported.

The second time the sender resent the observation, because the status (from Estimate to normal) and the observation confidentiality (from "Confidential" to "Free") were modified.

Example 2:

On 10 March 1999 the segment ARR++M:YY:ZZ:199902:610:-7.9:E:C' was reported.

On 15 March 1999 the segment ARR++M:YY:ZZ:199902:610:-7.9:A' was reported.

The second time the sender sent the segment without the observation confidentiality value; in this case the receiver would need to decide how to interpret the fact that the confidentiality attribute was not reported: he could either delete the previously existing value (C) or leave it unchanged (i.e. continue to mark the observation as confidential).

Example 3:

On 10 April 1999 the segment ARR++M:FG:T1:199902:610:10:B:F:12' was reported.

On 15 April 1999 the segment ARR++M:FG:T1:199902:610:10:A:F' was reported.

Here it is assumed that in the first transmission (10-Apr-1999) the sending institution reported a break in series together with a pre-break value. In the second transmission it is now assumed

that the sending institution revised all back data, built a series without breaks and re-reported the same observation without the "break" flag and without the pre-break observation.

Time range and single observation technique

With the ***time range technique*** the period and time format specification in the ARR segment are set to define a range of periods, which is then followed by a number of data elements, which has to exactly match the number periods in the defined time range.

Also, as it had been discussed earlier (see paragraph describing the data elements of the ARR segment):

- no more than 9,999 observations can be reported in the same ARR segment;
- **the first and the last observation of the range should be explicitly written in the segment**: in this way, the number of observations reported in an ARR segment should always be equal to the number of periods indicated by the time range (otherwise, the EDIFACT truncation rules would allow deviation from this principle).

Example 1. - time range technique

ARR++M:YY:ZZ:199301199304:710:39.9:A+21.5:A+23.4:A+43.0:E'

This segment contains four observations (with their corresponding status) for the series M:YY:ZZ. The observation 39.9 (status: A) is for Jan.93, 21.5 (A) is for Feb.93, 23 (A) for Mar. 93 and the observation 43.0 (E) is for Apr.93.

⇒ With the ***single observation technique*** the period and time format specification are set to define a single time period, which is then followed by a single component data element for the observation and its attribute(s).

Example 2. - single period technique

ARR++M:YY:ZZ:199301:610:39.9:A'

ARR++M:YY:ZZ:199302:610:21.5:A'

ARR++M:YY:ZZ:199303:610:23.4:A'

ARR++M:YY:ZZ:199304:610:43.0:E'

These four segments contain one observation each and they are equivalent to the single segment shown in the previous example.

The two techniques can be mixed in the same message (but, of course, not within the same ARR segment):

Example 3:

the following set of segments is valid (and equivalent to the ones shown in examples 1 & 2):

ARR++M:YY:ZZ:199301:610:39.9:A'

ARR++M:YY:ZZ:199302199304:710:21.5:A+23.4:A+43.0:E'

and the following one (example 4) is also valid and an equivalent form of the previous set of segments:

Example 4:

ARR++M:YY:ZZ:199301199302:710:39.9:A+21.5:A'
ARR++M:YY:ZZ:199303:610:23.4:A'

ARR++M:YY:ZZ:199304:610:43.0:E'

The examples below (5, 6 and 7) show the use of the two techniques when "reporting" **unchanged or non-reported observations** and their flags (it was discussed theoretically on a previous page); of course, it makes sense to refer to reporting of *unchanged* observations only when using the ***time range technique***.

Example 5:

It is assumed that an institution has to report for the series M:YY:ZZ only data for Nov.92, Jan.93 and Apr.93 since only these data were revised or updated (the observation for April 1993 is an estimate and it should be treated as confidential); for Dec92, Feb93 and Mar93 the data which had been reported in the past remain still valid:

Series: M:YY:ZZ

Nov92	Dec93	Jan93	Feb93	Mar93	Apr93
-7.9 (A)	unchanged	37.8(A)	unchanged	unchanged	43 (E, C)

When using the time range technique all separators (including the ones referring to the unchanged observations) should be explicitly written; then, the sequence above is reported as:

ARR++M:YY:ZZ:199211199304:710:-7.9:A++37.8:A+++43:E:C'

Example 6:

The following three segments, written using the single observation technique, are equivalent to the one presented just above (example 4):

ARR++M:YY:ZZ:199211:610:-7.9:A'
ARR++M:YY:ZZ:199301:610:37.8:A'
ARR++M:YY:ZZ:199304:610:43:E:C'

Example 7:

In the following segments, another way is shown to report the same data by mixing the two techniques:

ARR++M:YY:ZZ:199211199301:710:-7.9:A++37.8:A'
ARR++M:YY:ZZ:199304:610:43:E:C'

Practical application of the two techniques:

In general, because of the elimination of redundant information, the *time range technique* is the most preferred method for exchanging complete time series or databases: e.g. in the first reporting of a new series to a centre institution or when a central institution disseminates a full data base to other institutions.

The *single observation technique* is simple and efficient for reporting a single observation per series, for example in the updates and revisions context. It can be useful also in the case of series with sparse irregular observations; e.g. for some series with only some observations per year, it is sensible to report only these observations (e.g. using daily frequency with the single observation technique). Usually, this is a matter which will be administrated by the centre (e.g. if the observations are not too sparse, it might be preferable to keep a continuous track of the exchanged series using missing values in conjunction with the "cannot-exist" obs. status).

In general, both the *time range* and the *single observation* technique are equally valid.

The use of the Pre-break value attribute: examples

The following examples correspond to Example 1 and 2 previously shown. The difference is that here they also include a pre-break value.

Example 1 - time range technique:

ARR++M:YY:ZZ:199301199304:710:39.9:A+21.5:B**::20.1**+23.4:A+43.0:E'

The observation for Feb.93, which is a break, includes also ***a pre-break-observation 20.1***.

Example 2 - single period technique:

ARR++M:YY:ZZ:199301:610:39.9:A'
ARR++M:YY:ZZ:199302:610:21.5:B**::20.1**'
ARR++M:YY:ZZ:199303:610:23.4:A'
ARR++M:YY:ZZ:199304:610:43.0:E'

The same as before, using the single period technique.

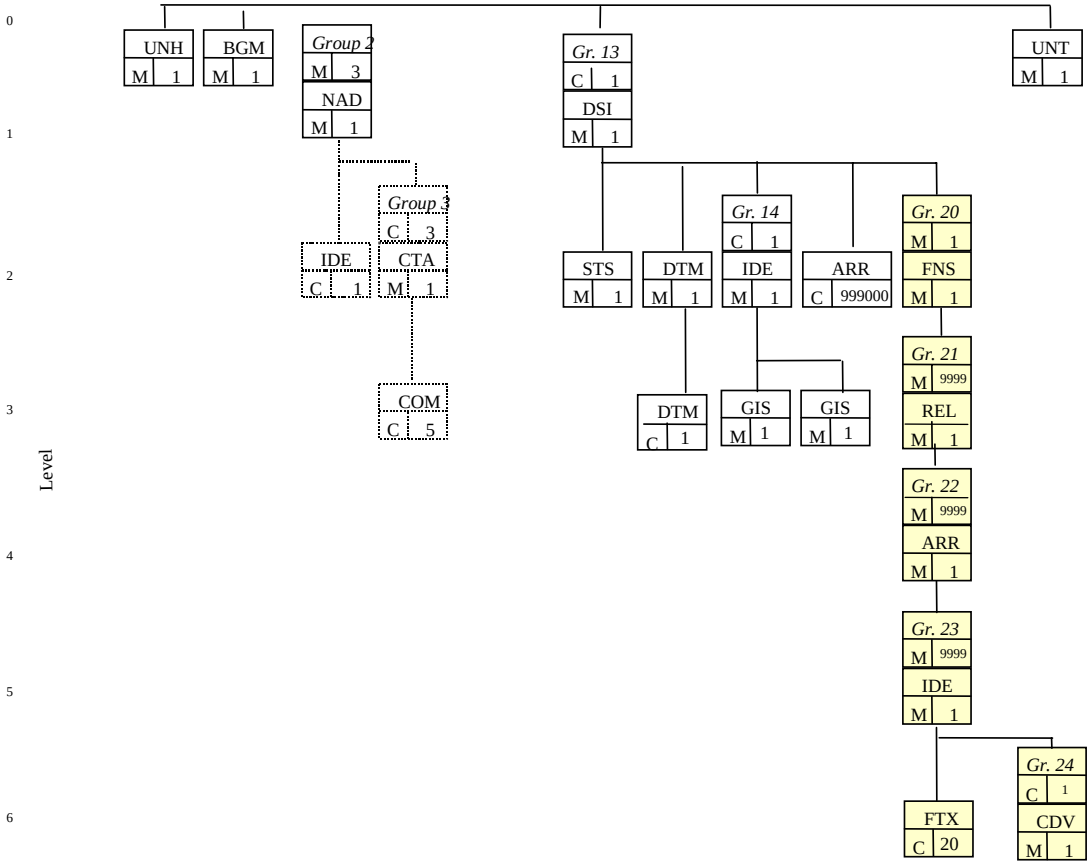
9.9 Attribute Section

9.9.1 Scope

An attribute is an object relating to (i) to an observation or (ii) to a time series or (iii) to a group of sibling series or to (iv) a data set. It provides information about the corresponding data (at that level), such as the measurement unit or the confidentiality status. An attribute can be coded (taking values from a code list) or uncoded (free text). The purpose of the segments included in the Attributes Section is to allow the exchange of attributes at the levels foreseen by the data model.

9.9.2 GESMES Segment Usage

The Attributes Section comprises all groups from Group 20 to 24 (they are shown shaded in the diagram below). These groups have to be used in order to exchange attributes (that it is why Group 20 is shown here as mandatory). Regardless of whether the message carries observations as well (in which case the ARR segment which follows Group 14 would be also used), Groups 13 and 14 need also to be used as they provide information about the relevant data set and key family.



Guidelines for Usage

Group 20 is used always in the Data and Attribute message (or in an "attribute only" update message). It is used also when specific attributes need to be deleted.

The attributes are all included in a set introduced by the FNS segment.

The REL segment is used to define the scope of the attribute in terms of the object type to which the attributes relate.

The ARR segment defines the statistical objects to which the attributes relate.

The IDE segment identifies the statistical concept.

If the statistical concept is coded, then the CDV segment provides its value;

If it is uncoded, then the FTX segment contains the textual value.

9.9.3 Attribute Section: Segment Structure

FNS+*set-identifier:identity-number-type*'
REL+*relationship-type+array-scope*'
ARR+*last-dimension-position+array-cell-data:array-cell-data:array-cell-data:etc.*'
IDE+*object-type+identifier*'
CDV+*code-value*' *or*
FTX+*text-subject+++text*'

Segment Group: **Group 20 (FNS-Group 21)**

Max. Number of Occurrences: *1*

Status: *Conditional*

Content and General Usage Rules

This group contains a set of attributes relevant to the specified data set and following the specified key family definition in use throughout this message.

The **examples** used for the segments relating to the FNS group on attributes are built up in such a way that with each added segment the reader retains the overall view of the whole group. The completed examples are then presented following the description of the CDV segment (Group 24).

Segment Type: **FNS**

Segment Type Name: *Footnote Set*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

FNS+*set-identifier:identity-number-type*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
set-identifier	mandatory the identity of the set of attributes	 an..35	(any text) e.g.: Attributes
identity-number-type	mandatory the object is “attribute” (=10)	 an..3	 10 - Attributes

Content and General Usage Rules

This segment is used to identify the attribute set. In principle, the reading applications will use the FNS segment to confirm that an attribute section starts, but they can ignore the *set-identifier* given in the segment (it is enough to acknowledge the existence of the FNS segment).

Example:

FNS+Attributes:10'

In the examples which follow in the next few subsections, the FNS segment is not shown again, as it has a constant format.

Segment Group: **Group 21 (REL-Group 22)**

Max. Number of Occurrences: *9999*

Status: *Conditional*

Content and General Usage Rules

This group contains a specification of the scope of the attribute(s) (REL), followed by the attributes (group 22).

Segment Type: **REL**

Segment Type Name: *Relationship*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

REL+*relationship-type+array-scope*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
relationship-type	mandatory	an..3	Z01 - relationship is with a statistical array
array-scope	mandatory this provides information about the scope of the attributes which follow; the attributes can refer to: - the data set identified in the DSI segment (=1); - series or sibling series level (=4); - observation level (=5)	an..3	1 - data set. 4 - specific combination of dimension values 5 - observation

Content and General Usage Rules

This is used to provide the scope of the footnote(s) or attribute(s).

Dependencies

The *array-scope* given here should be consistent with the contents of the Group 22-ARR segment which follow (until a new REL is found). If *array-scope=4*, then in the ARR segments (which follow the REL segment) the references should be to specific dimension values identifying either a time series or a sibling group (in the latter case the frequency would be wildcarded). If *array-scope=5*, then the references for the attributes will indicate attachments at the observation level (all dimensions plus time reference plus time-format). If *array-scope=1*, then the ARR segments which follow should not contain dimension values, but all dimensions should be wild-carded (attributes attached at the data set level).

Example 1 - attribute for a data set:

DSI+ECB_TEST'	DSI is identical to IDE and identifies the data set ECB_TEST
. . .	
IDE+5+ECB_TEST_KF'	Key family: ECB_TEST_KF
. . .	
REL+Z01+1'	**Relationship is with the data set**

Example 2 - attributes for sibling series:

REL+Z01+4'	**Array scope is "combination of dimension values"**
...	ARR segment to follow

Example 3 - attributes for a time series:

REL+Z01+4'	**Array scope is "combination of dimension values"**
...	ARR segments to follow

Example 4 - attribute for an individual observation:

REL+Z01+5'	**Array scope is "observation"**
...	ARR segments to follow

Segment Group: **Group 22 (ARR-Group 23)**

Max. Number of Occurrences: *9999*

Status: *Conditional*

Content and General Usage Rules

This group contains an ARR segment which defines the statistical objects to which the attributes which follow refer.

Then, the name (identifier) of the attribute follows (in an IDE segment) and its value follows which can be:
either text (FTX segment)
or a value from a code list (group 24, CDV)

The examples that are presented in this subsection include also relevant segments (e.g. REL) which should precede the appearance of the segments of the group.

Segment Type: **ARR**

Segment Type Name: *Array Information*

Max. Number of Occurrences: *1*

Status: *Mandatory (trigger)*

Data Element Sequence

ARR+*last-dimension-position+array-cell-data:array-cell-data:array-cell-data:etc.*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
Last-dimension-position	Mandatory this element is used to indicate the number of dimensions which are relevant for the attachment of the attribute: (a) if the attribute qualifies a sibling group or a time series, then the *position* indicates the number of dimensions in the key family. (b) if the attribute qualifies an observation, then the *position* should indicate the number of dimensions in the key family increased by two (additional dimensions: period, time format). (c) if the attribute qualifies the whole data set (given by the data-set-identifier in the DSI segment) , then the *position* should be set equal to zero.	an..12	This value can be: (a) the number of dimensions in the key family (for attributes at sibling group or time series level) (b) the number of dimensions in the key family increased by two (for the OBS_COM attribute) (c) 0 - for attributes attached at the data set level (as identified by the DSI segment).
Array-cell-data	Conditional this contains dimension values which specify uniquely the position in the array to which the attribute relates: - wildcarding of one dimension is valid for the sibling group level - if *array-cell-data* refers to the time dimension value (the case of an observation), then both the period and periodicity must be specified in the sequence period:time format	an..35	see examples; (not used for attributes attached at the data set level)

Content and General Usage Rules

The segment is used to specify the values in the data set for which the attribute(s) are relevant.

For attributes at the observation level, the *last-dimension-position* is the number of dimensions plus two: the reason is that the conceptual dimension time (which is needed in order to address the observation level) is represented in GESMES by two physical dimensions: period and time format (see also examples).

For attributes at the data set level, the *last-dimension-position* must be set equal to 0: this is the method to wildcard all dimension values.

This ARR segment plays an important role also in deletions of attributes (see p. 136).

Example 1 - attribute for a data set:

DSI+ECB_TEST'	It identifies the data set ECB_TEST
. . .	
IDE+5+ECB_TEST_KF'	Key family: ECB_TEST_KF
. . .	
REL+Z01+1'	Relationship is with the data set
ARR+0'	**Attached at the data set level**
...	IDE segments to follow

Example 2 - attributes for sibling series:

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+:XX:ZZ:CC'	**attached to sibling group :XX:ZZ:CC**
...	IDE segments to follow

Example 3 - attributes for a time series:

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+M:XX:ZZ:CC'	**attached to time series M:XX:ZZ:CC**
...	IDE segments to follow

Example 4 - attribute for an individual observation:

REL+Z01+5'	Array scope is "observation"
ARR+6+M:XX:ZZ:CC:199606:610'	**attached to obs. of June 1996 for series M:XX:ZZ:CC**
...	IDE segments to follow

These attribute segments refer to the observation for June 1996 of the series M:XX:ZZ:CC.

It is obvious that in this case we have to write ARR+6, because apart from the key family dimensions (=4) the reading applications should expect *two more dimension references* which are needed to refer to a specific period (the “199606” -date component- and the “610” -format component-). **This is an important point to be taken into account when designing both the *writing* and *reading* applications**.

Segment Group: **Group 23 (IDE-FTX-Group 24)**

Max. Number of Occurrences: *9999*

Status: *Conditional*

Content and General Usage Rules

Each occurence of this group is used to identify an attribute which relates to the object specified in the preceding ARR segment. The attribute present in the IDE segment can be assigned as value:

either a text (FTX segment) for an uncoded attribute

or a value from a code list (group 24, CDV) for a coded attribute

Segment Type: **IDE**

Segment Type Name: *Identity*

Max. Number of Occurrences: *1*

Status: *Mandatory (Trigger)*

Data Element Sequence

IDE+*object-type+identifier*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
object-type	mandatory It indicates the type of the object	an..3	Z10 - coded attribute Z11 - uncoded attribute
identifier	mandatory the identity of the object	an..35	attribute identifier

Content and General Rules

Each occurrence of this segment is used to identify an attribute related to the object specified in the preceding ARR segment.

The *identifier* in this segment points

either to the *text* in the FTX segment

or to the *code-value* in Group 24 (CDV)

which follow.

Example 1 - attribute for a data set:

DSI+ECB_TEST'	It identifies the data set ECB_TEST
. . .	
IDE+5+ECB_TEST_KF'	Key family: ECB_TEST_KF
. . .	
REL+Z01+1'	Relationship is with the data set
ARR+0'	attached at the data set level
IDE+Z10+UNIT'	**Coded attribute UNIT**
...	CDV segment to follow

2362 **Example 2 - attributes for sibling series:**

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+:XX:ZZ:CC'	attached to sibling :XX:ZZ:CC
IDE+Z10+AVAILABILITY'	**Coded attribute AVAILABILITY**
...	CDV segment to follow
IDE+Z11+TITLE'	**Uncoded attribute TITLE**
...	FTX segment to follow

2363

2364 **Example 3 - attributes for a time series:**

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+M:XX:ZZ:CC'	attached to time series M:XX:ZZ:CC
IDE+Z10+COLLECTION'	**Coded attribute COLLECTION**
...	CDV segment to follow
IDE+Z11+COLL_DETAIL'	**Uncoded attribute COLL_DETAIL**
...	FTX segment to follow

2365

2366 **Example 4 - attribute for an individual observation:**

REL+Z01+5'	Array scope is "observation"
ARR+6+M:XX:ZZ:CC:199606:610'	attached to obs. of June 1996 for series M:XX:ZZ:CC
IDE+Z11+OBS_COM'	**uncoded attribute OBS_COM**
...	FTX segment to follow

2367

2368

Segment Type: **FTX**

Segment Type Name: *Free Text*

Max. Number of Occurrences: *20*

Status: *Conditional*

Data Element Sequence

FTX+*text-subject*+++*text*:*text*:*text*:*text*:*text*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
text-subject	mandatory	an..3	ACM - statistical description
text	mandatory	an..70	text value (of an uncoded attribute); it could continue with up to four more component element (up to five in total).
text (2nd comp.element)	conditional	an..70	(text could continue)
text (3rd comp.element)	conditional	an..70	(text could continue)
text (4th comp.element)	conditional	an..70	(text could continue)
text (5th comp.element)	conditional	an..70	(text could continue only using a new FTX segment)

Content and General Usage Rules

Each occurrence of this segment is used to provide the text value of an uncoded attribute. The overall allowed length of the text depends on the specification of the respective attribute in the key family definition. If the foreseen length is longer than 70 characters (upper limit for original text inside a single component element, not counting release characters), then the *text* should be split in (the maximum) 5 component element-texts (each one of them not being longer than 70 characters and separated among them with a *component separator*). The reading applications should read the components of the received text and consider that they form a string which is the result of the added sub-strings. The writing applications should break the original text into pieces of 70 characters or shorter (it does not matter whether a word is broken between two component elements or not), keeping spaces as significant characters and adding release characters as required afterwards.

Multiple FTX segments can be sent (up to 20, one below the other) and the reading applications should consider them as an ordered sequence of sections of the same text. As before, the use of a new FTX does not imply that this should be interpreted as *line feed*.

The free text in FTX should respect the EDIFACT rules for the treatment of the characters used in the syntax (see also separate chapter on using text in uncoded attributes).

Example 1 - coded attribute for a data set:

DSI+ECB_TEST'	It identifies the data set ECB_TEST
. . .	
IDE+5+ECB_TEST_KF'	Key family: ECB_TEST_KF
. . .	
REL+Z01+1'	Relationship is with the data set
ARR+0'	attached at the data set level
IDE+Z10+UNIT'	Coded attribute UNIT
...	CDV segment to follow

Example 2 - attributes for sibling series:

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+:XX:ZZ:CC'	attached to sibling :XX:ZZ:CC
IDE+Z10+AVAILABILITY'	Coded attribute AVAILABILITY
...	CDV segment to follow
IDE+Z11+TITLE'	Uncoded attribute TITLE
FTX+ACM+++MONETARY AGGREGATE M1'	**Attribute text value**

Example 3 - attributes for a time series:

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+M:XX:ZZ:CC'	attached to time series M:XX:ZZ:CC
IDE+Z10+COLLECTION'	Coded attribute COLLECTION
...	CDV segment to follow
IDE+Z11+COLL_DETAIL'	Uncoded attribute COLL_DETAIL
FTX+ACM+++COLLECTED ON 2 LAST WORKING DAYS OF THE MONTH'	**Attribute text value**

Example 4 - attribute for an individual observation:

REL+Z01+5'	Array scope is "observation"
ARR+6+M:XX:ZZ:CC:199606:610'	attached to obs. of June 1996 for series M:XX:ZZ:CC
IDE+Z11+OBS_COM'	Uncoded attribute OBS_COM
FTX+ACM+++NEW ACCOUNTING METHOD'	**Attribute text value**

Segment Group: **Group 24 (CDV)**

Max. Number of Occurrences: *1*

Status: *Conditional*

Content and General Usage Rules

This group provides the value of a coded attribute.

Segment Type: **CDV**

Segment Type Name: *Code value*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

CDV+*code-value*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
code-value	mandatory a code value	an..18	attribute code value

Content and General Rules

Each occurrence of this segment is used to assign a value to the attribute specified in the preceding IDE segment.

Example 1 - attribute for a data set:

DSI+ECB_TEST'	It identifies the data set ECB_TEST
. . .	
IDE+5+ECB_TEST_KF'	Key family: ECB_TEST_KF
. . .	
REL+Z01+1'	Relationship is with the data set
ARR+0'	Attached at the data set level
IDE+Z10+UNIT'	Coded attribute UNIT
CDV+USD'	**code value for US dollar**

Example 2 - attributes for sibling series:

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+:XX:ZZ:CC'	attached to sibling :XX: ZZ:CC
IDE+Z10+AVAILABILITY'	Coded attribute AVAILABILITY
CDV+A'	**code value: A** (=assumed to mean availability="free")
IDE+Z11+TITLE'	Uncoded attribute TITLE
FTX+ACM**+++**MONETARY AGGREGATE M1'	attribute text value

2432 **Example 3 - attributes for a time series:**

REL+Z01+4'	Array scope is "combination of dimension values"
ARR+4+M:XX:ZZ:CC'	attached to time series M:XX:ZZ:CC
IDE+Z10+COLLECTION'	Coded attribute COLLECTION
CDV+A'	**coded value for collection is A = "average of period"**
IDE+Z11+COLL_DETAIL'	Uncoded attribute COLL_DETAIL
FTX+ACM**+++**simple arithmetic mean of daily values'	attribute text value

2433

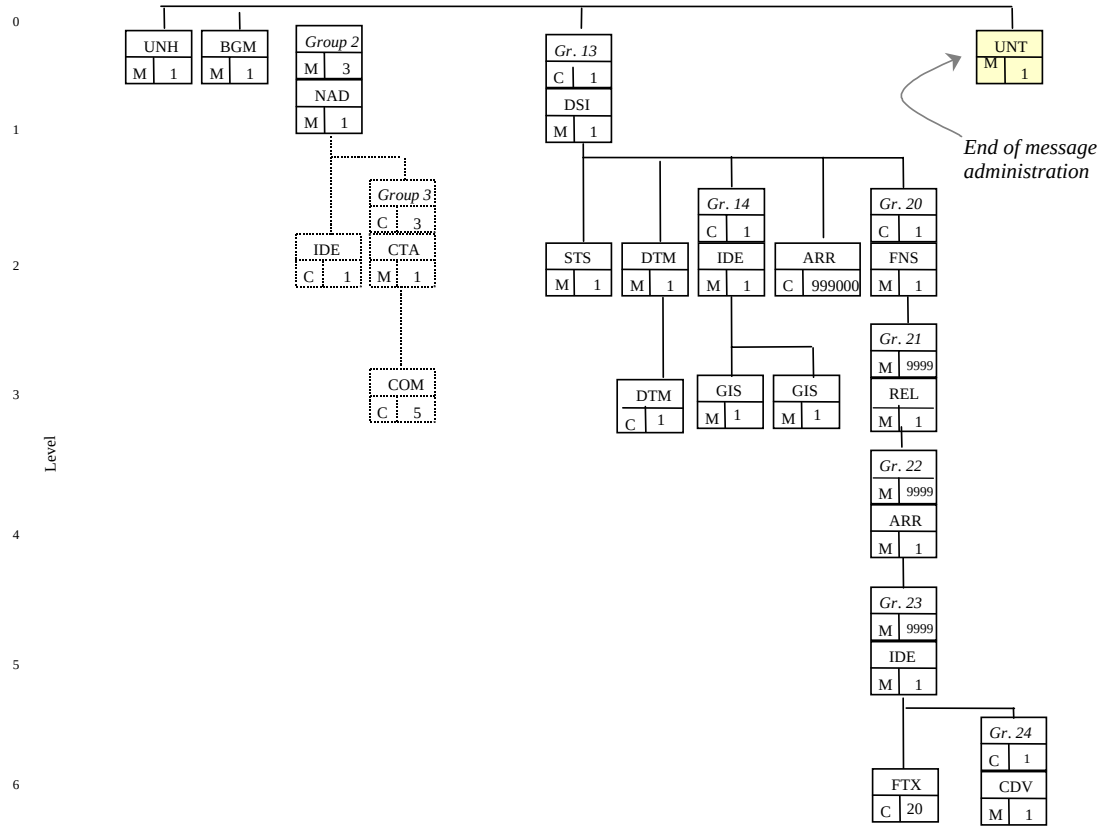
2434 **Example 4 - attribute for an individual observation:**

REL+Z01+5'	Array scope is "observation"
ARR+6+M:XX:ZZ:CC:199606:610'	attached to obs. of June 1996 for series M:XX:ZZ:CC
IDE+Z11+OBS_COM'	uncoded attribute OBS_COM
FTX+ACM**+++**NEW ACCOUNTING METHOD'	attribute text value

2435

9.10 End of Message Administration Section

The UNT segment ends the message and contains control count data that can be used to check the integrity of the data received.



Segment Type: **UNT**

Segment Type Name: *Message Trailer*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

UNT+*number-of-segments+message-reference-number*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
number-of-segments	mandatory this is a total count of all the segments contained in the message, including the UNH and the UNT segments	n..6	
message-reference-number	mandatory this is the same value as the *message-reference-number* in the UNH at the beginning of the message	an..14	Format in SDMX-EDI: MREFnnnnnn

Content and General Usage Rules

This segment ends the message and contains control data that can be used to validate message integrity.

Example:

UNT+59+MREF000001'

This message contains 59 segments and the UNH which started the message has a message-reference-number of MREF000001

9.11 End of Interchange Administration Section

The UNZ segment ends the interchange (UNZ) and contains control count data that can be used to check the integrity of the messages received.

Segment Type: **UNZ**

Segment Type Name: *Interchange Trailer*

Max. Number of Occurrences: *1*

Status: *Mandatory*

Data Element Sequence

UNZ+*number-of-messages+interchange-reference*'

Data Element Usage Rules

local name	usage rules	format	SDMX-EDI code values
number-of-messages	mandatory this is a count of the number of messages (i.e. UNH/UNT pairs) that are in the interchange	n..6	
interchange-reference	mandatory this must be the same value that is placed in the *interchange-reference* in the UNB at the beginning of this interchange	an..14	Format: IREFnnnnnn

Content and General Usage Rules

This segment ends the interchange and it is put immediately following the last UNT of the last message of the interchange. It contains a count of the number of messages in the interchange and a reference number that can be used to validate the integrity of the interchange.

Example:

UNZ+3+IREF000006'

There are three messages in the interchange and the value of the interchange-reference in the UNB that relates to this UNZ is IREF000006

10 DELETE MESSAGES

A SDMX-EDI message turns into a message containing *delete directives* when its STS segment has been written as:

STS+3+6'

Once this segment has been found, the message is considered as a **delete message.** All its ARR segments should then contain exclusively references to values of objects which should be deleted; all references for deletion should refer exclusively to statistical objects belonging or related to the data set indicated by the DSI segment of this message.

10.1 Rules and Techniques for Deletions

References for deletions can appear in two different parts of a “delete” message and they are discussed separately below:

10.1.1 Deletions of observations, time series, sibling groups and data sets

These deletions are activated via references appearing in the main set of ARR segments (FOLLOWING GROUP 14) of the message

Rule*: numeric observations cannot be present in the ARR segments of the message.*

Deletion of specific observations and the corresponding observation flags (observation status, observation confidentiality, observation pre-break);

recommendation: deletions which could create conceptually unjustified “holes” within a time series composed of continuous observations should be avoided;

e.g. if M:XXX:YYY and M:XXX:ZZZ are the keys of two series, then the segments

ARR++M:XXX:YYY:199201:610' and

ARR++M:XXX:ZZZ:199203:610'

imply the deletion of the Jan92-observation (and the corresponding obs. flags) for the series M:XXX:YYY and the deletion of the Mar93-observation (and the corresponding obs. flags) for the series M:XXX:ZZZ

Deletion of ranges of observations and the corresponding observation flags (observation status, observation confidentiality, observation pre-break);

recommendation: deletions which could create conceptually unjustified “holes” within a time series composed of continuous observations should be avoided;

e.g. if M:XXX:YYY is the key of the series, then the segment ARR++M:XXX:YYY:199201199205:710'

implies the deletion of the observations of the series (and its attributes) from Jan.1992 to May.1992.

Deletion of a specific time series;

Rule: dates/periods/time ranges cannot be present in segment;

e.g. if M:BE:XXX:YYY is the key of a series, then the segment

ARR++M:BE:XXX:YYY'

implies the deletion of this series and all its attribute values at all levels not higher than the time series level (incl. the coded and uncoded attribute values)

Deletion of a group of sibling series;

Rule: dates/periods/time ranges cannot be present in segment;

e.g. if :BE:XXX:YYY is the key of the sibling group (second position: frequency wildcarded: for any frequency), then the segment

ARR++:BE:XXX:YYY'

implies the deletion of all series of the group (e.g. Q:BE:XXX:YYY and M:BE:XXX:YYY) and all their attribute values at all levels not higher than the sibling group level (incl. the coded and uncoded attribute values).

Deletion of a whole data set;

Rule: dates/periods/time ranges cannot be present in segment;

In this case, the ARR segment should be written in the following way: ARR+0'

This segment should imply the deletion of all series and all attribute values of the data set indicated by the DSI segment of the message.

10.1.2 Deletions of attributes

These deletions are activated via references appearing in the set of Group 21-ARR segments:

***Rule**: both the ARR and IDE segments should be present in a delete message, but CDV or FTX segments (which follow in a normal update message) cannot be present in this case.*

Deletions of specific observation comments (Obs_Com) at the observation level

e.g. the segments

ARR+7+M:ABB:A:DE:S1:199606:610'

IDE+Z11+OBS_COM'

imply the deletion of the observation comment value for Jun.96 for the series M:ABB:A:DE:S1.

Deletions of specific attributes at a series level;

e.g. the segments

ARR+5+M:ABB:A:DE:S1'

IDE+Z11+COVERAGE'

imply the deletion of the value of the attribute COVERAGE for the series M:ABB:A:DE:S1.

Deletions of specific attributes at a sibling group level;

e.g. if in this key family the attribute COVERAGE has been defined at the sibling group level and frequency is the first dimension of the key family, then the segments

ARR+5+:ABB:A:DE:S1'

IDE+Z11+COVERAGE'

imply the deletion of the value of the attribute COVERAGE for the sibling group :ABB:A:DE:S1.

Deletion of specific attributes at a data set level;

e.g. if in this key family the attribute OTHER_METH_EXPL has been defined at the data set level, then the segments

ARR+0'

IDE+Z11+OTHER_METH_EXPL'

imply the deletion of the text for the attribute OTHER_METH_EXPL .

Remark: it is assumed that, before writing the ARR/IDE pairs, appropriate settings have been used for the REL segment: REL+Z01+4' for deleting time series' and sibling groups' attributes, REL+Z01+5' for deleting observation comments and REL+Z01+1' for data set attributes' deletions.

10.2 Comments on Deletions

Deletions affect:

- objects identified by the ARR segment which comes immediately after Group 14 (for deletion of observations, series, sibling groups and data sets);
- specific attribute values identified via references in the ARR segment of Group 22 (in conjunction with a corresponding attribute identifier).

Deletion of observations or ranges of observation should not create “holes” in a series containing continuous observations or if these holes could not be conceptually justified. Usually, an observation deletion is expected at the start or the end of a series. For example, should a need arise to temporarily “delete” an observation in the middle of a series with continuous observations (for example if the value is wrong and the correct one is not known yet), an “update” instruction should be used instead: a “missing value” should be sent with an appropriate missing value explanation flag (a value of the corresponding OBS_STATUS attribute indicating the reasoning for the reported missing value).

The deletion of a whole data set can be initiated only by the centre which informs the receiving ends in advance, using other administrative means, providing a timetable and describing the concrete actions which should take place. The deletion of a data set implies the deletion of all series included in the data set and of their corresponding attribute values at all levels (observation, time series, sibling, data set). The key family, from which the data set took its structure, is not affected by the deletion of the data set.

The deletion of structural definitions (e.g. key families) using automated means is not foreseen. If such a need arises, administrative guidance will be provided by the centre administrating the corresponding data exchange (if the institution administrating the data exchange is different from the institution devising the structural definitions used, a prior consultation between the two central institutions might be needed).

11 MESSAGE ACKNOWLEDGEMENTS

SDMX-EDI does not currently support message acknowledgements.

Message acknowledgement services provided by a centre may differ between centres.

12 TEXT IN UNCODED ATTRIBUTES: RULES AND CONSIDERATIONS

12.1 Character Set

In a SDMX-EDI message the upper case Latin character set, the service characters (: ' ? +), the dash (used in negative numbers and to denote missing values) and the dot (decimal point in numbers) are used; **text given as a value to an uncoded attribute can also use any character of the standard character set (e.g. lower case characters) and characters of the** extended set (160-255), e.g. é, ñ, ê, ö, ç. However, the use of the extended character set needs to be evaluated before being put in production, as it could create interoperability problems between partners' applications.

When a need arises to use another character set beyond Latin (e.g. Greek, Cyrillic, Japanese), then Latin characters should be used chosen according to the closest sound of the characters.

12.2 Special Characters

***Special non-alphanumeric characters in uncoded attributes should be used with caution**:*

beyond the ANSI/ASCII code number 126, only the ANSI-accented-characters are allowed especially, control characters (coded in positions 128 - 159) must not be used inside a SDMX-EDI message, as they cause problems in applications.

The release character ("?" in SDMX-EDI) must precede any appearance of the SDMX-EDI service characters (+':?), but it should not precede decimal points; see also Appendix on the EDIFACT syntax. The release character should not be included in the character count for FTX component elements.

12.3 Uncoded Attribute Values: Maximum Length of Text

The length of the text that can be used in an uncoded attribute value is specified in the key family definition. This issue is discussed extensively on page 64 and 127 (providing also a number of examples): a FTX segment can contain up to 350 characters (broken down to five component elements of maximum 70 characters of original text each[1]). The general recommendation for uncoded attributes is not to use more than one FTX segment per uncoded attribute, therefore, the resulting string not to exceed 350 characters of original text in total.

[1] Release characters are not counted in the 70 characters so that a component element in an FTX segment may actually be longer than 70 characters, depending on the number of release characters that had to be inserted.

13 EXAMPLE MESSAGES

13.1 Data Update Message

Let us assume that the following data have to be sent by the National Bank of Belgium to the ECB:

Key family: **PRICES_TEST_DATA** - Monthly data for five time series

	M:BE:PROD:	M:BE:CONS:	M:BE:WHOL:	M:BE:WHOL:	M:BE:WHOL:
Sep.95	GN:NS	GN:NS	GN:NS	RM:NS	RM:SA
Oct.95	99.10 A	112.20 A	111.80 A	110.20 A	102.90 A
Nov.95	98.10 A	112.30 A	112.10 A	110.10 A	103.00 A
Dec.95	98.40 A	112.40 A	111.90 A	110.00 A	104.60 A
Jan.96	99.50 A	112.50 A	112.10 A	110.10 A	106.30 A
Feb.96	100.00 A	112.70 A	112.40 A	110.80 A	109.00 A
Mar.96	99.20 A	113.10 A	112.30 A	111.00 A	111.10 A
	99.80 A	113.80 A	112.00 A	110.10 A	109.80 A

Key family: PRICES_TEST_DATA - Quarterly data for five time series

Series key:	Q:BE:PROD:GN:NS	Q:BE:CONS:GN:NS	Q:BE:WHOL:GN:NS	Q:BE:WHOL:RM:NS	Q:BE:WHOL:RM:SA
95q4	98.67 A	112.40 A	112.03 A	110.07 A	104.63 A
96q1	99.67 A	113.20 A	112.23 A	110.63 A	109.97 A

Key family: **INT_RATES_KF,** series: D:BE:IR:MM:THRM (211 daily observations: 1/9/95 to 29/3/96)

01. Sep.95	4.31 A	*11. Sep.95*	4.22 A	...	...
02. Sep.95	(na) H	*12. Sep.95*	4.23 A	*etc.*	
03. Sep.95	(na) H	*13. Sep.95*	4.17 A	...	...
04. Sep.95	4.30 A	*14. Sep.95*	4.16 A	*23. Mar.96*	(na) H
05. Sep.95	4.32 A	*15. Sep.95*	4.16 A	*24. Mar.96*	(na) H
06. Sep.95	4.25 A	*16. Sep.95*	(na) H	*25. Mar.96*	3.31 A
07. Sep.95	4.25 A	*17. Sep.95*	(na) H	*26. Mar.96*	3.31 A
08. Sep.95	4.23 A	*18. Sep.95*	4.16 A	*27. Mar.96*	3.31 E
09. Sep.95	(na) H	...	...	*28. Mar.96*	3.32 F C
10. Sep.95	(na) H	...	...	*29. Mar.96*	3.33 F C

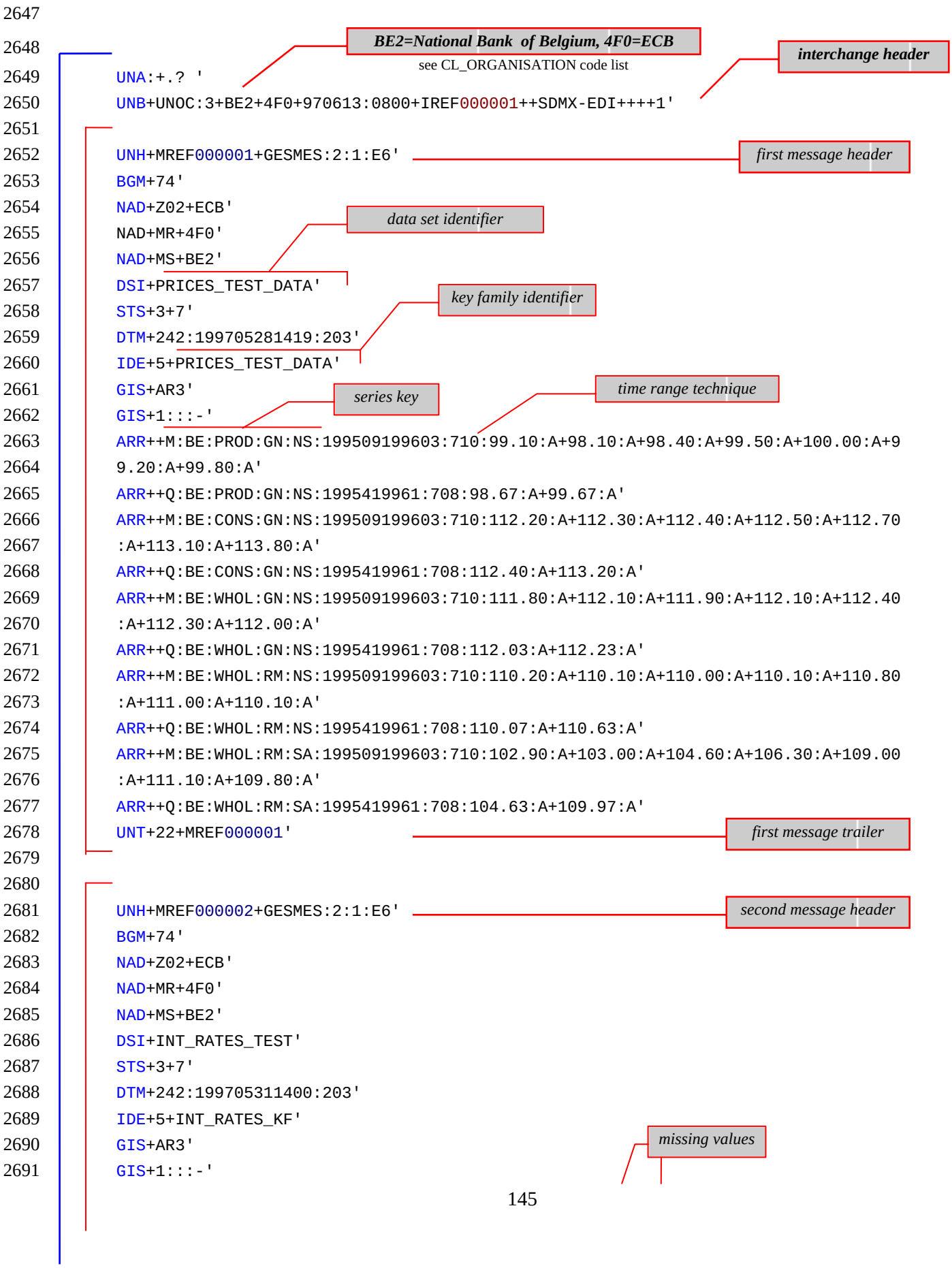
As it is shown on the table above the last two daily observations have been flagged as confidential.

For the series M:BE:IR:MM:THRM the observation 4.10:A for February 1994 was revised and needs to be reported as well.

For the monthly series M:BE:IR:MM:THRM the following observations have also to be sent:

Sep.95	Oct.95	Nov.95	Dec.95	Jan.96	Feb.96	Mar.96
4.24 A	4.22 A	3.98 A	3.84 A	3.53 A	3.32 A	3.31 A

The following sequence of segments is a full interchange containing two messages (one for each key family) and performs the required reporting:




```
UNA:+.? '
UNB+UNOC:3+BE2+4F0+970613:0800+IREF000001++SDMX-EDI++++1'

UNH+MREF000001+GESMES:2:1:E6'
BGM+74'
NAD+Z02+ECB'
NAD+MR+4F0'
NAD+MS+BE2'
DSI+PRICES_TEST_DATA'
STS+3+7'
DTM+242:199705281419:203'
IDE+5+PRICES_TEST_DATA'
GIS+AR3'
GIS+1:::-'
ARR++M:BE:PROD:GN:NS:199509199603:710:99.10:A+98.10:A+98.40:A+99.50:A+100.00:A+9
9.20:A+99.80:A'
ARR++Q:BE:PROD:GN:NS:1995419961:708:98.67:A+99.67:A'
ARR++M:BE:CONS:GN:NS:199509199603:710:112.20:A+112.30:A+112.40:A+112.50:A+112.70
:A+113.10:A+113.80:A'
ARR++Q:BE:CONS:GN:NS:1995419961:708:112.40:A+113.20:A'
ARR++M:BE:WHOL:GN:NS:199509199603:710:111.80:A+112.10:A+111.90:A+112.10:A+112.40
:A+112.30:A+112.00:A'
ARR++Q:BE:WHOL:GN:NS:1995419961:708:112.03:A+112.23:A'
ARR++M:BE:WHOL:RM:NS:199509199603:710:110.20:A+110.10:A+110.00:A+110.10:A+110.80
:A+111.00:A+110.10:A'
ARR++Q:BE:WHOL:RM:NS:1995419961:708:110.07:A+110.63:A'
ARR++M:BE:WHOL:RM:SA:199509199603:710:102.90:A+103.00:A+104.60:A+106.30:A+109.00
:A+111.10:A+109.80:A'
ARR++Q:BE:WHOL:RM:SA:1995419961:708:104.63:A+109.97:A'
UNT+22+MREF000001'

UNH+MREF000002+GESMES:2:1:E6'
BGM+74'
NAD+Z02+ECB'
NAD+MR+4F0'
NAD+MS+BE2'
DSI+INT_RATES_TEST'
STS+3+7'
DTM+242:199705311400:203'
IDE+5+INT_RATES_KF'
GIS+AR3'
GIS+1:::-'
```

```
ARR++D:BE:IR:MM:THRM:1995090119960329:711:4.31:A+-:H+-:H+4.30:A+4.32:A+4.25:A+4.
25:A+4.23:A+-:H+-:H+4.22:A+4.23:A+4.17:A+4.16:A+4.16:A+-:H+-:H+4.16:A+4.15:A+4.1
4:A+4.15:A+4.44:A+-:H+-:H+4.32:A+4.24:A+4.25:A+4.27:A+4.30:A+-:H+-:H+4.24:A+4.23
:A+4.17:A+4.16:A+4.30:A+-:H+-:H+4.25:A+4.25:A+4.24:A+4.22:A+4.20:A+-:H+-:H+4.20:
A+4.21:A+4.26:A+4.23:A+4.25:A+-:H+-:H+4.26:A+4.20:A+4.21:A+4.22:A+4.20:A+-:H+-:H
+4.15:A+4.14:A+-:H+4.12:A+4.11:A+-:H+-:H+4.10:A+4.11:A+4.06:A+4.04:A+4.04:A+-:H+
-:H+-:H+4.01:A+4.00:A+3.98:A+4.00:A+-:H+-:H+3.95:A+3.91:A+3.94:A+3.91:A+3.90:A+-
:H+-:H+3.87:A+3.84:A+3.86:A+3.92:A+3.91:A+-:H+-:H+3.96:A+3.92:A+3.91:A+3.92:A+3.
93:A+-:H+-:H+3.90:A+3.89:A+3.88:A+3.75:A+3.79:A+-:H+-:H+3.79:A+3.81:A+3.79:A+3.8
0:A+3.79:A+-:H+-:H+-:H+-:H+3.76:A+3.74:A+3.73:A+-:H+-:H+-:H+3.71:A+3.69:A+3.69:A
+3.69:A+-:H+-:H+3.69:A+3.67:A+3.63:A+3.61:A+3.58:A+-:H+-:H+3.58:A+3.50:A+3.47:A+
3.45:A+3.44:A+-:H+-:H+3.43:A+3.44:A+3.41:A+3.39:A+3.44:A+-:H+-:H+3.43:A+3.41:A+3
.33:A+3.28:A+3.25:A+-:H+-:H+3.29:A+3.31:A+3.27:A+3.27:A+3.28:A+-:H+-:H+3.27:A+3.
27:A+3.32:A+3.32:A+3.33:A+-:H+-:H+3.42:A+3.42:A+3.42:A+3.36:A+3.31:A+-:H+-:H+3.3
4:A+3.35:A+3.31:A+3.32:A+3.31:A+-:H+-:H+3.30:A+3.30:A+3.31:A+3.30:A+3.29:A+-:H+-
:H+3.34:A+3.31:A+3.31:A+3.31:A+3.31:A+-:H+-:H+3.32:A+3.31:A+3.32:A+3.31:A+3.31:A
+-:H+-:H+3.31:A+3.31:A+3.31:E+3.32:F:C+3.33:F:C'
ARR++M:BE:IR:MM:THRM:199402:610:4.10:A'
ARR++M:BE:IR:MM:THRM:199509199603:710:4.24:A+4.22:A+3.98:A+3.84:A+3.53:A+3.32:A+
3.31:A'
UNT+15+MREF000002'

UNZ+2+IREF000001'
```

obs-confid (2nd -optional- attr.)

second message trailer

interchange trailer

In a completely realistic scenario, the *line feed* characters are not needed after the end of each segment and the whole interchange could be transmitted in only *one line*. This is shown below, in the following interchange:

```
UNA:+.? 'UNB+UNOC:3+BE2+4F0+970613:0800+IREF000002++SDMX-
EDI++++1'UNH+MREF000001+GESMES:2:1:E6'BGM+74'NAD+Z02+ECB'NAD+MR+4F0'NAD+MS+BE2'DSI+PRIC
ES_TEST_DATA'STS+3+7'DTM+242:199705281419:203'IDE+5+PRICES_TEST_DATA'GIS+AR3'GIS+1:::-'
ARR++M:BE:PROD:GN:NS:199509199603:710:99.10:A+98.10:A+98.40:A+99.50:A+100.00:A+99.20:A+
99.80:A'ARR++Q:BE:PROD:GN:NS:1995419961:708:98.67:A+99.67:A'ARR++M:BE:CONS:GN:NS:199509
199603:710:112.20:A+112.30:A+112.40:A+112.50:A+112.70:A+113.10:A+113.80:A'ARR++Q:BE:CON
S:GN:NS:1995419961:708:112.40:A+113.20:A'ARR++M:BE:WHOL:GN:NS:199509199603:710:111.80:A
+112.10:A+111.90:A+112.10:A+112.40:A+112.30:A+112.00:A'ARR++Q:BE:WHOL:GN:NS:1995419961:
708:112.03:A+112.23:A'ARR++M:BE:WHOL:RM:NS:199509199603:710:110.20:A+110.10:A+110.00:A+
110.10:A+110.80:A+111.00:A+110.10:A'ARR++Q:BE:WHOL:RM:NS:1995419961:708:110.07:A+110.63
:A'ARR++M:BE:WHOL:RM:SA:199509199603:710:102.90:A+103.00:A+104.60:A+106.30:A+109.00:A+1
11.10:A+109.80:A'ARR++Q:BE:WHOL:RM:SA:1995419961:708:104.63:A+109.97:A'UNT+22+MREF00000
1'UNH+MREF000002+GESMES:2:1:E6'BGM+74'NAD+Z02+ECB'NAD+MR+4F0'NAD+MS+BE2'DSI+DAILY_INT_R
```

```
ATES_TEST'STS+3+7'DTM+242:199705311400:203'IDE+5+DAILY_INT_RATES_TEST'GIS+AR3'GIS+1:::-
'ARR++D:BE:IR:MM:THRM:1995090119960329:711:4.31:A+-:H+-:H+4.30:A+4.32:A+4.25:A+4.25:A+4
.23:A+-:H+-:H+4.22:A+4.23:A+4.17:A+4.16:A+4.16:A+-:H+-:H+4.16:A+4.15:A+4.14:A+4.15:A+4.
44:A+-:H+-:H+4.32:A+4.24:A+4.25:A+4.27:A+4.30:A+-:H+-:H+4.24:A+4.23:A+4.17:A+4.16:A+4.3
0:A+-:H+-:H+4.25:A+4.25:A+4.24:A+4.22:A+4.20:A+-:H+-:H+4.20:A+4.21:A+4.26:A+4.23:A+4.25
:A+-:H+-:H+4.26:A+4.20:A+4.21:A+4.22:A+4.20:A+-:H+-:H+4.15:A+4.14:A+-:H+4.12:A+4.11:A+-
:H+-:H+4.10:A+4.11:A+4.06:A+4.04:A+4.04:A+-:H+-:H+-:H+4.01:A+4.00:A+3.98:A+4.00:A+-:H+-
:H+3.95:A+3.91:A+3.94:A+3.91:A+3.90:A+-:H+-:H+3.87:A+3.84:A+3.86:A+3.92:A+3.91:A+-:H+-:
H+3.96:A+3.92:A+3.91:A+3.92:A+3.93:A+-:H+-:H+3.90:A+3.89:A+3.88:A+3.75:A+3.79:A+-:H+-:H
+3.79:A+3.81:A+3.79:A+3.80:A+3.79:A+-:H+-:H+-:H+-:H+3.76:A+3.74:A+3.73:A+-:H+-:H+-:H+3.
71:A+3.69:A+3.69:A+3.69:A+-:H+-:H+3.69:A+3.67:A+3.63:A+3.61:A+3.58:A+-:H+-:H+3.58:A+3.5
0:A+3.47:A+3.45:A+3.44:A+-:H+-:H+3.43:A+3.44:A+3.41:A+3.39:A+3.44:A+-:H+-:H+3.43:A+3.41
:A+3.33:A+3.28:A+3.25:A+-:H+-:H+3.29:A+3.31:A+3.27:A+3.27:A+3.28:A+-:H+-:H+3.27:A+3.27:
A+3.32:A+3.32:A+3.33:A+-:H+-:H+3.42:A+3.42:A+3.42:A+3.36:A+3.31:A+-:H+-:H+3.34:A+3.35:A
+3.31:A+3.32:A+3.31:A+-:H+-:H+3.30:A+3.30:A+3.31:A+3.30:A+3.29:A+-:H+-:H+3.34:A+3.31:A+
3.31:A+3.31:A+3.31:A+-:H+-:H+3.32:A+3.31:A+3.32:A+3.31:A+3.31:A+-:H+-:H+3.31:A+3.31:A+3
.31:E+3.32:F:C+3.33:F:C'ARR++M:BE:IR:MM:THRM:199402:610:4.10:A'ARR++M:BE:IR:MM:THRM:199
509199603:710:4.24:A+4.22:A+3.98:A+3.84:A+3.53:A+3.32:A+3.31:A'UNT+15+MREF000002'UNZ+2+
IREF000002'
```

13.2 Data and Attribute Update Message

```
UNA:+.? '
UNB+UNOC:3+BE2+4F0+970926:1948+IREF000001++SDMX-EDI++++1'
UNH+MREF000001+GESMES:2:1:E6'
BGM+74'
NAD+Z02+EMI'
NAD+MR+4F0'
NAD+MS+BE2'
IDE+10+Test message (26-Sep-97)'
DSI+EMI_TEST_BOP'
STS+3+7'
DTM+242:199709261948:203'
IDE+5+EMI_TEST_BOP'
GIS+AR3'
GIS+1:::-'
ARR++M:BE:N:2:269:1:199610199703:710:2.51:A+2.55:A+2.62:A+2.73:A+2.84:A+2.92:E'
ARR++M:BE:S:3:982:2:199703199708:710:0.74:A+0.76:A+0.79:A+0.85:A+0.97:A+1.07:A'
ARR++M:BE:N:4:379:3:199706199710:710:4.36:A+4.47:A+4.71:A+4.59:A+3.73:F:C'
FNS+Attributes:10'
REL+Z01+4'
ARR+6+:BE:N:1:379:3'
IDE+Z11+TITLE'
FTX+ACM+++Test virtual title'
IDE+Z10+UNIT'
CDV+BEF'
IDE+Z10+UNIT_MULT'
CDV+6'
ARR+6+M:BE:S:2:379:3'
IDE+Z10+COLLECTION'
CDV+2'
ARR+6+M:BE:N:2:379:2'
IDE+Z10+COLLECTION'
CDV+C'
REL+Z01+5'
ARR+8+M:BE:N:4:379:3:199705:610'
IDE+Z11+OBS_COM'
FTX+ACM+++Test virtual observation comment'
UNT+35+MREF000001'
UNZ+1+IREF000001'
```

interchange administration section

message administration section

data set administration section

array structure and data section

attribute section

end of message administration

end of interchange administration

2804

2805 The message above contains data and attributes:

2806 • some numeric values (array structure and data section) are given for the series
2807 M:BE:N:2:269:1, M:BE:S:3:982:2 and M:BE:N:4:379:3.

2808 • the text “Test virtual title” is assigned as TITLE to the sibling group “:BE:N:1:379:3” (here
2809 “frequency” is assumed to be the first dimension and it is wild-carded); for the same sibling
2810 group, the value “BEF” is assigned to the attribute UNIT and the value “6” to the attribute
2811 UNIT_MULT (unit multiplier).

2812 • the value “C” is assigned to the attribute COLLECTION for the series M:BE:S:2:379:3 and the
2813 same attribute value also for the series M:BE:N:2:379:2.

2814 • the text “Test virtual observation comment” is assigned as OBS_COM (observation comment)
2815 for the May 1995 observation for the series M:BE:N:4:379:3.

2816

2817 ## 13.3 Key Family Definition Message

2818 In the previous example the objects which were used were referring to series which, of course,
2819 belong to a key family. So, let us assume that this key family's definition is reflected in the
2820 following table[1]:

KEY FAMILY: Test key family ref. to balance of payments data
KEY FAMILY MNEMONIC: EMI_TEST_BOP
Release: 1.0, 26 Sep. 1997

		Concept (mnemonic)	Concept name	Code list (mnemonic)	Code list name	code values: N / AN
	ORDER	**KEY FAMILY DIMENSIONS**				
	1	FREQ	Frequency	CL_FREQ	Frequency codelist	AN1
	2	REF_AREA	Reference area	CL_AREA_EE	Area codelist (EUROSTAT-BoP/EMI)	AN2
	3	ADJUSTMENT	Adjustment indicator	CL_ADJUSTMENT	Adjustment indicator codelist	AN1
	4	DATA_TYPE	BoP type/position	CL_DATA_TYPE	BoP type/position codelist	AN1
	5	BOP_ITEM	BoP item	CL_BOP_ITEM	BoP item codelist	AN3
	6	BASIS	Data basis	CL_BASIS	Data basis codelist	AN1
Attachment level	**Status**					
I. Attributes (attached to the observation in the message)						
Observation	*M*	OBS_STATUS	Observation status	CL_OBS_STATUS	Observation status codelist	AN1
Observation	*C*	OBS_CONF	Observation confidentiality	CL_OBS_CONF	Observation confidentiality codelist	AN1
II. Attributes (handled in the FNS group)						
Sibling	*M*	TITLE	Title	*<uncoded>*	*<uncoded>*	AN..70
Sibling	*M*	UNIT	Unit	CL_UNIT	Unit codelist	AN..12
Sibling	*M*	UNIT_MULTIPLIER	Unit multiplier	CL_UNIT_MULT	Unit multiplier codelist	AN..2
Sibling	*M*	DECIMALS	Decimals	CL_DECIMALS	Decimals codelist	AN1
Time series	*M*	COLLECTION	Collection indicator	CL_COLLECTION	Collection indicator codelist	AN1
Time series	*M*	AVAILABILITY	Availability	CL_AVAILABILITY	Organisation availability code list	AN1
Sibling	*M*	COMPILATION	Compilation explanations	*<uncoded>*	*<uncoded>*	AN..70
Observation	*C*	OBS_COM	Observation comment	*<uncoded>*	*<uncoded>*	AN..70
Time series	*C*	BREAKS	Explanation for breaks	*<uncoded>*	*<uncoded>*	AN..70
Sibling	*C*	AGG_EQUN	Aggregation equation used	*<uncoded>*	*<uncoded>*	AN..70
Other concepts used in the reporting/dissemination messages (message administration section)						
		ORGANISATION	Organisation	*CL_ORGANISATION*	*Organisation codelist*	AN3
Other concepts used in the Gesmes/CB key family dissemination message (advanced use)						
Advanced use (ASI group)		USAGE_STATUS	Usage status	*USS*	*Usage status code list*	AN1
Advanced use (ASI group)		ATTACHMENT_LEVEL	Attachment level	*ALV*	*Attachment level code list*	AN1

2821
2822 Each message referring to a key family, like the one presented in the previous example, has to
2823 comply with the corresponding key family definition.

2824 The key family definition table which is presented above could be disseminated from the centre
2825 (e.g. ECB=4F0) to its partners (ZZZ=unspecified) using SDMX-EDI For this, the ASI group of
2826 segments is used. The structural definitions maintenance agency is assumed to be represented
2827 by the code EMI. The message below is the coded version of the previous table:

2828

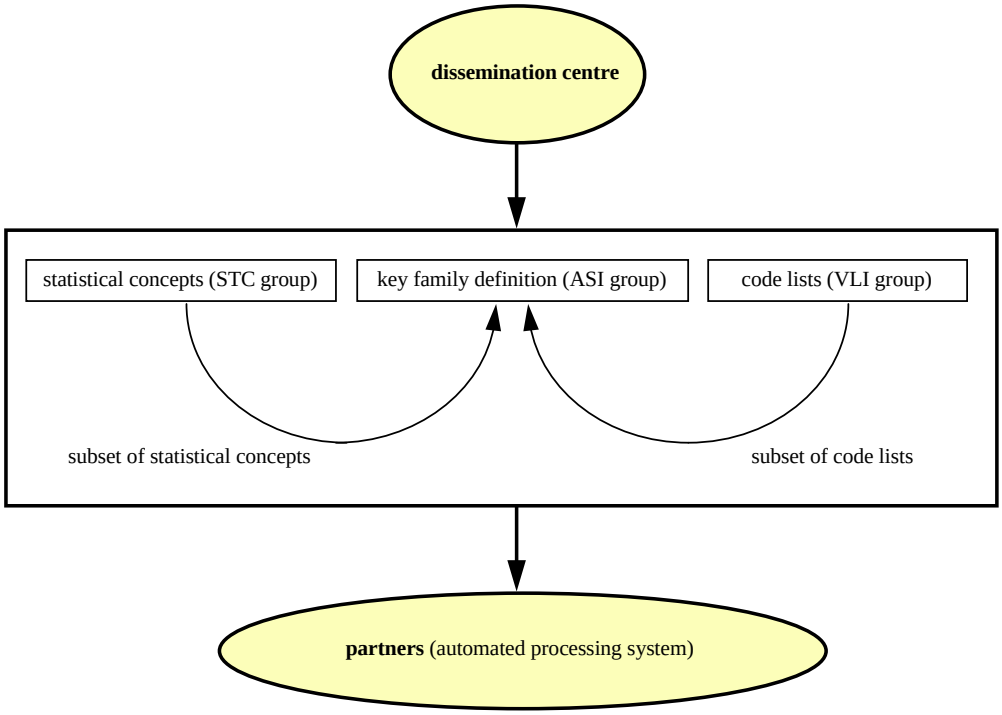
[1] This is just an example, it does not refer to any key family used in actual testing or reporting. Also, for reasons of simplicity, the TIME_PERIOD and TIME_FORMAT concepts are not shown in the table.

```
UNA:+.? '
UNB+UNOC:3+4F0+ZZZ+970525:1539+IREF000003++SDMX-EDI'
UNH+MREF000001+GESMES:2:1:E6'
BGM+73'
NAD+Z02+EMI'
NAD+MR+ZZZ'
NAD+MS+4F0'
IDE+10+Ad hoc message - example key family'
ASI+EMI_TEST_BOP'
FTX+ACM+++Test kf for BoP Statistics'
SCD+13+FREQ++++:1'ATT+3+5+:::AN1'IDE+1+CL_FREQ'
SCD+4+REF_AREA++++:2'ATT+3+5+:::AN2'IDE+1+CL_AREA_EE'
SCD+4+ADJUSTMENT++++:3'ATT+3+5+:::AN1'IDE+1+CL_ADJUSTMENT'
SCD+4+DATA_TYPE++++:4'ATT+3+5+:::AN1'IDE+1+CL_DATA_TYPE'
SCD+4+BOP_ITEM++++:5'ATT+3+5+:::AN3'IDE+1+CL_BOP_ITEM'
SCD+4+BASIS++++:6'ATT+3+5+:::AN1'IDE+1+CL_BASIS'
SCD+1+TIME_PERIOD++++:7'ATT+3+5+:::AN..35'
SCD+1+TIME_FORMAT++++:8'ATT+3+5+:::AN3'
SCD+3+OBS_VALUE++++:9'ATT+3+5+:::AN..15'
SCD+3+OBS_STATUS++++:10'ATT+3+5+:::AN1'ATT+3+35+2:USS'ATT+3+32+5:ALV'IDE+1+CL_OBS_STATU
S'
SCD+3+OBS_CONF++++:11'ATT+3+5+:::AN1'ATT+3+35+1:USS'ATT+3+32+5:ALV'IDE+1+CL_OBS_CONF'
SCD+Z09+TITLE'ATT+3+5+:::AN..70'ATT+3+35+2:USS'ATT+3+32+9:ALV'
SCD+Z09+UNIT'ATT+3+5+:::AN..12'ATT+3+35+2:USS'ATT+3+32+9:ALV'IDE+1+CL_UNIT'
SCD+Z09+UNIT_MULTIPLIER'ATT+3+5+:::AN..2'ATT+3+35+2:USS'ATT+3+32+9:ALV'IDE+1+CL_UNIT_MU
LT'
SCD+Z09+DECIMALS'ATT+3+5+:::AN1'ATT+3+35+2:USS'ATT+3+32+9:ALV'IDE+1+CL_DECIMALS'
SCD+Z09+COLLECTION'ATT+3+5+:::AN1'ATT+3+35+2:USS'ATT+3+32+4:ALV'IDE+1+CL_COLLECTION'
SCD+Z09+AVAILABILITY'ATT+3+5+:::AN1'ATT+3+35+2:USS'ATT+3+32+4:ALV'IDE+1+CL_AVAILABILITY
'
SCD+Z09+COMPILATION'ATT+3+5+:::AN..70'ATT+3+35+2:USS'ATT+3+32+9:ALV'
SCD+Z09+BREAKS'ATT+3+5+:::AN..70'ATT+3+35+1:USS'ATT+3+32+4:ALV'
SCD+Z09+AGG_EQUN'ATT+3+5+:::AN..70'ATT+3+35+1:USS'ATT+3+32+9:ALV'
SCD+Z09+OBS_COM'ATT+3+5+:::AN..70'ATT+3+35+1:USS'ATT+3+32+5:ALV'
UNT+88+MREF000001'
UNZ+1+IREF000003'
```

These three elements should be also defined

The partners could build fully automated systems by reading the key family definitions directly from SDMX-EDI messages, like the one above.

In fact, all three SDMX-EDI structural sections (each one could be in a different message), shown in the graph below, are needed in order to allow the centre to disseminate the complete key family definition in SDMX-EDI and the receiving institutions to make a paperless interpretation of the received time series data.



The receiver's system could be automated by "reading" the list of statistical concepts and the relevant code lists and linking them in the context of a key family definition message. The next two examples complete the whole example by providing the list of statistical concepts and the code lists.

13.4 List of Concepts Dissemination Message

The example message shown below contains the concepts which were used in building the previous key family.

```
UNA:+.? '
UNB+UNOC:3+4F0+ZZZ+970525:1539+IREF000001++SDMX-EDI'
UNH+MREF000001+GESMES:2:1:E6'
BGM+73'
NAD+Z02+EMI'
NAD+MR+ZZZ'
NAD+MS+4F0'
IDE+10+Statistical concepts-example list'
STC+REF_AREA'FTX+ACM+++Reference area'
STC+FREQ'FTX+ACM+++Frequency'
STC+ADJUSTMENT'FTX+ACM+++Adjustment indicator'
STC+DATA_TYPE'FTX+ACM+++BoP type/position'
STC+BOP_ITEM'FTX+ACM+++BoP item'
STC+BASIS'FTX+ACM+++Data basis'
STC+TIME_PERIOD'FTX+ACM+++Time period or range'
STC+TIME_FORMAT'FTX+ACM+++Time format code'
STC+OBS_VALUE'FTX+ACM+++Observation value'
STC+TITLE'FTX+ACM+++Title'
STC+UNIT'FTX+ACM+++Unit'
STC+UNIT_MULTIPLIER'FTX+ACM+++Unit multiplier'
```

```
STC+DECIMALS'FTX+ACM+++Decimals'
STC+AVAILABILITY'FTX+ACM+++Availability'
STC+COLLECTION'FTX+ACM+++Collection indicator'
STC+AVAILABILITY'FTX+ACM+++Availability'
STC+COMPILATION'FTX+ACM+++Compilation explanations'
STC+BREAKS'FTX+ACM+++Explanation for breaks'
STC+AGG_EQUN'FTX+ACM+++Aggregation equation used'
STC+OBS_COM'FTX+ACM+++Observation comment'
STC+ORGANISATION'FTX+ACM+++Organisation'
UNT+49+MREF000001'
UNZ+1+IREF000001'
```

13.5 Code List Dissemination Message

The example message shown below contains the code lists. These lists should be used for giving values to the coded statistical concepts of a key family (according to its definition). The message below is for demonstration purposes only: the code lists shown are much shorter than their actual length (e.g. countries and other codes missing).

```
UNA:+.? '
UNB+UNOC:3+4F0+ZZZ+970525:1539+IREF000002++SDMX-EDI'
UNH+MREF000001+GESMES:2:1:E6'
BGM+73'
NAD+Z02+EMI'
NAD+MR+ZZZ'
NAD+MS+4F0'
IDE+10+Message with example code lists'
VLI+CL_FREQ+++Frequency code list'
CDV+M'FTX+ACM+++monthly'
CDV+Q'FTX+ACM+++quarterly'
VLI+CL_AREA_EE+++Area code list (EUROSTAT/ECB)'
CDV+4D'FTX+ACM+++European Commission'
CDV+4F'FTX+ACM+++European Monetary Institute'
CDV+5B'FTX+ACM+++Bank for International Settlements'
CDV+BE'FTX+ACM+++Belgium'
CDV+DK'FTX+ACM+++Denmark'
CDV+FR'FTX+ACM+++France'
CDV+GB'FTX+ACM+++United Kingdom'
CDV+GR'FTX+ACM+++Greece'
CDV+LU'FTX+ACM+++Luxembourg'
CDV+NO'FTX+ACM+++Norway'
CDV+US'FTX+ACM+++United States of America'
VLI+CL_ADJUSTMENT+++Adjustment indicator code list'
CDV+N'FTX+ACM+++neither seasonally or working day adjusted'
CDV+S'FTX+ACM+++seasonally adjusted, not working day adjusted'
CDV+W'FTX+ACM+++working day adjusted, not seasonally adjusted'
CDV+Y'FTX+ACM+++working day and seasonally adjusted'
VLI+CL_DATA_TYPE+++BoP type/position code list'
CDV+1'FTX+ACM+++Stocks at the beginning of the period'
CDV+2'FTX+ACM+++Credit flows'
CDV+3'FTX+ACM+++Debit flows'
CDV+4'FTX+ACM+++Net flows'
CDV+5'FTX+ACM+++Price valuation adjustment'
CDV+6'FTX+ACM+++Exchange rate adjustments'
CDV+7'FTX+ACM+++Reclassification, other adjustments and other revaluations'
CDV+8'FTX+ACM+++Stocks at the end of the period / Gross stocks'
VLI+CL_BOP_ITEM+++BoP item code list'
CDV+269'FTX+ACM+++Merchanting and other trade-related services '
CDV+379'FTX+ACM+++Current transfers'
CDV+971'FTX+ACM+++Freight transport by air and sea'
CDV+981'FTX+ACM+++Other services'
CDV+982'FTX+ACM+++Services not allocated'
CDV+991'FTX+ACM+++Goods and services'
```

```
2965        VLI+CL_BASIS+++Data basis code list'
2966        CDV+1'FTX+ACM+++Transaction'
2967        CDV+2'FTX+ACM+++Cash'
2968        CDV+3'FTX+ACM+++Accruals'
2969        CDV+Z'FTX+ACM+++Other'
2970        VLI+CL_OBS_STATUS+++Observation status code list'
2971        CDV+A'FTX+ACM+++normal value'
2972        CDV+E'FTX+ACM+++estimated value'
2973        CDV+F'FTX+ACM+++forecast value'
2974        CDV+H'FTX+ACM+++missing, holiday or weekend'
2975        VLI+CL_OBS_CONF+++Observation confidentiality code list'
2976        CDV+F'FTX+ACM+++Free'
2977        CDV+C'FTX+ACM+++Confidential (and non-publishable)'
2978        VLI+CL_UNIT+++Unit code list'
2979        CDV+BEF'FTX+ACM+++Belgian Franc'
2980        CDV+DEM'FTX+ACM+++German Deutsche Mark'
2981        CDV+ITL'FTX+ACM+++Italian Lira'
2982        VLI+CL_UNIT_MULT+++Unit multiplier code list'
2983        CDV+6'FTX+ACM+++millions'
2984        CDV+9'FTX+ACM+++billions'
2985        VLI+CL_COLLECTION+++Collection indicator code list'
2986        CDV+C'FTX+ACM+++end of period'
2987        CDV+M'FTX+ACM+++average of observations through period'
2988        VLI+CL_ORGANISATION+++Organisation code list'
2989        CDV+BE2'FTX+ACM+++Banque Nationale de Belgique (Belgium)'
2990        CDV+DK2'FTX+ACM+++Danmarks Nationalbank (Denmark)'
2991        CDV+DE2'FTX+ACM+++Deutsche Bundesbank (Germany)'
2992        CDV+4F0'FTX+ACM+++European Central Bank (ECB)'
2993        CDV+5B0'FTX+ACM+++Bank for International Settlements (BIS)'
2994        CDV+4D0'FTX+ACM+++European Commission (Eurostat)'
2995        CDV+ZZZ'FTX+ACM+++Unspecified'
2996        VLI+USS+++Usage status code list'
2997        CDV+1'FTX+ACM+++Conditional'
2998        CDV+2'FTX+ACM+++Mandatory'
2999        VLI+ALV+++Attachment level code list'
3000        CDV+1'FTX+ACM+++Key family'
3001        CDV+4'FTX+ACM+++Time series'
3002        CDV+5'FTX+ACM+++Observation'
3003        CDV+9'FTX+ACM+++Sibling group'
3004        UNT+143+MREF000001'
3005        UNZ+1+IREF000002'
```

14 APPENDIX: THE UN/EDIFACT SYNTAX

14.1 Introduction

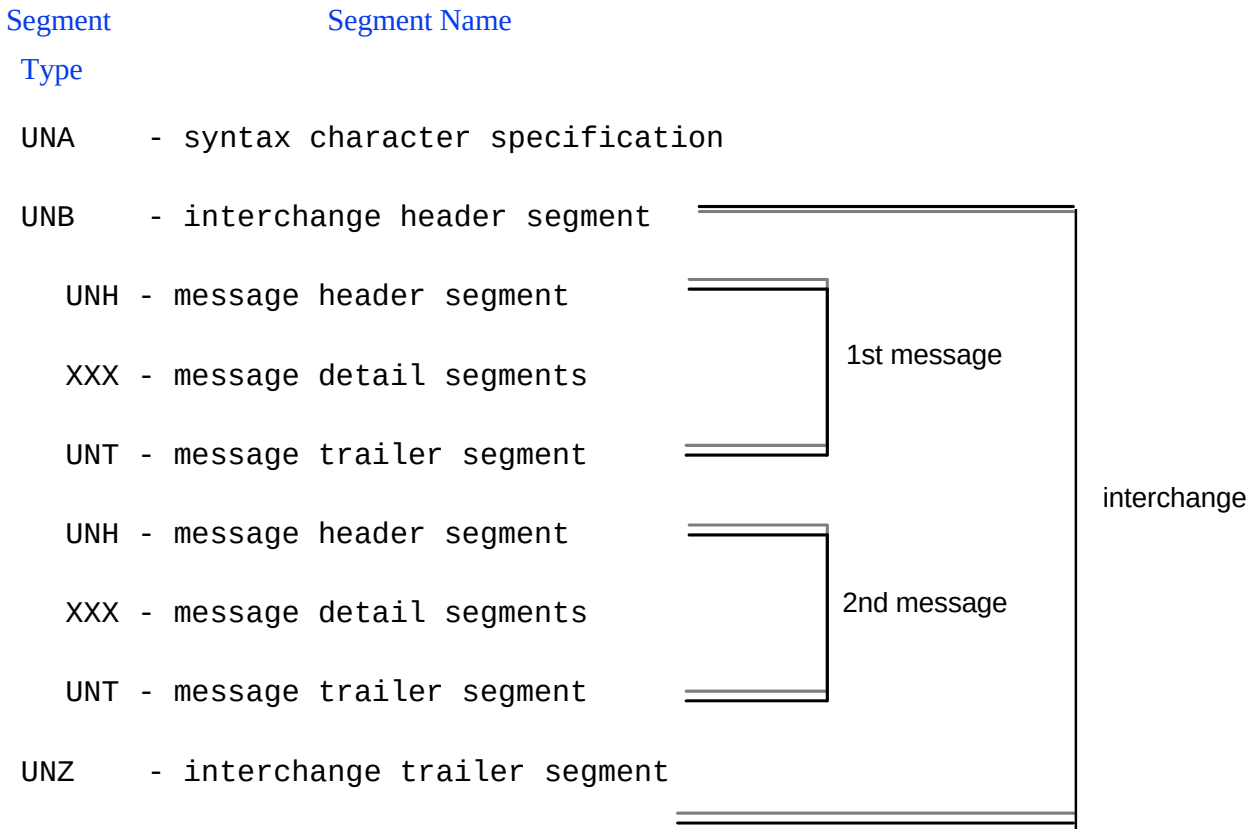
The specification of the EDIFACT syntax is contained in the ISO document ISO 9735 Reference number ISO 9735: 1988 (E). The latest version of this document is dated 1990-11-01.

This explanation given below is a simplified version of some of the sections of ISO 9735. The purpose of this explanation is to highlight the parts of ISO 9735 which are relevant to the use of GESMES explained in this guide. In general, if an organisation wishes to process a GESMES message directly from an application (rather than use a commercial EDIFACT translator), then particular attention should be given to the EDIFACT truncation rules (though, the "fixed" rules of SDMX-EDI incorporate and imply in fact these truncation rules).

Organisations are advised to read ISO 9735 if a fuller understanding of the EDIFACT syntax is required.

14.2 EDIFACT Interchange and Message Structure

The structure of an EDIFACT interchange is shown below.



An EDIFACT interchange comprises a sequence of segments. Each segment has a unique 3 character identifier. Segment identifiers which begin with UN are called "service segments" and are defined as a part of the EDIFACT Syntax (ISO 9735). Other segments are called User Data Segments and are defined in the Segments Directory of the UN Trade Data Interchange Directory (UNTDID).

14.3 Purpose of the interchange and message envelope segments

UNA - Service String Advice

In general, in GESMES, this segment is conditional and is only used if the sender wishes to define the service characters used in the interchange. Its use is advised unless a restricted character set is required (e.g. upper case only). **In SDMX-EDI this segment is mandatory** and the specified set is :+.? '

UNB - Interchange Header

This contains information which identifies the interchange, the sender, receiver, date and time of preparation and other optional data. An interchange can contain many messages.

UNH - Message Header

This contains information which identifies the type and version number of the message that follows.

User Defined Segments

This contains one or more segments that have been designed to support the business function required of the message (e.g. DTM, NAD, DSI etc.).

UNT - Message Trailer

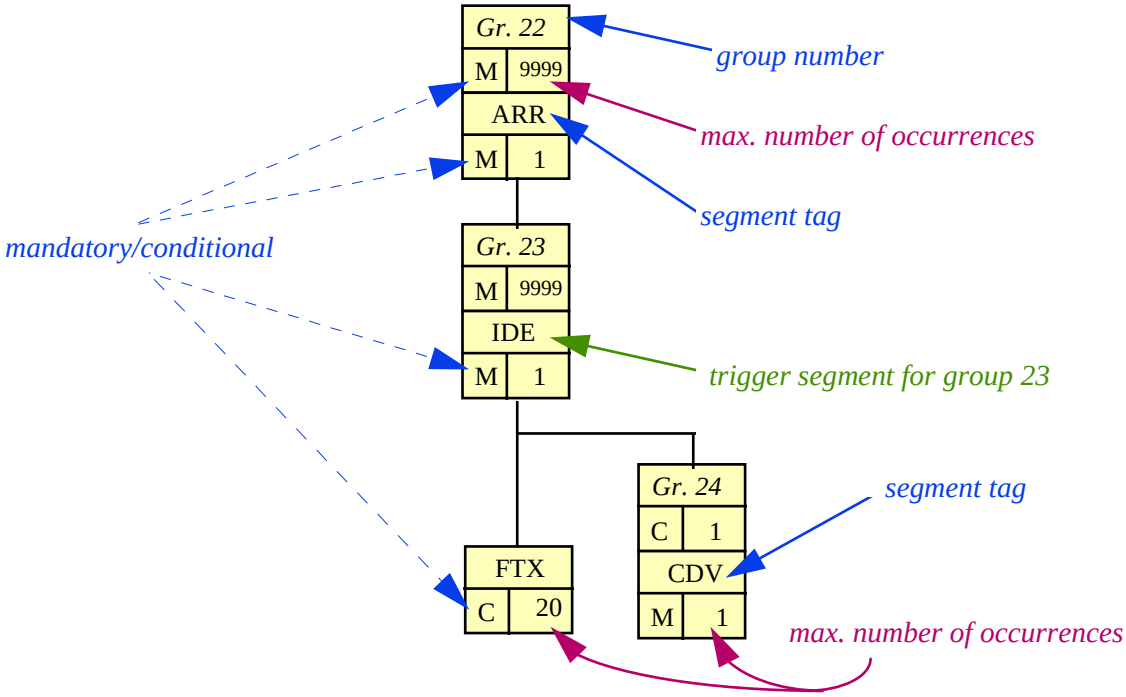
This indicates the end of the message identified in the UNH and contains data which supports the checking of the integrity of the data in the message.

UNZ - Interchange Trailer

This indicates the end of the interchange identified in the UNB, and contains data which supports the checking of the integrity of the interchange.

14.4 Interpreting a Branching Diagram

An EDIFACT message can be represented in a diagrammatic form called a branching diagram. An extract from the branching diagram of SDMX-EDI is shown below.



Group 22 is the collection of segments which is used to assign values to an attribute(s) which is (are) attached at a level or object defined in the ARR segment (group 22); the attribute(s) is (are) identified in the IDE segment and its (their) value(s) is (are) given in the FTX (if the attribute is uncoded) or in the CDV segment (if it is coded). Group 22 can appear up to 9,999 times.

14.4.1 Interpretation of the Diagram

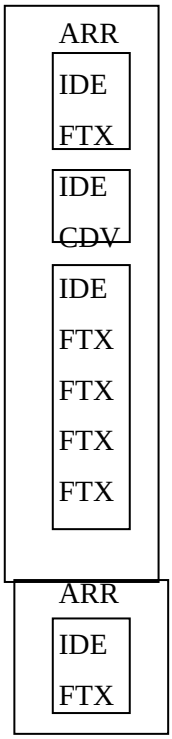
The diagram is interpreted hierarchically. The group number is not sent in the message, the existence of the group in a particular transmission is indicated by the existence of the first segment in the group - this is called the “trigger" segment. The trigger segment is always mandatory if the group is used and can only occur once for each occurrence of the group. The trigger segment for group 22 is the ARR segment. It is followed by between 1 and 9999 occurrences of group 23. Group 23 comprises between 1 to 20 occurrences of the FTX segment or 1 occurrence of the Group 24 (here it is not a “genuine” group as it comprises only one segment; in SDMX-EDI it is shown as a *group* for reasons of presentation consistency with generic GESMES, in which Group 24 includes more segments). An example sequence of segments is shown below.

Example:

The following ordered sequence of segments could be present in a message:

ARR, **IDE**, FTX, **IDE**, CDV, **IDE**, FTX, FTX, FTX, FTX, **ARR**, **IDE**, CDV.

This message contains 2 occurrences of Group 21; the first one contains three occurrences of Group 22 and the second one contains just one occurrence of Group 22. The sequence of these groups and segments (when reading/writing this message) is more clearly illustrated below:



14.5 EDIFACT Service Characters

(For a discussion concerning the use of control characters see Box 2, last section of this Appendix)

The service character set:

The following service characters are used in EDIFACT:

- Segment Terminator
- Data Element Separator
- Component Separator
- Release Indicator

The characters that are used as separators are specified in the transmission (using the UNA segment, first segment of the transmission). If they are not specified then the default characters are used as specified in the EDIFACT syntax. These defaults are dependent upon the character repertoire used. The defaults for the Level A character repertoire are:

+ the data element separator
: the component separator
' the segment terminator
? the release indicator

These service characters are specified in the UNA segment of SDMX-EDI.

Segment and data element separators:

A segment is terminated by the segment terminator.

Each data element is terminated by a data element separator, except the last data element in the segment, which is terminated by a segment terminator. The segment tag is separated from the first data element in the segment by a data element separator.

A component data element is terminated by a component separator, except for the last component in a composite which is terminated by a data element separator. If the component data element is the last data element in the segment then it is terminated by a segment terminator.

Release indicator:

The release indicator is used when it is necessary to transmit one of the separator, release or terminator characters as data. It is sent immediately before the character to be released and is valid for the following character only.

e.g. consider the following text:
Is today's temperature more than +10 degrees?

This would be sent in an EDIFACT message as follows:
Is today?'s temperature more than ?+10 degrees??

14.6 EDIFACT Truncation Rules

The EDIFACT syntax uses truncation rules to eliminate redundant data. Software creating EDIFACT messages must observe these rules. In order to explain them, the following segment is used:

```
TAG+DE1+CE1:CE2:CE3:CE4+DE3+DE4'
```

TAG *the segment tag (e.g. NAD)*
DE *a data element*
CE *a component element*
\+ *the data element separator*

: *the component separator*

' *the segment terminator*

The truncation rules are:

i) Leading zeros are eliminated from numeric data/component elements and trailing spaces are eliminated from alphabetic or alphanumeric data/component elements if the data/component element is defined as variable length.

ii) If no data exists for a data/component element then the relevant separator follows immediately after the separator for the previous data/component element.

iii) A composite element is terminated after the last component element for which there are data for the composite.

```
TAG+DE1+CE1:CE2:CE3:CE4++DE4'
```

data element (DE3) is omitted
(i.e. there are no data for DE3)

```
TAG+DE1+CE1::CE3:CE4+DE3+DE4'
```

a component element (CE2) is omitted
(i.e. there are no data for CE2)

iv) If all the component elements of a composite are omitted then this is indicated by the data element separator.

```
TAG+DE1+CE1:CE2+DE3+DE4'
```

the last two component elements (CE3, CE4) are omitted
(i.e. there are no data for CE3 and CE4)

```
TAG+DE1++DE3+DE4'
```

the composite element is omitted
(i.e. there are no data for CE1, CE2, CE3, CE4)

v) The segment is terminated immediately after the last data element for which there are data.

```
TAG+DE1+CE1:CE2+DE3'
```

the segment is terminated before the last data element
(i.e. there are no data for DE4)

```
TAG+DE1+CE1:CE2'
```

the segment is terminated after the second component element of the composite
(i.e. there are no data for CE3, CE4, DE3 and DE4)

3177

3178

BOX 2. CONTROL CHARACTERS IN EDIFACT MESSAGES

- EDIFACT messages are simple, flat, text files.
- A whole message or a whole interchange (containing possibly several messages) can be written in one single line (as one very long string of characters) without any line feed character or other control characters.
- However, as without any *line feed* the messages are not well readable to the human eye, several institutions prefer *to insert a line feed at the end of each segment*.
- It has to be noted that in EDIFACT messages the control characters are not significant and they should not affect the "reading" or the interpretation of a message.
  (List of control characters: {from 0 to 0x1f and 0x7f to 0x9f})
- **Applications which "read" EDIFACT messages must ignore all control characters**, regardless of where such characters are located inside the incoming file.

15 APPENDIX: CODE LISTS FOR OBSERVATION STATUS AND OBS. CONFIDENTIALITY

As a reference, the following two tables present the code lists for the observation status and the observation confidentiality attributes. It is strongly recommended in all implementations of SDMX-EDI to use these two code lists unchanged in order to guarantee interoperability when interpreting the body of the main ARR segments.

Code list for the Observation Status attribute (CL_OBS_STATUS)

Code_Value	Code_Description
A	Normal value
B	Break
E	Estimated value
F	Forecast value
H	Missing value; holiday or weekend
L	Missing value; data exist but were not collected
M	Missing value; data cannot exist
P	Provisional value
S	Strike

When more than one "condition" occurs for the same observation, then the next table should be used: it indicates the level of importance of each specific "event" (e.g. the information that an observation is a "break" is more important than it is an "estimate" and the flag B -rather than E- should be used).

Observation status hierarchy	Relevant in conjunction with...	
	numeric values	**missing values**
B / break	Yes	Yes
M / undefined, data cannot exist		Yes
L / data not collected		Yes
H / holiday or weekend		Yes
S / strike	Yes	Yes
F / forecast value	Yes	
E / estimated value	Yes	
P / provisional value	Yes	
A / normal value	Yes	

Code list for the Observation Confidentiality attribute (CL_OBS_CONF)

Code_Value	Code_Description
C	Non-publishable and confidential
F	Free
N	Non-publishable, but non-confidential
R	Confidential statist. information due to identifiable respondents

16 APPENDIX: FREQUENTLY ASKED QUESTIONS ABOUT SDMX-EDI

Which are the most important "quantitative" restrictions applying to a message?

More than one messages (UNH/UNT) can be present in the same file/interchange (UNB/UNZ).

If a message contains statistical data and/or attributes, only one DSI Group (group no. 13) can be written inside this message. If a message contains statistical definitions, then a DSI cannot be present and one or more repetitions of Groups VLI (no.4), STC (no.9) and ASI (no.10) can appear in the same message. See also below the paragraph on message types.

A message can be either a "deletion" (parameter "6" in STS, see page 95) message or an "update" (parameter "7" in STS) message (not both).

A message can be (only one of the following three types):
a "structural" message (parameter in BGM segment equal to "73") containing code lists (VLI), concepts (STC) and/or key family definitions (ASI);

a normal message containing data/attributes or instructions for deletions (parameter in BGM segment equal to "74", see page 48);

a data set list message (parameter in BGM segment equal to "DSL").

Maximum length of fields (SDMX-EDI limits):

- **Code lists.** Identifiers: an..18, code list names: an..70, code values: an..18, code value descriptions: an..350
- **Statistical concepts.** Identifiers: an..18, concept names: an..70.
- **Key families** and data sets**.** Identifiers: an..18, key family and data set names: an..70.

In practice, an effort is made by centre institutions to keep the length of series keys no longer than 35 characters.

Up to 999,000 ARR segments (following Group 14) can be written in the same message.

In each ARR segment only one time series key can be present.

Up to 9,999 data elements can be written in the same ARR, i.e. a time series with more than 9,999 observations needs to be split over 2 or more ARR segments

Up to 15 positions (n..15) can be used inside a data element for writing a numeric value (including the place needed for the minus sign for negative values and/or the decimal point; the plus sign for positive numbers must never be used).

The only attributes which can be present in the main ARR segment are the observation status (mandatory coded attribute), the observation confidentiality (conditional coded attribute) and the pre-break value (conditional numeric field).

Can an "update" and a "delete" message be present in the same interchange?

Yes.

Is there a way to distinguish between "reporting new data" and "reporting corrections in previously reported data"?

SDMX-EDI is not equipped with a specific mechanism for this and it is left to the receiving application how to process the information contained in a SDMX-EDI interchange. Receiving applications could check - before "writing" on the reception databases - whether each received observation/attribute/time series/sibling group is a new one or a replacement of a previously existing observation/attribute/time series/sibling group. Nevertheless, if in a specific data exchange circuit this was required, the distinction could be served by the second (conditional) DTM segment which carries the "reporting period" information: for example, this segment could be used only in a message carrying new data (a reporting period could be anyway more relevant in this case).

Can spaces be used at the end of strings in the FTX segment?

No, as this could cause interpretation problems: according to the EDIFACT rules, insignificant trailing spaces are eliminated from alphabetic or alphanumeric component elements.

For example, the segment

```
FTX+ACM+++Test kf for BoP Statistics  '
```

is wrong, since it has redundant spaces at the end of the string.

The terms "mandatory" and "conditional" are confusing.

Indeed, without referring to a specific context, it is not possible to explain the meaning of the adjectives mandatory and conditional: their meaning depends on the context in which they are used. The table below provides an overview of their specific meaning in each case.

3255

Context	"mandatory" attributes	"conditional" attributes	Remarks
Data model	The values of mandatory attributes have to become known to partners. From a statistical point of view they are considered *essential* pieces of information to interpret the data.	The values of conditional attributes have to become known to partners, *if* they are or *when* they become available. From a statistical point of view they are considered *important* pieces of information to interpret the data.	The definition about which attributes are mandatory and which ones are conditional is provided by the key family definition.
Exchange: Attributes exchanged using the FNS group	Attribute values are exchanged the first time and then again every time they change ("updates and revision" principle).		
Exchange: attributes exchanged as elements of the ARR segment (and defined at the observation level): - Observation Status - Observation Confidentiality - Pre-break value	A value for the observation status attribute (OBS_STATUS) should be given with the exchange of every single observation, even if its value does not change. A need to update either the observation value or the observation status (or both) must imply (at least) the reporting of both the observation and the observation status.	Values for the observation confidentiality and/or the pre-break attributes are given when: - it is relevant to provide such a value(s) - if the obs. value or the observation status change and a relevant value for the observation confidentiality and/or the pre-break observation should be kept - the values of the observation confidentiality and/or pre-break observation need to be updated	A value for the observation status attribute (OBS_STATUS) should be given with the exchange of every single observation, even if its value does not change.
SDMX-EDI segments EDIFACT segments	A segment is mandatory if it must be used in a message. Sometimes this depends on whether a previous segment or group is present. However, there are segments that must be present in any message (e.g. UNB, BGM).	A segment is conditional if there is the option not to use it. For example, for a message including numeric data (DSI and ARR present), there is the option to include or not to include the segments carrying attributes (FNS segment and other segments of the same group).	

3256

3257

3258

17 APPENDIX: MAP OF ISO 8859-1 (UNOC) CHARACTER SET (LATIN 1 OR "WESTERN")

Standard character set (decimal character codes 32-126):

	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	

Extended character set (decimal character codes 160-255):

	¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

Code positions 128 - 159 are reserved for control purposes only (they should not be used in messages).

18 APPENDIX. EDIFACT CODE LISTS USED IN SDMX-EDI

The table below contains the relevant EDIFACT code values used in SDMX-EDI segments. Please note that the provided code lists are not complete as they list only those values used in SDMX-EDI. The entries are sorted alphabetically by segment.

ATT segment — 9017 - Attribute Qualifier (domain)

3 array structure component

ATT segment — Attribute Qualifier (type)

5 presentation
35 usage status
32 object link type (attachment level)

ATT segment — Attachment level code

1 data set
4 time series
5 observation
9 sibling group

ATT segment — Code list qualifier

ALV attachment level
USS usage status

ATT segment — Attribute status

1 conditional
2 mandatory

BGM segment — Message function

73 statistical definitions
74 statistical data
DSL list of data sets

COM segment — 3155 - Communication channel qualifier

EM electronic mail
TE telephone
FX telefax
XF X.400

CTA segment — 3139 - Contact Function, Coded

CP Responsible person for computer data processing
CF Head of unit for information production
CC Responsible person for information production
CG Head of unit for information dissemination

DTM segment — 2005 - Date/time/period qualifier

242 preparation date
Z02 reporting period

DTM segment 2379 - date/time/period format qualifier

101 YYMMDD
102 CCYYMMDD
201 YYMMDDHHMM
203 CCYYMMDDHHMM
602 CCYY
604 CCYYS
608 CCYYQ
610 CCYYMM
616 CCYYWW
702 CCYY-CCYY
704 CCYYS-CCYYS
708 CCYYQ-CCYYQ
710 CCYYMM-CCYYMM
711 CCYYMMDD-CCYYMMDD
716 CCYYWW-CCYYWW

FTX segment 4451 - Text Subject Qualifier

ACM statistical description

IDE segment Identification Qualifier

1 value list
4 code value
5 data set structure
10 message context
Z10 coded attribute
Z11 uncoded attribute

NAD segment Party type

MS Message sender
MR Message receiver
Z02 Maintenance agency for codes

REL segment 9141 - Relationship Qualifier

Z01 the relationship is with a statistical array

SCD segment 7497 - Component function qualifier

1 array time dimension
3 array cell
4 array dimension (other than time dimension)
13 dimension is “frequency”
Z09 attribute

STS segment 9011 - Status event, coded

6 delete
7 add and replace

VLI segment

3 Coded list

3277

3278
3279