

- پایین آوردن چشمگیر هزینه های عملیاتی
- انجام خرید و فروش (واردات و صادرات)، انجام مناقصه های بین المللی، کنترل موجودی ها، امور بانکی و پرداخت ها، انجام فعالیتهای گمرکی و غیره به گونه ای آسانتر، سریعتر و دقیقتر
- بالا رفتن سطح تولید و برگشت سریع سرمایه ها (توسعه صادرات)
- ایجاد هماهنگی و تطبیق شرایط کاری با کشورهای دیگر به ویژه از دیدگاه موضوع رقابت در بازار جهانی و همچنین پیوستن به قطب های اقتصادی
- به حداقل رساندن اشتباهات انسانی و بوجود آوردن امکانات بهتر برای کنترل و برنامه ریزی
- ورود تجارتهای کوچک و متوسط به صحنه تجارت بین المللی، ایجاد اشتغال، فقر زدایی و انتقال فناوریهای نوین

هزینه های استفاده از EDI در تجارت بین المللی

- هزینه های مربوط به آموزش و آگاه سازی عمومی و ایجاد فضای قانونی
- هزینه های مربوط به استقرار سخت افزارها که معمولا در هر خودکارسازی وجود دارد و هزینه های مطالعاتی برای برقراری EDI به شیوه هایی در پروژه های مانیزه وسیع عمل میشود.
- هزینه های مربوط به نرم افزارهای مورد نیاز
- هزینه های مربوط به بستر سازی های سیستمهای ارتباطی و مخابراتی مورد نیاز
- هزینه های مربوط به تجدید ساختار تجاری و EDI در بخشهای مرتبط با تجارت خارجی مانند پروژه SWIFT در بانک مرکزی و ASYCUDA در گمرک ایران.
- هزینه های مربوط به ایجاد شبکه های اطلاعات بازرگانی
- هزینه های عملیاتی و بهم وصل شدن که شامل هزینه های مربوط به توسعه سیستمها و نرم افزارهای جدیدتر نیز میشود.

نتایج حاصل از استفاده از EDI و سایر زیر مجموعه های آن

نتایج و مزایای کلی راهبردی

- شامل کاهش هزینه های اداری و اجرایی، ارتقای ارزش افزوده از طریق افزایش سرعت و دقت در پردازش عملیات، بهبود کنترل موجودی، یکپارچگی استراتژیک داده های حاصل از EDI، پردازش اطلاعات و مزیت بازاریابی نسبت به رقبایی که در کاربر EDI نیستند.

نتایج و مزایای عملیاتی و فنی

- بهبود ارائه خدمات به مشتریان، حذف کاغذ نمابر و پیامهای پستی بدون الگو، کاهش زمان کلی پردازش معاملات، حذف پردازشهای تکراری داده ها، بهبود مدیریت نقدینگی، حذف پایانه های کامپیوتری وقت گیر، حذف بایگانی دستی، بهبود نظارت بر هزینه های عملیاتی، حذف نیاز به انطباق اسناد و نسخه برداری و پرونده سازی، حذف هزینه های پستی، طراحی و چاپ فرمهای اداری.

EDIFACT

EDIFACT is an acronym for EDI For Administration, Commerce and Transport.

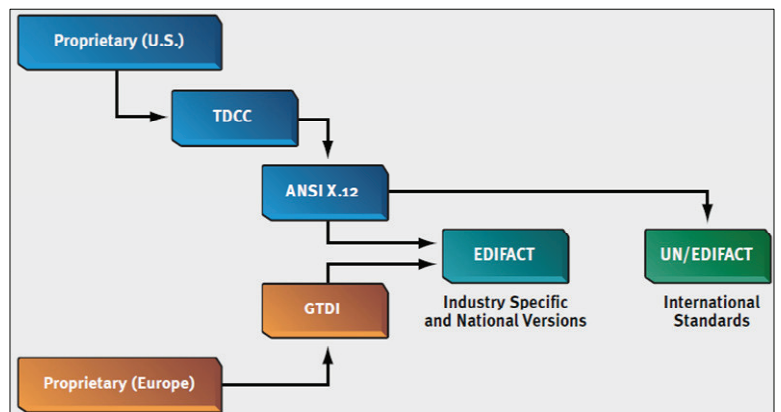
EDI standards facilitate electronic data interchange (EDI) by providing:

- Rules of syntax
- Definition of the data organization
- Editing rules and conventions
- Published public documentation

- EDIFACT
- an international EDI standard
 - a set of syntax rules
 - data elements, segments and codes
 - messages

- EDI standards
- Allow an 'open' system
 - Reduce implementation effort
 - Provide 'third-party interfaces'

EDIFACT is the product of the evolution in bringing the Proprietary Standards of the US and Europe together to form a single international EDI standard.



Message Definition

A **message** is a single business document. Each message is identified by a six character name.

From the buyer-side these include:

- Buyer
- ORDERS—Purchase Orders
 - CUSDEC—Customs Declaration
 - IFTMIN—Instruction Message
 - REMADV—Remittance Advice
 - PAYORD—Payment Order

Seller-side messages include:

- Seller
- IFTMAN—Arrival Notice
 - CUSRES—Custom Response
 - INVOIC—Invoices

Messages are made up of a collection of sequenced **segments** within defined **areas**.
 Some segments may be used in more than one area.
 The segments that can be used in each area are defined by the EDIFACT documentation.
 EDIFACT provides a hierarchical structure for messages.

Area—Areas contain groups of segments that perform specific functions. The areas, in turn, make up a message. To be complete, a message must contain a header area, detail area and summary area.

Segment—A part of an EDI message or transaction set, made up of a number of logically-related data elements in a fixed, defined, sequence, separated by a delimiter, conveying a part of the business transaction being made.

UNH+...	Header
BGM+...	
MOA+...	
...	
DOC+...	Detail
...	
UNS+S'	Summary
...	
MOA+...	
UNT+...	

Messages begin with the Message Header (UNH) Segment
 Messages end with the Message Trailer (UNT) Segment.

These two segments are the first, and innermost, level of the three levels of "electronic envelopes" within EDIFACT. Here is an example of an Extended Payment Order (PAYEXT) message that illustrates this structure

Message Structure: Segment Tables

Here is an example of a segment table for the Extended Payment Order (PAYEXT)

Position	Tag	Name	Req	Rept
0010	UNH	Message Header	M	1
0020	BGM	Beginning of Message	M	1
0030	BUS	Business Function	C	1
0040	DTM	Date/Time/Period	M	4
0060	RFF	Reference	M	1
0070	DTM	Date/Time/Period	C	1
0080	FTX	Free Text	C	5
0090	PAI	Payment Instructions	C	1
0100	FCA	Financial Charges Allocation	C	1
0120	MOA	Monetary Amount	M	1
0130	CUX	Currencies	C	1
0140	DTM	Date/Time/Period	C	2
0150	RFF	Reference	C	1
	etc.			
	etc.			

The message structure is defined in **segment tables**. These give the 'rules' of the message. They also show which segments are used in a particular message and the order in which the segments must appear.

Segment Tables—The message structure is defined in a listing known as a 'segment table'. These give the 'rules' of the message.

Requirements Designator—Usage indicator of segment detail: Mandatory, Optional or Floating.

Repetition Field—Entry in a segment table which states how many times a particular segment may repeat.

Mandatory—Attribute that indicates that the data element must contain data. **Nested Segment Group**—Segment groups that reside within other segment groups.

Conditional—Attribute that indicates that the data element may or may not contain data, depending on the message.

Message Structure: Segment Groups

When collections of segments repeat as a group, they are called **segment groups**.

Segment groups may be '**nested**'. This means that a segment group is fully contained within another segment group.

Here is an example of segment groups for the Extended Payment Order (PAYEXT)

Position	Tag	Name	Req	Rept
0010	UNH	Message Header	M	1
0020	BGM	Beginning of Message	M	1
0030	BUS	Business Function	C	1
0040	DTM	Date/Time/Period	M	4
0050		Segment Group 1	C	2
0060	RFF	Reference	M	1
0070	DTM	Date/Time/Period	C	1
0080	FTX	Free Text	C	5
0090	PAI	Payment Instructions	C	1
0100	FCA	Financial Charges Allocation	C	1
0110		Segment Group 2	M	1
0120	MOA	Monetary Amount	M	1
0130	CUX	Currencies	C	1
0140	DTM	Date/Time/Period	C	2
0150	RFF	Reference	C	1
	etc.			
	etc.			

Here is an example of a **Nested Segment Group**

Position	Tag	Name	Req	Rept
0280		Segment Group 6	C	9999
0290	DOC	Beginning of Message	M	1
0310	DTM	Date/Time/Period	C	5
0320	RFF	Reference	C	5
0330	NAD	Name and Address	C	2
0340		Segment Group 7	C	5
0350	CUX	Currencies	M	1
0360	DTM	Date/Time/Period	C	1
0370		Segment Group 8	C	100
0380	AJT	Adjustment Details	M	1
0390	MOA	Monetary Amount	C	1
0400	RFF	Reference	C	1
	etc.			
	etc.			

Segment

A **segment** is a collection of logically-related **data elements** in a fixed, defined sequence.

Segment contain [

- A three-character alphanumeric code that identifies the segment. This is called the **segment tag**.
- **Variable length data elements**. These can be either **simple** or **composite**.

Data Element—One or more data items, forming a unit or piece of information as defined in the data dictionary of a system of EDI Standards, and contained in an EDI message or transaction set. The term “data element” is often abbreviated as “DE” followed immediately by the data element number (i.e., data element 128 would be abbreviated as DE128) in some texts.

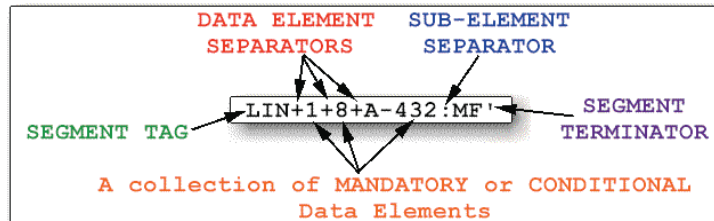
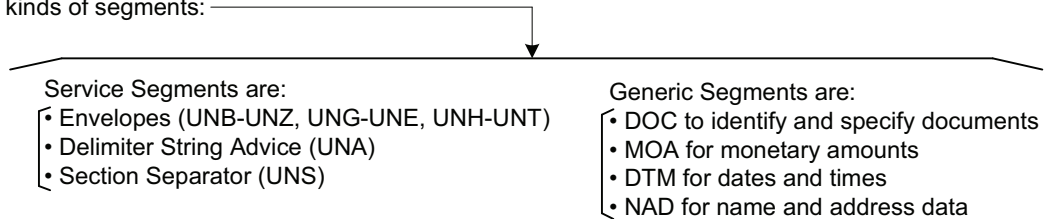
Segment Tag—A three character alphanumeric code that identifies the segment.

Variable Length Data Element—Data element whose length changes depending on the number of characters in the actual data. Minimum and maximum size of the element is given in the element table.

Segments must be separated by a data element separator (data element delimiter), which is normally **+** and **:**, and terminated by a segment terminator, normally **'**.

In EDIFACT, there are two kinds of segments:

- **Service Segments**
- **Generic Segments**



Segment Terminators and Delimiters

Indicates no more data

`NAD+BY+123456789::16'`

The end of each segment is determined by the Data Segment Terminator. In EDIFACT the standard data segment terminator is **'**.

`NAD+BY+123456789::16'`

Indicates optional data not used

Optional or conditional data elements that are not used must be accounted for by their position

`NAD+BY+123456789::16++++++'`

Not Correct

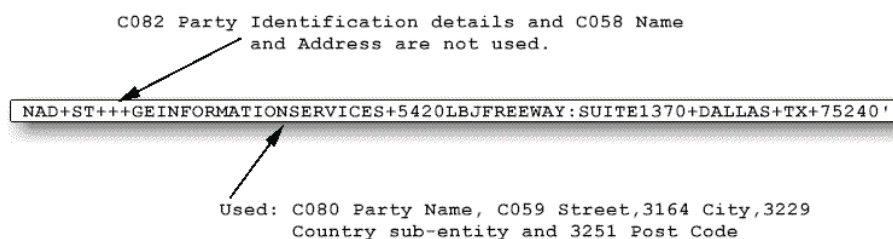
optional or conditional data elements without data that appear at the end of a data segment do not need additional data element separators to correctly position the data.

Mapping—The solution was to standardize the data that was read by a computer program so that the data could be read by all programs with that standard.

Data Elements: Simple and Composition

A **simple** data element contains one piece of information.

The **composite** data element contains more than one piece of information, usually containing qualifiers.



Numeric :A numeric segment may contain only digits, a decimal point and, if negative, a minus sign.

Attributes	Sample Data
n..4	Permitted: 1 123 1234 -1234 1.1 1.234 Not Permitted: A12 12345
n4	Permitted: 1234 -1234 Not Permitted: 1 123 A12 -12345
n8	Permitted 20000214 (dates are numerics)

If the numeric is given as a decimal, the number must have a digit before and after the decimal point. For example: 2.0 is correct (as is 2), however, 2. is wrong. 0.50 is correct (as is 0.5), .50 is wrong.

Alphabetic An alphabetic segment contains the specified number of alpha characters, including embedded blanks. Leading spaces must be preserved.

Attributes	Sample Data
a..8	Permitted: ABCD TOM ABC ALSMITH Not Permitted: DONGREENWOOD
a4	Permitted: ABCD WXYZ Not Permitted: A A123 BCDEF

Alphanumeric Alphanumeric segments contain the specified number of alphanumeric characters (including embedded blanks). Leading spaces must be preserved.

Attributes	Sample Data
an..8	Permitted: 12345 TOM A12 ALSMITH Not Permitted: DONGREENWOOD
an4	Permitted: A123 12B3 Not Permitted: A A12 1234 ABCDE

Different types of data elements also have specific rules they must follow. The data element dictionary usually specifies the codes (identifiers) by using the words 'coded' or 'qualifier' in the data element name:

6345	Currency, coded	C	an..3
6343	Currency qualifier	C	an..3

Composite Data Elements: Qualifier and Value

In EDIFACT, the composite data element is made up of two or more pieces of data (known as **components**) which form a single data unit. Typically the first data element is the **value**, which is being qualified. The second data element is typically the **qualifier**. These are typically ID (code values) fields. The qualifier gives additional definition to the value.

Here is an example of a composite data element. This data element is in regard to financial institution information. This is the information provided in the segment detail:

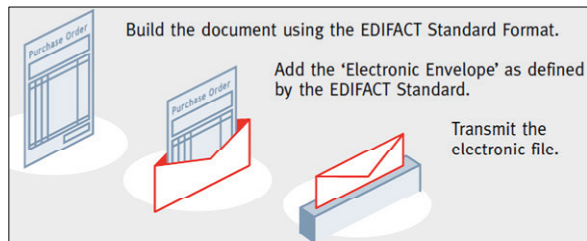
3035	Party Qualifier	M	an..3
C078	Account Identification	C	
3194	Account holder number	C	an..17
3192	Account holder name	C	an..35
3192	Account holder name	C	an..35
6345	Currency, coded	C	an..3
C088	Institution Identification	C	
3433	Institution name identification	C	an..11
1131	Code list qualifier	C	an..3
3055	Code list responsible agency, coded	C	an..3
3434	Institution branch number	C	an..17
1131	Code list qualifier	C	an..3
3055	Code list responsible agency, coded	C	an..3
3432	Institution name	C	an..70
3436	Institution branch place	C	an..17
3207	Country, coded	C	an..3

This is how the Party Qualifier data element (3035) is displayed in the message

FII+BK+2160644555:W.G.CAFIERO+111902039:25:19:::NATIONS BANK: RICHARDSON, TX'											
DE3194		DE3192		DE3433		DE1131		DE3432		DE3436	
C078				C088							

The composite data elements (C078 and C088) are made up of various conditional components from the segment table. Because they are conditional, not all of the data elements are used. All components are separated by a sub-element qualifier (:).

Message Structure and Electronic Enveloping and Architecture



Levels and Character Sets

In EDIFACT there are two levels in which messages may be transmitted. The use of a particular level designates which character set will be used:

- LEVEL A (UNA): only upper case; only printable characters
- LEVEL B (UNB): upper and lower case; includes non-printing characters for delimiters

The UNA Interchange is transmitted as a single string of 9 characters prior to the UNB Interchange segment.

UNA is optional, and if not used, the defaults shown below apply:

A release character is used when there is a need to transmit a message in which a character is designated as a delimiter, indicator or terminator, but it also represented in the message data.

	LEVEL A	LEVEL B	
Character 1	:	IS1	Sub-element delimiter
Character 2	+	IS3	Data element delimiter
Character 3	. or ,	Same	Decimal point indicator
Character 4	?	Not Used	Release character (space if not used)
Character 5	Reserved for future use	Space	Space
Character 6	'	IS4	Segment terminator

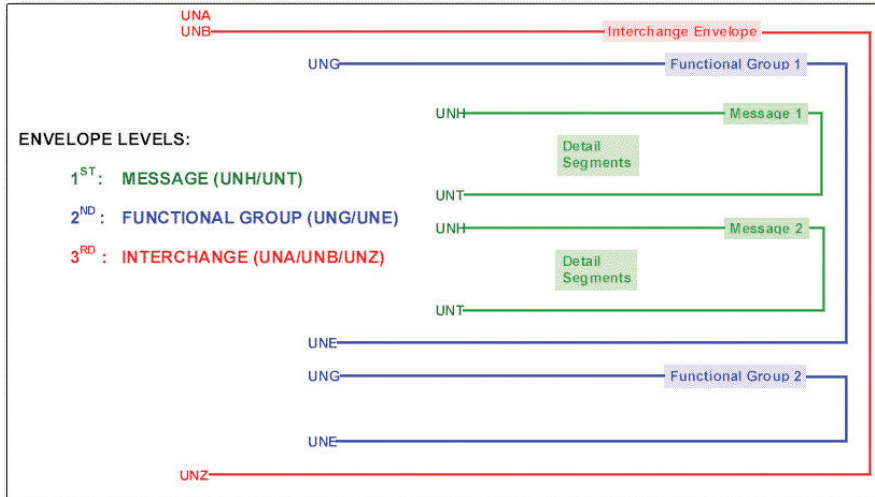
Electronic Enveloping

EDIFACT has two required levels of envelopes:

- **Interchange (UNB/UNZ):** a set from one sender's mailbox address to another sender's mailbox address
- **Message (UNH/UNT):** the envelope around one particular message

In addition, there is one optional envelope level: **Functional Group (UNG/UNE)**. It is used to group like messages together and for sub-addressing within an organization. In the US ANSI X.12 standards, this group level is where the message format and version are specified. Use of the UNG/ UNE is mandatory to/from North America

The following diagram illustrates Electronic Enveloping



The UNH segment has four data elements:

- **Message Reference Number (M):** assigned by the sender's computer and is part of the CONTROL mechanism.
- **Message Identifier (M):** Composite which identifies the message being enveloped using the six character message code (e.g., PAYEXT, REMADV) and the Version/Release data.
- **Common Access Reference Number (C):** Relates multiple transactions together.
- **Status of the Transfer (C):** Sequences a series of related messages.

example of how the CONTROL mechanism in the UNH element is used to validate message data:

Suppose this message was sent:

```

UNH+1001+ORDERS :D: 94B: UN'
BGM+105+12345+... '
DTM+4: 931102:101'
LIN+1+... '
LIN+2+... '
LIN+3+... '
UNS+S'
CNT+2: 3'
UNT+9+1001'
UNH+1002+ORDERS :D: 94B: UN'
BGM+105+12346+... '
DTM+4: 931102:101'
LIN+1+... '
LIN+2+... '
UNS+S: 2'
CNT+2: 2'
UNT+8+1002'
                
```

But this part was 'lost' in the process

And this is what was received:

```

UNH+1001+ORDERS :D: 94B: UN'
BGM+105+12345+... '
DTM+4: 931102:101'
LIN+1+... '
LIN+2+... '
UNS+S: 2'
CNT+2: 2'
UNT+8+1002'
                
```

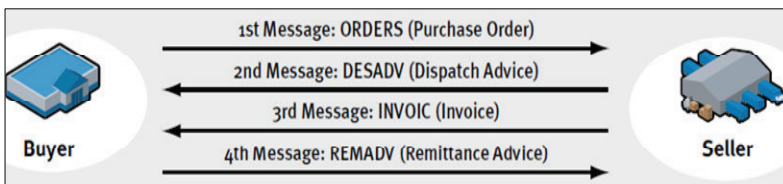
Even though the COUNTS are correct, a CONTROL NUMBER match will catch the error

The UNT segment has two data elements:

- **Number of Segments in a Message (M):** count of segments in the message (including UNH and UNT).

- **Message Reference Number (M):** same one that was used in the UNH for that particular message.

The **Common Access Reference Number** is used to identify a series of related EDIFACT messages. For example, one purchase may involve a message exchange that requires four messages to accomplish the complete business transaction as given here:



- For Message #1: UNH+2348+ORDERS:D:94B:UN+10381+1:C'
- For Message #2: UNH+156009+DESADV:D:94B:UN+10381+2'
- For Message #3: UNH+156078+INVOIC:D:94B:UN+10381+3'
- For Message #4: UNH+2451+REMAADV:D:94B:UN+10381+4:F'

The Functional Group Envelope

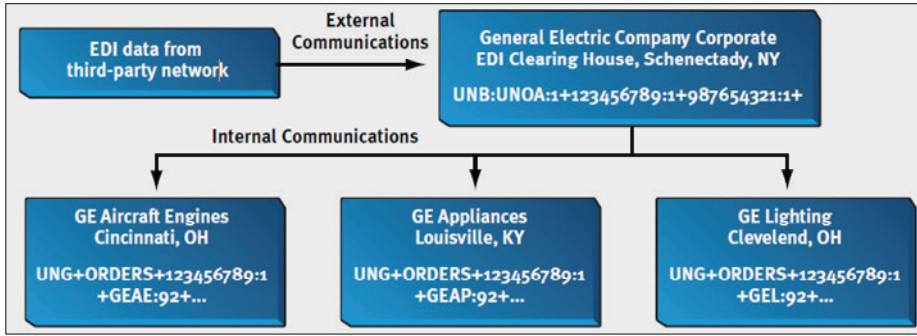
The second (middle) envelope level is around each functional group. It is defined by the UNG/UNE segments. The use of the UNG and UNE envelopes is mandatory for EDI to/from North America. This envelope groups like types of messages within a transmission. Here are a few examples of the data elements in the functional group envelope:

- Functional Group (M)
- Message Identifier (M)
- Date/Time Stamp (M): Relates multiple transactions together.
- Status of the Transfer (C): Sequences a series of related messages.
- Group Reference Number (M)
- Controlling Agency (M)
- Message Version (M)
- Application Password (C)

The UNE segment includes:

- Number of Segments in a Message (M)
- Message Reference Number (M)

The Functional Group Envelope



Functional Group Sub-Addressing

Functional Group envelopes contain a sub-addressing capability. The data that is sent to a particular receiver is addressed to the mailbox address on the UNB. Many companies want to route a group of data internally, so the UNG segment has a provision for user-defined addresses in the S006 and S007 elements.

The Interchange Envelope

The outermost level of the message envelope structure is the interchange envelope. It is defined by the UNA, UNB and UNZ segments.

This envelope is used to identify data sent from one sender to one receiver

The UNA segment contains:

- Delimiter String Advice
- Examples of included data elements

The UNB segment contains:

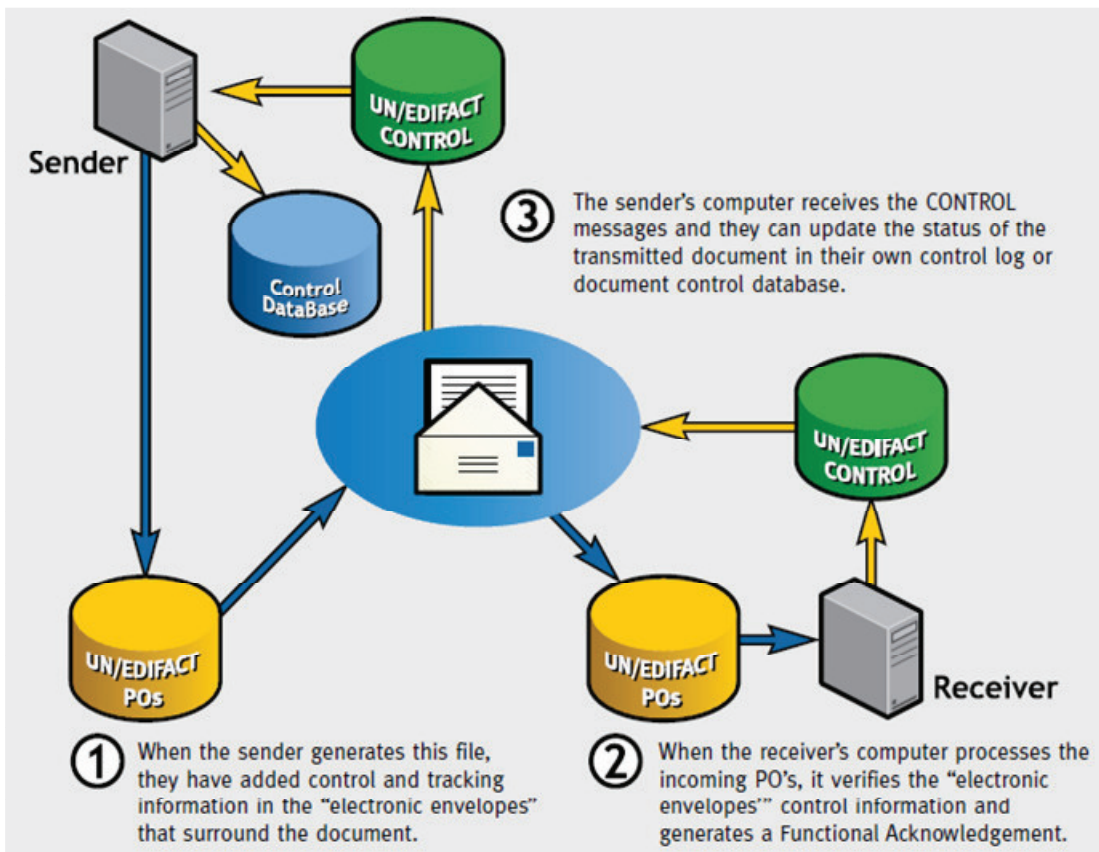
- Date/Time Stamp (M)
- Interchange Control Numbers (M)
- Password and Application Reference (C)
- Processing Priority Reference (C)
- Acknowledgment Request Indicators (C)
- Communications Agreement ID (C)
- Test Indicators (C)

The UNZ segment includes:

- Interchange Control Numbers (M)
- Counts of Messages or Groups in the Interchange (M)

The CONTRL Message

It is the responsibility of the receiver's computer to check the syntax and control numbers of the transmission and to build and transmit back to the sender this Functional Acknowledgment. The **EDIFACT CONTRL** message will provide this functionality



EDIFACT Wrap-up

Some things to keep in mind:

- EDIFACT was originally developed from a base of US ANSI standards and UN GTDI standards.
- The syntax of both standards are remarkably similar.
- Many segments are similar in the two standards.
- Translation software can readily generate/accept either EDIFACT or ANSI ASC X.12 data.
- A key issue will be the organizational improvements needed to develop/maintain EDIFACT standards within the US and Canada.

EDIFACT Self-Test I: After Completion of Part 5

Select the most appropriate answer to the following questions:

1. A primary benefit of EDIFACT is that...
it is a form of data encryption.
2. Messages:
collections of sequenced segments within defined areas.
being with the UNH segment and end with the UNT segment.
identified by a six character name.
3. Message structure is defined in the:
segment table.
4. Segments contain...
a segment tag.
simple or composite variable length data elements.
the data element separators + or :.

EDIFACT Self-Test II: After Completion of Part 10

1. Which is true of data elements:
All optional or conditional elements must be accounted for.
All conditional elements are dependent on message requirements.
All mandatory data elements must contain data.
2. EDIFACT uses two separate pieces of data in a single element:
Value and Qualifier
3. If there is an alphanumeric data element with an attribute of AN5, all of these would be correct
except:48597
4. Optional data elements without data at the end of a segment...
do not need additional data element separators to correctly position the data.

EDIFACT Self-Test III: After Completion of Part 17

1. The characteristics of Level A EDIFACT transmissions are:
upper and lower case.
2. A release character...
allows the use of a delimiter or terminator within data.
3. EDIFACT has two required levels of envelopes:
Functional Group and Message
4. For an incoming message, it is the responsibility of the receiver's computer to...
EDIFACT Standards

Banking status message

Example 1

The following is an example of a Financial Statement message sent by the bank identified by the ISO bank identification code KREDBEBB to a message recipient. The message, identified by the number 538851, which was generated on the 1st of August 2002, reports the successful execution of the payment order number 5432.

UNH+ME0000001+BANSTA:D:01B:UN:EAN003'	Message header
BGM+46+538851+9'	Banking status number 538851
DTM+137:20020801:102'	Date of message 1st of August 2002
FII+MS++BK:25:5:37010050'	Message sender identified by an institution branch number 37010050
NAD+MR+5422331123459::9'	Message recipient identified by GLN 5422331123459
LIN+1'	Start of level B
RFF+AEK:5432'	Payment order number 5432
DTM+171:20020828:102'	Payment order date 28th of August 2002
SEQ+YF2+1'	Start of level C
GIS+53'	Order executed
UNT+11+ME0000001'	Total number of segments in the message equals 11

Example 2

The following is an example of a Financial Statement message sent by the bank identified by the ISO bank identification code KREDBEBB to a message recipient. The message, identified by the number 95851, which was generated on the 1st of August 2002, reports that the execution of the payment order number 685432 was rejected because the beneficiary's bank was unknown. The incorrect beneficiary's bank details are reproduced for the message receiver. In addition the message also reports the successful execution of the payment order number 705432.

UNH+ME0000001+BANSTA:D:01B:UN:EAN003'	Message header
BGM+46+95851+9'	Banking status number 95851
DTM+137:20020801:102'	Date of message 1st of August 2002
FII+MS++KREDBEBB:25:5'	Message sender identified by ISO bank identification code KREDBEBB
NAD+MR+5422331123459::9'	Message recipient identified by the GLN 5422331123459
LIN+1'	Start of level B, number 1
RFF+AEK:685432'	Payment order number 685432
DTM+171:20020828:102'	Payment order date 28th of August 2002
SEQ+55+1'	Start of level C, number 1
GIS+83'	Transaction pending
FTX+NAI++002::91'	Rejected because the beneficiary's bank is unknown
FII+BF+994-3277711:J HOLMES+XXEDBEBB:25:5'	Beneficiary's bank and account number identification
LIN+2'	Start of level B, number 2
RFF+AEK:705432'	Payment order number 705432
DTM+171:20020828:102'	Payment order date 28th of August 2002
SEQ+55+2'	Start of level C, number 2
GIS+53'	Order executed
UNT+18+ME0000001'	Total number of segments in the message equals 18