SECTION A

THE TRADACOMS UNGTDI SYNTAX RULES

CONTENTS

SECTION A - THE TRADACOMS UNGTDI SYNTAX RULES

1. INTRODUCTION TO THE SYNTAX RULES

1.1 Background

The TRADACOMS UNGTDI Syntax Rules are a subset of the full SITPRO Syntax Rules which were originally developed for international trade applications. The full name for the Syntax Rules is the United Nations Economic Commission for Europe (UNECE) Guidelines for Trade Data Interchange (GTDI). The ANA now maintains this for the benefit of TRADACOMS users in the UK.

The UNGTDI syntax has been used for the exchanged of trade data in several countries, and was specified by H M Customs and Excise for their Period Entry Exports interchange scheme. This and other international applications are now based on the later UN EDIFACT (EDI for Administration, Commerce and Transport) syntax. For further details of EDIFACT trade messages contact the ANA.

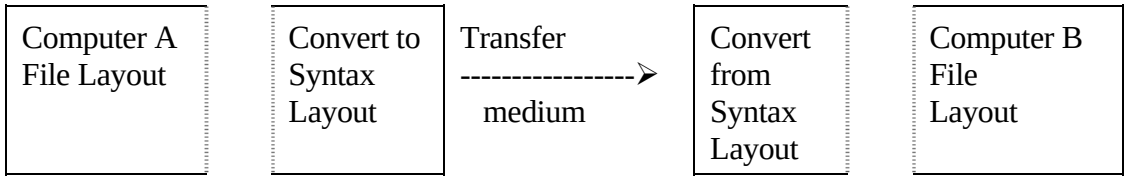
1.2 Scope and Purpose

The Syntax rules provide a method of assembling data elements within messages interchanged between organisations.

The standards for individual data elements are provided in the Standard Data Element Directory in Section B. The standards for the contents of interchange messages are provided in the Message Format Specifications in Volumes 2 and 3. The syntax rules in this Section specify how the data elements in Section B are structured into the message formats in Volumes 2 and 3.

A full understanding of the syntax rules is necessary for the correct interpretation and use, in practice, of the standard message formats.

The syntax rules are independent of machine, media and systems considerations, and are, therefore, suitable for the interchange of data between computer systems, by a telecommunications link or by physical media such as magnetic tapes or floppy discs (diskettes). They are also independent of the structure of data within users' internal computer systems. They are intended as a bridge between systems, for use at the point of interchange. The sender converts from his internal record layouts and the recipient converts into his internal layouts:



Methods for the initial capture of data (in the sender's own system) and the final output of data (in the recipient's own system) are, therefore, outside the scope of the syntax rules, except possibly where the interchange does not involve computer processing at one end of the exchange, as may happen in certain fall-back situations. These are briefly discussed in paras. 3.1.1 to 3.1.4 at the end of this Section.

The syntax rules do not dictate the extent to which data should be validated prior to transmission. **The onus is on the sender of a message to provide accurate information which has been correctly assembled.** The recipient of a message should carry out validation checks, however, to help ensure that invalid data does not corrupt internal computer records.

The syntax rules only apply to the **data** interchanged. They do not apply to the communications protocols and media labels within which the data is transmitted.

Privacy of data is outside the scope of the rules. Where required, encryption or other techniques can be agreed between interchange partners. Facilities are provided, however, for the transmission of control totals in the TRADACOMS Standards message formats, to check the safe transmission of data.

1.3 Advantages

The advantages of using the syntax rules can therefore be summarised as follows:-

1.3.1 Media Independence

Data is structured in the same way whether it is interchanged via magnetic media or a telecommunications link. The syntax rules help minimise the cost of changing from one medium to another in inter-organisation exchange, and the cost to a single organisation sending or receiving data via several types of media.

1.3.2 Machine Independence

Incompatibilities of record size and character representation between different makes of computer are avoided. No reprogramming costs need be incurred by existing members of an interchange application when a new member, who has a different make of computer, joins.

1.3.3 System Independence

The rules are not influenced by the constraints of any particular system design, such as batch, on-line, file/record design or retrieval techniques, communications techniques, etc.

1.3.4 Flexibility

The rules allow a great deal of flexibility in the design and structuring of messages. Changing message specifications should be less costly with data structured using the syntax rules than with conventional records. Using table-driven programs makes changes particularly easy to effect.

1.3.5 Efficiency

The rules allow data to be transmitted in an economical manner (see paragraph 2.9 below). This can be important over telecommunications links.

1.3.6 Intelligibility

By using special character data separators and alpha segment codes, the syntax rules make messages relatively easy to interpret, and thus aid tasks such as investigating errors.

A fuller discussion of the advantages of messages designed according to the TRADACOMS Syntax Rules over conventional computer records, for data exchange, appears at the end of this Section, in paras. 3.2.1 to 3.2.4.

1.4 Messages and Documents

An interchange message can often be considered as the equivalent of a document, for example an invoice or an order. However, other forms of interchange messages are not directly equivalent to conventional documents. Depending on the nature and purpose of the interchange system, these are then equivalent to several documents or to only part of a document, or might have no documentary equivalent at all.

1.5 Syntax Rules Example

The syntax rules can be illustrated by comparing part of a typical order document (Figure A.1) with the equivalent interchange messages assembled according to the rules (Figure A.2). The example messages are in standard TRADACOMS format.

Figure A.1

OFFICIAL ORDER

CUSTOMER NAME:	Any Shop plc	SUPPLIER NAME:	XYZ
ADDRESS:	57 High Street Newtown	ADDRESS:	Manufacturing plc Leeds
LOCATION CODE:	50 00600 003240		
TELEPHONE NO:	TELEX NO:		
CUSTOMER ACCOUNT NO:	68322	LOCATION CODE:	50 00100 00101 4
DELIVERY INSTRUCTIONS: Ring before delivery		SUPPLIER ACCOUNT NO:	7215

Customer Order No:	981
Customer Order Date:	21.3.94
Supplier Order No:	
Supplier Order Date:	
Delivery Date Earliest: 28.3.94 Latest:	

EAN Consumer Unit Code 50 00100	Traded Unit Description	Pack Size	Customer Traded Unit Code	EAN/In-House Code for Traded Unit 50 00100	Qty Ordered
07432 6 06827 1 15406 6	Product A Product B Product C	12 6 4		48145 2 35066 6 15407 3	10 10 30
Conditions - Special Instructions				TOTAL QTY	6

Signature:
Name in Capitals:

Figure A.1 (Cont'd)

OFFICIAL ORDER

CUSTOMER NAME:	Any Shop plc	SUPPLIER NAME:	XYZ
ADDRESS:	32 Market Street Oldtown	ADDRESS:	Manufacturing plc Leeds
LOCATION CODE:	50 00600 00328 2		
TELEPHONE NO:	TELEX NO:		
CUSTOMER ACCOUNT NO:	68347	LOCATION CODE:	50 00100 00101 4
DELIVERY INSTRUCTIONS:		SUPPLIER ACCOUNT NO:	7215

Customer Order No:	982
Customer Order Date:	21.3.94
Supplier Order No:	
Supplier Order Date:	
Delivery Date Earliest: Latest:	

EAN Consumer Unit Code 50 00100	Traded Unit Description	Pack Size	Customer Traded Unit Code	EAN/In-House Code for Traded Unit 50 00100	Qty Ordered
76967 3	Product K	6		76968 0	5
66813 6	Product L	6		65846 5	8
Conditions - Special Instructions				TOTAL QTY	13

Signature:
Name in Capitals:

Figure A.2

```
STX  =  ANA:1+:ANY SHOP PLC+:XYZ MANUFACTURING PLC+940321+REFS+REFR'
MHD  =  1+ORDHDR:9'
TYP  =  0430+NEW-ORDERS'
SDT  =  5000100001014:7215+XYZ MANUFACTURING PLC+LEEDS'
CDT  =  5000600003101:68100+ANY SHOP PLC+HEAD OFFICE:72 KING STREET:
        LONDON EC2'
FIL  =  1+1+940321'
MTR  =  6'
MHD  =  2+ORDERS:9'
CLO  =  5000600003240::68322'
ORD  =  981::940321'
DIN  =  940328+++RING BEFORE DELIVERY'
OLD  =  1+5000100481452+5000100074326++12+10++++PRODUCT A'
OLD  =  2+5000100350666+5000100068271++6+10++++PRODUCT B'
OLD  =  3+5000100154073+5000100154066++4+30++++PRODUCT C'
OTR  =  3+6'
MTR  =  9'
MHD  =  3+ORDERS:9'
CLO  =  5000600003282::68347'
ORD  =  982::940321'
OLD  =  1+5000100769680+5000100769673++6+5++++PRODUCT K'
OLD  =  2+5000100658465+5000100668136++6+8++++PRODUCT L'
OTR  =  2+13'
MTR  =  7'
MHD  =  4+ORDTLR:9'
OFT  =  2'
MTR  =  3'
END  =  4'
```

2. THE SYNTAX RULES

2.1 Introduction

In the following paragraphs:

a) the general terminology used within the specification of the syntax rules is explained

b) the character set recommended for interchange is defined

c) the basic syntax rules are specified.

These basic syntax rules are applicable for the majority of interchange applications, including the TRADACOMS system.

2.2 General Terminology

2.2.1 Data Elements and Sub-Elements

Different organisations use different terminology to describe or define basic items of data. In these rules, the following terms are used:-

a) Data Element

This is an item of data which has been identified separately for interchange by its inclusion in the TRADACOMS Data Element Directory. It may consist of a single item of data, eg. Earliest Delivery Date, in which case it is called a **simple data element,** or it may consist of several items of data, eg. the data element Unit of Ordering. This consists of the number of consumer units in a trade unit, the ordering measure and a measure indicator. In this case it is called a **composite data element.** A data element is identified by its position within a 'segment', defined below.

b) Sub-Element

This is the name given to each item of data within a composite data element. In the example above, measure indicator is a sub-element. A sub-element is identified by its position within a data element.

2.2.2 Data Element Identifiers

Data elements are allocated unique 4 character data element identifiers. These are used to identify data elements in the TRADACOMS Data Element Directory, in message specifications and in the tables used to construct and translate standard TRADACOMS messages. They can also be used for system and program documentation. Data element identifiers are not transmitted.

2.2.3 Segments

a) Related data elements are grouped together within the syntax rules to form a segment. A segment is the basic unit for the transmission of data; individual data elements can only be transmitted within a segment. A segment can contain one or more data elements.

b) There are two types of segment:

- **Data Segments**
- **Standard Segments**

2.2.4 Data Segments

These contain the amounts, values, names, places and other data required for transmission. The contents of data segments are independent of the syntax rules.

2.2.5 Standard Segments

The only category of standard segments that is used in these basic syntax rules, applying to TRADACOMS transmissions, is that of the Transmission Structuring Segments that are used to assemble interchange transmission in a standard way. Standard segments are provided to start and end each interchange, and to start and end each message within an interchange. They are discussed in para. 2.6 of this Section.

2.2.6 Segment Codes

a) Segments are uniquely identified by a 3-character segment code which must be alphabetic.

b) Segment codes have been reserved for each of the standard segments included in the syntax rules.

c) Segment codes are transmitted at the start of the segments they identify.

2.2.7 Interchange Messages

a) An interchange message consists of a number of segments structured in accordance with the syntax rules, ie. it must begin with the standard **message header** segment and end with the standard **message trailer** segment.

b) In the TRADACOMS environment all interchange messages are **Data Messages.** These contain the data segments required for the message in addition to the message header and message trailer segments. The ANA maintains standard message specifications, which are found in Volumes 2 and 3 of this manual.

2.2.8 Message Types

Interchange messages are identified by a **message type** which is included in the standard message header segment. A message type consists of two sub-elements:

- the **message type** itself. This is a 6-character alphabetic code;

- a **version number**. This is a single digit numeric sub-element. It is used in connection with changes to data message specifications, to distinguish one version of a message from another.

2.2.9 Interchange Transmission

a) All the data to be transmitted between two organisations at one time is referred to as an interchange transmission. An interchange can contain a single message or many messages. The standard **Start of Transmission** segment is used to identify the start of the data being transmitted. The standard **End of Transmission** segment is used to denote the end of the data being transmitted.

b) The only restriction imposed by the syntax rules on the number of messages included in an interchange transmission is that the message count on the end of transmission segment cannot exceed 99,999. There are no restrictions on the number of different types of message which can be included in a transmission. However see paragraph 2.9 below.

c) The syntax rules do not themselves specify the order in which messages should be transmitted within an interchange transmission, but the standard messages are all grouped within TRADACOMS-designated **files** and within each file a message sequence is specified. The sender can forward a number of designated files, one after another, in an order of his choosing, unless participants in an exchange application agree otherwise. The conventional format of interchange transmissions is described in 2.9.

d) Participants can agree restrictions on the number of files or messages in a transmission if they so wish. See paragraph 2.9.

2.3 Data Separators and Terminators

2.3.1 Segment Terminator (')

Each segment is terminated by a segment terminator, the apostrophe (or single quote) (')

eg. ..... 123'

2.3.2 Data Element Separator (+)

a) Data elements within a segment are separated from each other by a data element separator, a plus (+)

eg. .....+ABC+123

b) The last data element in a segment is terminated by the segment terminator. A data element separator is not required.

2.3.3 Sub-Element Separator (:)

a) Sub-elements with a composite data element are separated from each other by a sub-element separator, the colon (:)

eg. .....+5530:KG+.....

b) The last sub-element within a composite data element is terminated by a data element separator. A sub-element separator is not required.

2.3.4 Segment Code Separator (=)

A segment code is separated from the data within a segment by a segment code separator, an equals sign (=)

eg. IRF=8728511+810320'

2.3.5 Allowable Sequences of Data and Separators/Terminators

The logic of the syntax rules allows only the sequences of data, separators and terminators indicated in the following table (Figure A.3).

Figure A.3

Item	Can only be followed by item(s) ticked							
	Segment Code	=	Data Element	+	Sub-Element	:	'	End of Transmission
Segment Code		/						
=			/	/	/	/		
Data Element				/			/	
+			/	/	/	/		
Sub-Element				/		/	/	
:					/	/		
'	/							/

2.4 Character Set and Related Subjects

2.4.1 Character Set for Interchange

In the TRADACOMS environment the character set for data is restricted to:

Blank Space		Ampersand	&	Asterisk	*
Open Bracket	(	Close Bracket	)	Comma	,
Hyphen	-	Full Stop	.	Solidus	/
Percent	%				

The numerals zero to 9.

The upper case alphabet A to Z.

The following characters are reserved for control purposes within the syntax rules and their use in data should be avoided wherever possible:

Apostrophe	'	Plus sign	+	Colon	:
Equals	=	Padding Character (Up-arrow/circumflex)	^		

2.4.2 Character Code

The coded representation of each allowable character for the various transmission media is discussed in Section E of Volume 1.

2.4.3 Release Character

The question mark, ?, is reserved for use within the syntax rules as a release character.

The release character is provided for use where separator or terminator characters (ie. =:'+) have to be transmitted as part of a data element or sub-element. Where a question mark is itself included in data it must also use the release character function.

It is strongly recommended that the five characters =:'+? be excluded from data elements and sub-elements wherever possible. It should be possible to avoid the use of the release character by replacing these characters in data by another character without any loss of meaning. For example, an apostrophe could be replaced by a space (O'REILLY would be transmitted as O REILLY).

The release character must be inserted in a transmission immediately preceding each occurrence in data of any of the five characters =:'+?. It signifies that the next single character is not to be interpreted as a data separator, terminator or release character. For example:

Data required 736 + 834 = 1570

Data transmitted 736? + 834? = 1570

The release character is not counted as part of the length of any data element or sub-element within which it is transmitted. Release characters can be inserted by the

computer so that data can be input and output without any special manual requirements, except where messages are manually structured or interpreted.

2.4.4 Scope of the Character Set

The full recommended character set including the special characters +=:' applies only to the **data** content of the transmission. Requirements for telecommunication protocols, media labels, etc., are outside the scope of this recommendation. They are dealt with in Section E of Volume 1 of this manual.

2.4.5 Character Representation

Character formats are recommended within these syntax rules. Binary, packed decimal or other forms of hardware/software-dependent representation must not be used for interchange as these features are not available, or are not dealt with in the same way, on all makes of computer.

2.4.6 Decimal Separator (.)

a) An implied decimal separator or point is used in the TRADACOMS system.

b) Thousands must not be indicated by any form of separator (such as commas).

2.4.7 Negative Numbers

a) A negative number is indicated by a leading minus sign.

eg. SEG=..... +-1250+.... is correct.

b) In many cases the need for negative numbers can be avoided through the use of data elements such as the Credit Line Indicator used in TRADACOMS invoice messages.

Notes: Provision should be made in translation tables for negative numbers using leading minus signs in messages where the totals could become negative if the credit values exceed the normal debit ones. For example, statement totals could be negative if credit adjustments exceeded new invoices.

2.5 Absence of Data

2.5.1 Mandatory and Conditional Data

a) Within a segment, data elements are specified as **mandatory** or **conditional**.

b) **Mandatory data elements** are those which must always be transmitted, eg. the message type on a standard message header segment or the invoice number and date in an invoice message.

c) **Conditional data elements** are those which need to be transmitted only in a particular set of circumstances, depending on the nature of the goods, the mode of transport, etc., eg. a special price indicator in an order message.

d) Segments must also be designated as mandatory or conditional within each message type.

e) A mandatory segment may consist entirely of conditional data elements. At least one of these must be transmitted for this type of segment.

f) Within a composite data element, individual sub-elements are also designated as mandatory or conditional and, similarly, if all are conditional within a mandatory data element, at least one sub-element must be transmitted.

2.5.2 Rules for Indicating the Absence of Data

a) Where no data exists for a conditional segment, the segment must not be transmitted.

b) Where no data is required for one or more data elements in a segment, which precede another data element for which data **is** required, the data element separator is used to indicate the absence of data:-

eg.

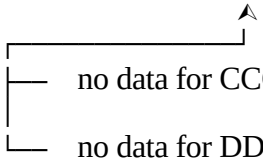
i) If there are 5 data elements, AAAA, BBBB, CCCC, DDDD and EEEE in a segment with the segment code SEG and data is only required for AAAA, BBBB, CCCC and EEEE:-

SEG = data for AAAA + data for BBBB + data for CCCC ++ data for EEEE'

no data for DDDD

ii) Similarly, if data is only required AAAA, BBBB and EEEE:

SEG = data for AAAA + data for BBBB +++ data for EEEE'



no data for CCCC

no data for DDDD

c) Where no data is required for one or more data elements at the **end** of a segment, two options are available:

i) The **preferred** option is to use the segment terminator to truncate the segment following the last data element for which data is required:

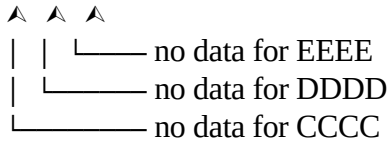
eg. Using the example shown in b) above, the following can be transmitted if data is only required for AAAA and BBBB:

SEG = data for AAAA + data for BBBB'

ii) Alternatively, data element separators can be used to indicate positively the absence of data for each data element.

eg. Using the above example, if data is only required for AAAA and BBBB:

SEG = data for AAAA + data for BBBB +++'



Note: Users' programs should cater for both these techniques, as each is valid.

d) The absence of data for one or more conditional sub-elements within a composite data element is indicated using similar principles to those described above. A data element separator is inserted following the last sub-element for which data is required. The absence of data for one or more sub-elements which precede another, for which data is required, is indicated by the sub-element separator.

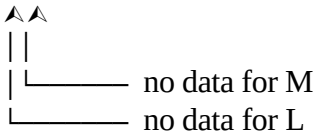
eg.

i) If a data element contains 4 sub-elements, K, L, M and N and only the first 2 are required:

....+ data for K: data for L +....

ii) In the same example, if data is only required for K and N:

....+ data for K:::data for N +....



e) A typical use of conditional sub-elements is for data elements defined as containing both a code and a clear language version, eg. transmission sender. The rules allow either or both the sub-elements to be transmitted in a consistent manner:

eg.

....+ 123:ABC CO+.....
....+ 123 +.....
....+ :ABC CO+.....

are valid ways of constructing a data element which can contain a code and/or a clear language sub-element.

2.5.3 Variable Length Data

a) Separators allow the omission of leading zeros in numeric items and trailing spaces in alphanumeric items for those data elements defined as "variable length" in the Data Directory.

For example:

i) RTL = ...+ ABC LTD'
RTL = ...+ YZ MANUFACTURING AND
PROCESSING LTD'

ii) OLD = ...+ 930:KG+.....
OLD = ...+ 12100:KG+.....

b) Where variable length data is interchanged, the sender may elect to transmit, in accordance with the syntax rules and without reference to the recipient, either the actual number of characters used or any number of characters up to and including the maximum number of characters. Where the maximum number of characters is transmitted, the following rules apply:

i) Numeric data should be right-aligned with leading zeros or spaces to pad the data to its maximum length, but there must be one zero before the decimal separator for amounts less than one.

ii) Alphabetic/alphanumeric data should be left-aligned with trailing spaces used to pad the data to its maximum length.

2.6 Transmission Structuring Segments

NB The normal rules for indicating the absence of conditional data, specified in subsection 2.5, apply to these standard segments.

2.6.1 A Transmission

a) The data transmitted at one time between two organisations is a transmission. Standard segments are provided to denote the start and the end of the **data** within a transmission.

b) The first segment in all transmissions must be the **Start of Transmission** segment which has the segment code STX.

c) The last segment in all transmissions must be the **End of Transmission** segment which has the segment code END.

d) A transmission can therefore be depicted as:-

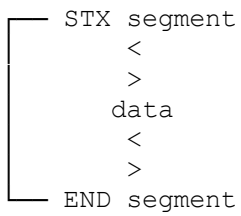


Figure A.4 Example STX Segment

STX=ANA:1+123:ABC CO+:XYZ CO+811206:121500+A143+B26+ASYS+A'

where	STX	is the segment code
	ANA:1	identifies version 1 of the TRADACOMS Syntax Rules
	123:ABC CO	identifies the sender of the message in code and plain language
	:XYZ CO	identifies the recipient of the messages in plain language only
	811206:121500	811206 is the transmission date 121500 is the time of transmission
	A143	is the sender's reference for the transmission
	B26	is the recipient's reference for the transmission, if known to the sender
	ASYS	is an application reference
	A	is a transmission priority code if required

Note:

1. The data element specifications for this and all other standard segments are contained in the TRADACOMS Data Element Directory and message specifications in this manual.

2. When using STX for transmissions across Value Added Networks, it is recommended that the user manual is consulted as each has used the elements in STX to achieve network addressing and control features. These are specific usages of the elements defined in this segment and will have to be respected if that service is used

2.6.2 Start of Transmission Segment (STX)

a) Mandatory information included in this segment identifies:

- the syntax rules being used
- the sender and recipient of the transmission
- the date of transmission
- the sender's reference for the transmission

b) Conditional information which can be included in the segment consists of:

- the time of transmission

- the recipient's reference for the transmission

- an application code (this would enable a recipient to identify, at the STX level, whether the transmission contained data for an import application, an export application, an accounting application, etc.). The application reference may also be used to indicate whether a transmission is 'live' or 'test' in status. In this case, the reference abbreviations given in Code Values List 16 should be used

- priority code

c) Figure A.4 contains an example of an STX segment in which all the conditional data is illustrated.

2.6.3 End of Transmission Segment (END)

a) This segment contains one data element, in which a count of the number of **messages** in a transmission must be provided.

b) The count includes all data messages in the transmission, to a maximum of 99,999 messages.

eg. END = 1' indicates a transmission consisting of 1 message
END = 25' indicates a transmission consisting of 25 messages

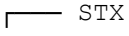
2.6.4 Interchange Messages

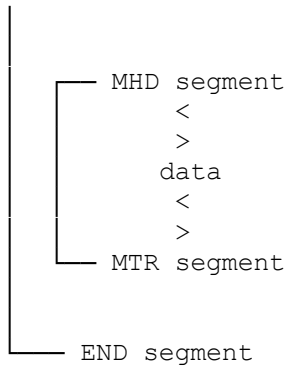
a) With the exception of the standard segments STX and END used to delimit a transmission, all data must be interchanged within a **message**.

b) All messages begin with a Standard **Message Header** segment which has the segment code MHD. All messages must end with the Standard **Message Trailer** segment which has the segment code MTR.

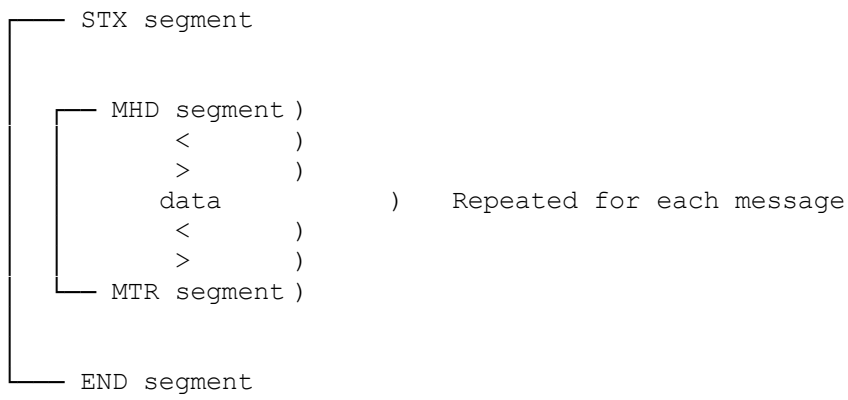
c) A transmission can consist of one message or any number of messages.

d) A transmission containing one message can therefore be depicted as:





e) A transmission containing more than one message can be depicted as:



2.6.5 Message Header Segment (MHD)

a) The MHD segment contains two mandatory data elements:

- **A Message Reference**

  This can be any unique reference allocated by the sender. In TRADACOMS messages this takes the form of a consecutive count of messages within a transmission.

- **A Message Type**

  This consists of two sub-elements, the message type itself and a number to distinguish the different versions of a message which can arise due to amendments to the message specification.

b) An example of an MHD segment is:

MHD = 12 + ORDERS:4'

This identifies a data message of type ORDERS, that contains data relating to the fourth version of the specification for ORDERS, and that it is the 12th message in the transmission.

2.6.6 Message Trailer Segment

a) This segment contains one mandatory data element, which is a count of the number of segments comprising the message.

b) The count includes both the MHD and MTR segments and data segments included in the message.

eg. MTR = 7' indicates that a message consists of 7 segments

2.6.7 Formats for the Standard Segments

The standard formats for segments MHD and MTR are illustrated in all the message formats in Volumes 2 and 3. The segments STX and END do not appear in messages. Their standard formats are shown in Figure A.5 and at the beginning of Volume 2.

2.7 Repeated and Nested Data

2.7.1 Introduction

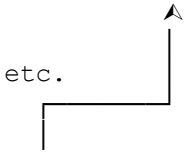
a) A common requirement of many types of message is the need to repeat data elements, or groups of data elements eg. an invoice could contain a number of items, each item containing a product number, quantity, price, etc. A particular data element must not be repeated within a data segment, but a data segment can be repeated within a message.

b) Sometimes a data element or group of data elements may repeat within an already repeating group, for example an invoice can cover goods supplied under several orders, and there may be many invoice item lines to each order. The data elements for an invoice line item form what is called a 'nested' repeating group. The data elements for a line item are grouped together in one repeating data segment while the details for each order are grouped in another repeating segment. The repeating segments are numbered and sequenced according to the following rules:

2.7.2 Repeating Segments

a) A **sequence number** data element is specified as part of a repeating data segment. The sequence number starts at 1 for the first transmitted repeating segment and is increased by 1 for each subsequent occurrence of this segment within the message.

eg.
```
ODD = 1 + .....'
ODD = 2 + .....'
ODD = 3 + .....'
```



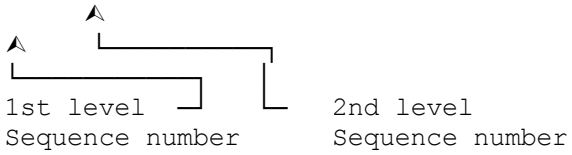
order sequence number/count within invoice message

b) The sequence number reverts to one at the start of each message.

2.7.3 Nested Segments

a) The principle of using sequence numbers is extended so that there is one sequence number for each level of nesting. In this case, however, the sequence numbers must indicate the hierarchical order of the data segments:-

eg.
```
ODD = 1 + .....'      1st level (highest or `parent' level)
ILD = 1 + 1 + .....'  2nd level (first invoice line sequence in first order)
ILD = 1 + 2 + .....'
etc.
ODD = 2 + .....'      1st level
ILD = 2 + 1 + .....'  2nd level
ILD = 2 + 2 + .....'
```



b) Where necessary, the technique can be extended to cover more complicated nesting situations:-

eg.
```
AAA = 1 + .........'
AAB = 1 + 1 + .........'
AAB = 1 + 2 + .........'
AAC = 1 + 2 + 1 + .........'
AAC = 1 + 2 + 2 + .........'
AAD = 1 + 2 + 2 + 1 + .........'
AAC = 1 + 2 + 3 + .........'
AAB = 1 + 3 + .........'
AAC = 1 + 3 + 1 + .........'
AAB = 1 + 4 + .........'
AAA = 2 + .........'
```

2.7.4 Sequence Numbers

a) On nested repeated segments the highest level sequence number must come first, followed by the next highest and so on.

b) Segments must be transmitted in the hierarchical sequence indicated in the above examples.

2.8 Batching of Messages

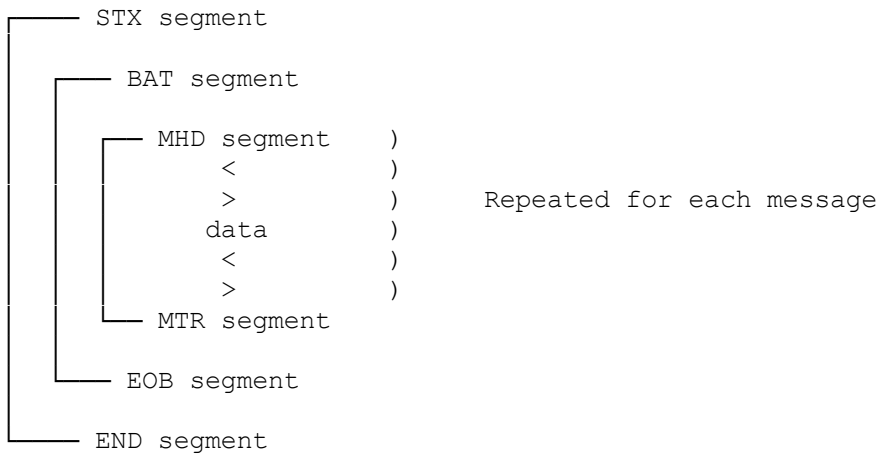
2.8.1 Batching of Messages within a Transmission (BAT and EOB)

a) A transmission can be divided into a number of batches if such a facility is required in a particular application. Batching is a **conditional** feature of the syntax rules as it will not be a requirement of all interchanges.

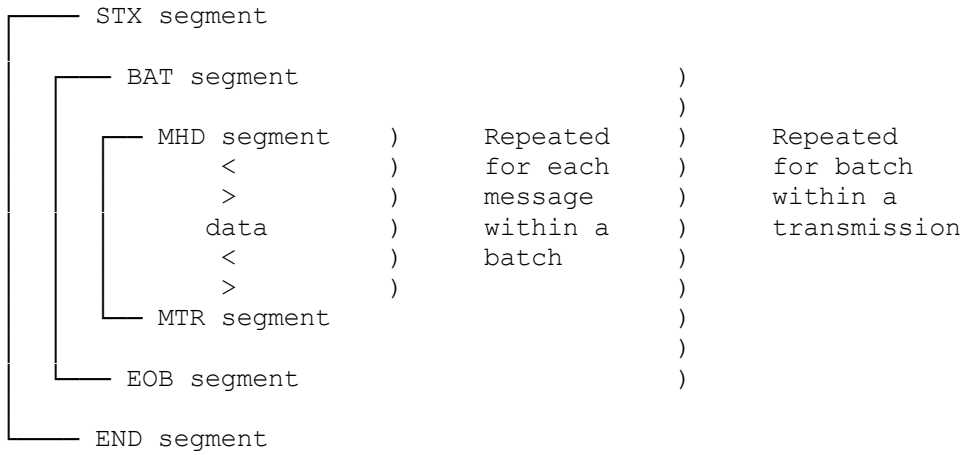
b) The facility must only be used, therefore, where the participants in an interchange agree this in advance.

c) A batch of data within a transmission begins with a **Batch Header** segment which has the segment code BAT. A Batch is terminated by a **Batch Trailer** segment which has the segment code EOB.

d) A transmission consisting of a single batch of messages can be depicted as:-



e) A transmission consisting of more than 1 batch of messages can be depicted as:-



2.8.2 Batch Header Segment (BAT)

a) This contains a reference allocated by the sender to the batch.

eg. BAT = 77421' identifies a batch with the reference 77421

2.8.3 Batch Trailer Segment (EOB)

a) This contains a count of the number of messages in a batch, including any control messages.

eg. EOB = 29' indicates that a batch consists of 29 messages

Figure A.5

START OF TRANSMISSION SEGMENT
END OF TRANSMISSION SEGMENT

SEGMENT			DATA ELEMENT		DATA ELEMENT NAME	M/C	F/V	PICTURE	REMARKS (See also General Remarks in Directory)
STX	=				START OF TRANSMISSION	M			
			STDS		Syntax Rules Identifier	M			
					Identifier	M	V	X(4)	Value = ANA, may be ANAA if reconciliation facility used
				:	Version	M	F	9	Value = 1
		+	FROM		Identification of Transmission Sender	M			Code or name transmitted
					Code	C	V	X(14)	Codes can be agreed by interchange parties
				:	Name	C	V	X(35)	Mandatory if no code used
		+	UNTO		Identification of Transmission Recipient	M			Code or name transmitted
					Code	C	V	X(14)	Codes can be agreed by interchange parties
				:	Name	C	V	X(35)	Mandatory if no code used
		+	TRDT		Date and Time of Transmission	M			
					Date	M	F	9(6)	Format: YYMMDD
				:	Time	C	F	9(6)	Format: HHMMSS (if required by the particular application)
		+	SNRF		Sender's Transmission Reference	M	V	X(14)	Reference for the transmission
		+	RCRF		Recipient's Transmission Reference	C	V	X(14)	If known to sender
		+	APRF		Application Reference	C	V	X(14)	Used by networks to validate transmission: does this sender have authority to send this type of message to this recipient?
		+	PRCD		Transmission Priority Code	C	F	X(1)	Code Values List 1
END	=				END OF TRANSMISSION	M			
			NMST		Number of Messages in Transmission	M	V	9(5)	Control count of the number of messages in a tape or transmission

NOTE: DETAILS OF THE WAY IN WHICH STX IS USED FOR TRANSMISSIONS VIA VALUE ADDED NETWORKS ARE GIVEN IN THE NETWORK USER MANUAL.
ANA recommends the use of EAN Location Numbers to identify sender and recipient.

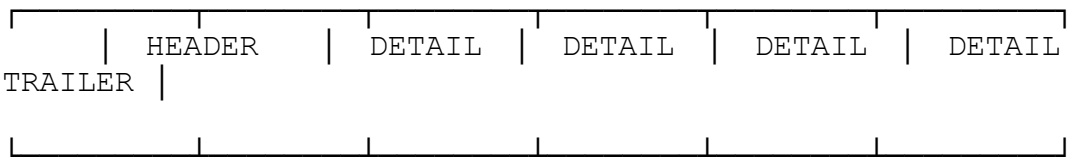
2.9 TRADACOMS CONVENTIONS

The UNGTDI syntax rules do not restrict the number and type of messages and files which may be exchanged. However to simplify processing, particularly for the recipient, the following TRADACOMS conventions have been established.

When other conventions are to be used in EDI communications, these need to be agreed with trading partners, to ensure that the combinations of files and messages can be processed correctly.

2.9.1 Single File Interchanges

A TRADACOMS File normally consists of a Header Message, one or more Detail Messages, followed by a Trailer Message. This convention is illustrated in the diagram below. For those messages which contain VAT information there is also a VAT Trailer Message before the File Trailer Message.



When several orders are being transmitted in one interchange to a supplier it is the usual practice to send all the Order Detail Messages within the same Header/Trailer Messages, rather than repeating the Header and Trailer Messages for each Detail Message. This improves the efficiency of the transmission.

It is recommended that there is one file for one recipient sent in each interchange transmission. This permits control of the files transmitted and end to end audit trails (see Volume 1 Section A Paragraph 8 Controls).

It is also recommended that there is one type of file in each interchange transmission as this allows recipients to sort and select files for different applications using the application reference in STX.

3. THE CONTEXT OF THE SYNTAX RULES

3.1 Applicability of the Syntax Rules to Various Interchange Environments

3.1.1 Introduction

In the following paragraphs the applicability of the rules to various types of interchange environments is discussed:-

- computer (or intelligent terminal) to computer (or intelligent terminal)

- computer to unintelligent terminal (including facsimile and telex)

- unintelligent terminal to computer

For the purpose of this discussion:-

- a computer is any equipment used to construct messages in accordance with the syntax rules (whether data is input manually or held on in-house computer records), or any equipment used to translate messages structured in accordance with the rules to meet in-house computer record or print layout requirements. for instance, it can add separators, terminators, segment codes etc., to data at the sender's end of a transmission and remove them at the recipient's end.

- an unintelligent terminal is any equipment which cannot be, or is not, used to construct and translate messages. It includes facsimile, telex and message switching systems where data has to be manually prepared or interpreted.

3.1.2 Computer (or Intelligent Terminal) to Computer (or Intelligent Terminal)

This includes the interchange of data over telecommunications links and by a physical medium, such as magnetic tape, where messages have been constructed by the sender's computer and are translated by the recipient's computer.

The syntax rules are primarily intended for this form of interchange. They act as a bridge between the internal requirements of the various interchange partners. Methods of data capture by senders of messages and methods of output of data by recipients are outside the scope of the syntax rules in this environment.

In this environment, errors caused by missing or wrongly used separators, or by other incorrect use of the syntax rules, are likely to be minimal once trials have been completed. It is strongly recommended that interchange partners carry out trials prior to implementing an interchange system, not only to ensure that data has been correctly formatted but also to ensure that the computers are compatible as far as the interchange medium is concerned.

Unless economy of transmission is vital, errors should be corrected by re-transmitting complete messages. This is the simplest form of correction. This subject is treated in detail in paragraph 9 of Section C in Volume 1.

3.1.3 Carriage Return and Line Feed Characters

These will not normally be included in computer-to-computer environments but it may not be possible to assume that such a transmission will not contain these characters, eg. the sender of the transmission may receive data which included these characters from another organisation and pass the data to the recipient without editing the data. In interchange systems where this could arise, participants could agree that senders of messages would remove all such characters prior to computer-to-computer interchange.

3.1.4 Computer to Unintelligent Terminal (Including Facsimile and Telex)

The TRADACOMS system is based on computer-computer transfers of data. When an unintelligent terminal is used to print or display messages the recipient will need a copy of the message specification.

Intelligibility is aided by the use of segment codes and the graphic separators = + :' but messages in syntax rules format are not designed primarily for human interpretation.

Carriage Return and Line Feed characters would need to be transmitted on telex. These are not separator characters and can be used freely to improve the readability of messages, eg. after each line of a name and address.

3.1.5 Unintelligent Terminal to Computer

The main difference between data prepared by computers and data which is manually prepared is the likely increase in the incidence of errors, as neither data nor formats are validated by computer prior to interchange.

Recipients should be tolerant of 'extra' spaces if any data might be manually prepared.

eg. MHD = 12 + ORDHDR :3'

Data input manually to the telex network or a message switching system will contain fewer errors if input forms specially designed for each message are used.

3.2 Advantages of the Syntax Rules over Conventional Computer Records for Interchange Systems

The conventional layouts in most in-house systems, in which each data element begins in a fixed position within the character string comprising a record, have a number of drawbacks for **interchange systems**. These paragraphs illustrate how

the syntax rules overcome these drawbacks, using progressive changes to conventional record layouts.

3.2.1 Drawbacks of Conventional Layouts for Interchange Purposes

Intelligibility

Conventional records are rarely directly intelligible to people without considerable computer processing experience. There are no segment codes or data element separators to aid the identification of each data element and, unless special arrangements are made to include a version number in a message, recipients cannot be certain that the necessary amendments have been made by senders.

Media Space

Utilisation of media space may be poor, as data elements are usually padded to their maximum length with leading zeros or trailing spaces. While this may not be critical on magnetic tape, it could be critical over telecommunication links.

Inflexibility

There is no data independence in conventional layouts. Any change to a data element length affects the position of all subsequent data elements within a message. A fully detailed record layout needs to be agreed by all interchange partners even though some individual participants may not require a particular data element. This makes maintenance difficult.

Machine Dependence

Different makes of computers may set varying constraints on logical records and block sizes. Conventions for variable length records on magnetic tape may differ.

3.2.2 How Syntax Rules Overcome these Disadvantages

(a) **Conventional Layout**

Figure A.6 illustrates a conventional record layout which might be used for a message based on part of an order. Each element begins in a fixed position and is padded to its maximum length. In this example six elements/sub-elements are not required, leaving spaces in the record layout. Only part of one order line is shown; a full order could contain over a thousand lines.

Figure A.6 Conventional Layout for Part of an Order Record

Record Layout	**Elements/Sub-elements**
810303 810313 USE GATE NO.6 0 6 12	Earliest delivery date, Latest delivery date, Delivery instructions narrative.
1 5012345678900 52 56 69	Order line sequence no., Supplier's EAN Trade Unit Code
12345A 5012345101019 77 90	Supplier's item no./code, Supplier's EAN consumer unit code, Retailer's own Brand no. (blank)
12 105 113 117	Retailer's item no. (blank), Consumer units in trade unit, Ordering measure (blank)
1000 125 131 136 144 150	Measure indicator (blank), No. of trade units ordered, Total measure ordered (blank), Measure indicator (blank), etc.

(b) **Segments**

(i) The first improvement that the Syntax Rules make to this conventional layout is to divide one record for the whole order into shorter sub-records or, as they are called in the standards, **segments**.

For example the delivery dates and instructions could be included in a segment called DIN. The order line details could form part of a segment called OLD. (Figure A.7.)

Figure A.7

```
 DIN      810303      810313      USE GATE NO.6
 0    3          9           15


 OLD      1      5012345678900      12345A      etc
 0    3      7                 20          28
```

(ii) This technique gives three advantages:-

- Better use can be made of media space. If there is no data in an entire segment, then it can be omitted from the interchange message.

- Segment codes make a message more intelligible.

- It is more flexible, as additional segments can be inserted without difficulty.

(iii) However, there are some disadvantages:-

- A data element which is blank in a segment containing some non-blank data elements still takes up media space eg. allowing for orders by trade unit or measure in the OLD segment.

- Data elements are identified solely by their position within the segment. A change in the length of one element would necessitate changing the position of all subsequent elements in the segment. Data independence is therefore limited.

- Another disadvantage is that each segment would need to start on a computer record boundary, as there is no other easy way of distinguishing the segment code. To complicate matters, each type of segment could have a different length and some computers cannot cope with variable length records: alternatively all segments would need to be padded out to a common length which would be wasteful of media space.

(c) **Separator Characters**

(i) The above difficulties are overcome in the Syntax Rules by the inclusion of special characters to separate one data element from another, to terminate a segment, to separate a segment code from the data and to split a composite data element, where required, into its component parts.

Figure A.8 contains the same data as shown in the conventional record layout in Figure A.6 but formatted according to the Syntax Rules.

Figure A.8 Syntax Rules Layout for Part of an Order Record

```
DIN=810303+810313++USE GATE NO.6'
OLD=1+5012345678900:12345A+5012345101019++12+1000+.....
```

In Figure A.8:-

- A plus (+) is used to terminate each data element.

- An apostrophe (') - or single quote - is used to terminate a segment.

- A colon (:) is used to separate one sub-element from another.

- An equals sign (=) is used to separate a segment code from the data.

- A plus (++) is not used to terminate the last data element of a segment as the apostrophe fulfils the purpose.

(ii) Padding can be omitted by incorporating separator characters. This improves the use of media space. Figure A.8 uses 83 characters against the 150 used in Figure A.6.

(iii) Separators help to make the message more intelligible.

(iv) Data element lengths can be altered without affecting the identification of subsequent data elements.

(v) Segments need not begin on a computer record boundary as the use of separators clearly distinguishes the segment code. Message and segment lengths, therefore, need not be dictated by such constraints as input/output block sizes, particular conventions on the use of variable length records, the physical characteristics of punched cards, telecommunications, protocols etc. Anyone interchanging data on a variety of media or between machines using different conventions needs to be free of such restraints.

3.2.3 Absence of Data

In many messages some of the data will be mandatory (ie. it must always be present) and some will be conditional (ie. it might or might not be present depending upon the sender, mode of transport, type of goods etc.). The absence of data can be indicated efficiently in three ways using the syntax rules:-

- Where no data need be transmitted for an entire segment, the segment need not be transmitted at all (ie. where all the data elements are conditional).

- Where one or more data elements at the end of a segment are absent, the segment may be truncated, eg. if there are five data elements in a segment and data is only required for the first two A and B, this can be represented by:-

  segment code = data for A + data for B'

- Where no data is required for an element which precedes another element for which data is required in a segment, the absence of data can be indicated by the data element separator character. For example, if there

is no data for element B in the sequence of elements A, B, C this can be represented by:-

segment code = data for A ++ data for C'

3.2.4 Conclusion

One point should be re-emphasised - it is not proposed that conventional, in-house record layouts or databases as used by application programs should be replaced by ones structured according to the syntax rules. The rules are used to structure messages at the point of interchange as a "bridge" between differing systems. The sender converts from his internal record layouts and the recipient converts into his internal record layouts. The fact that the sender may have conventional files and the recipient a database will not affect the success of the syntax rules as a bridge for the transfer of data between the two systems. The syntax rules are also being used to transfer data between different parts of the same company.