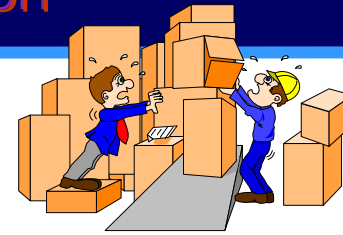




Transmission Verification and Reconciliation in ASC X12 and EDIFACT

Or ... "How Not to be a Control Freak"

This class will talk about ...



- ▶ How to verify that a transmission got from application to gateway or from gateway to application (within the trading partner's own system)
- ▶ How to verify that the transmission got from sender's gateway to recipient's gateway
- ▶ How the sender verifies that the transmission got from recipient's gateway to recipient's application
- ▶ How the X12 997 relates to the sent transaction
- ▶ How the EDIFACT CONTRL message relates to the sent message
- ▶ Overview of the X12 ISA and IEA segments
- ▶ Overview of the X12 GS and GE segments
- ▶ Overview of the EDIFACT UNB and UNZ segments
- ▶ How trading partners use/assign control numbers in X12 and EDIFACT

- ▶ **Note:** If you run across an acronym or abbreviation that is unfamiliar or not spelled out, check out the **EIDX Acronyms and Abbreviations** collection at <http://www.eidx.org/publications/abbrev/>.

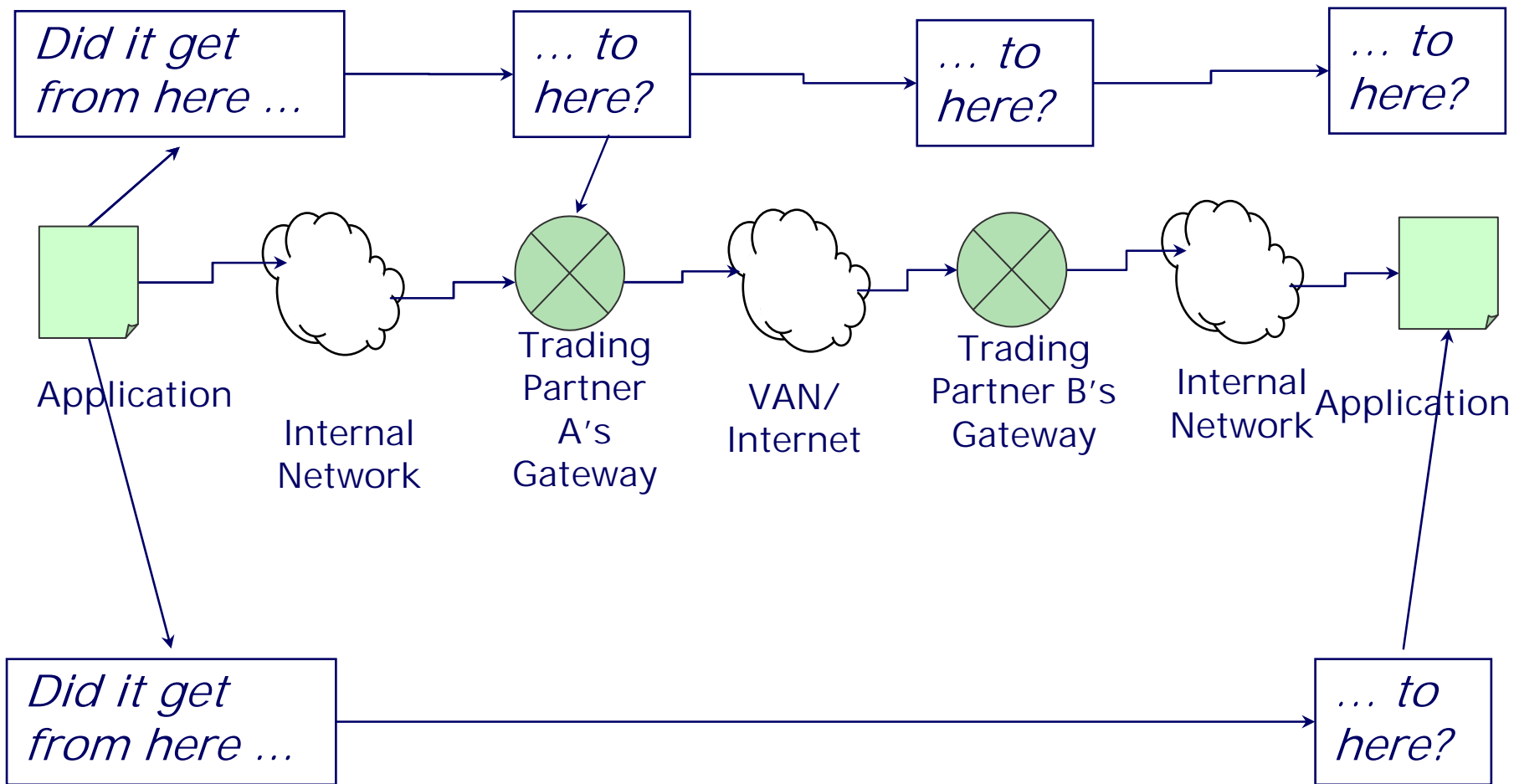


The Path

I know where it came from ...

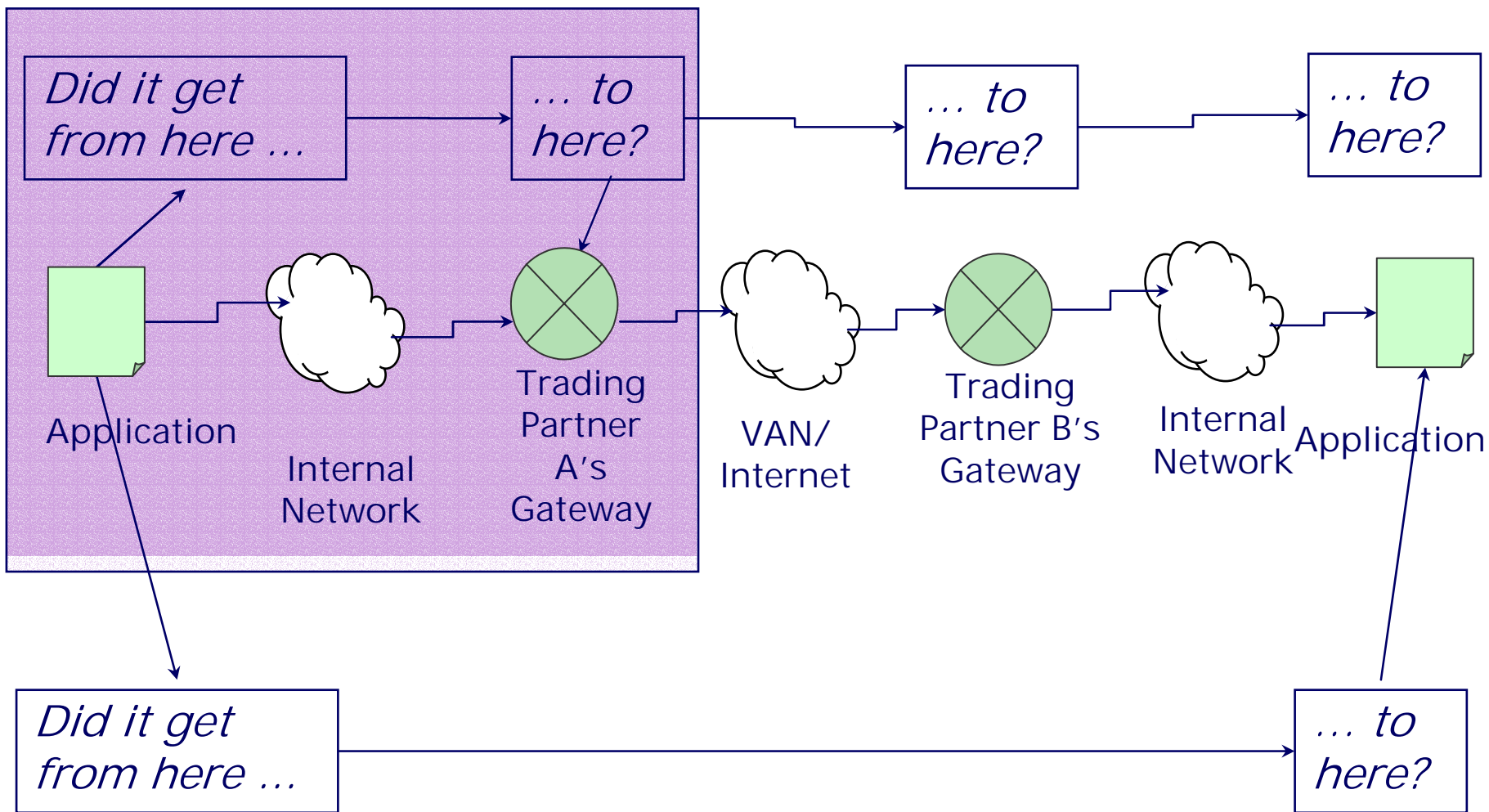
but where did it go?

Data Flow of B2B Transmissions



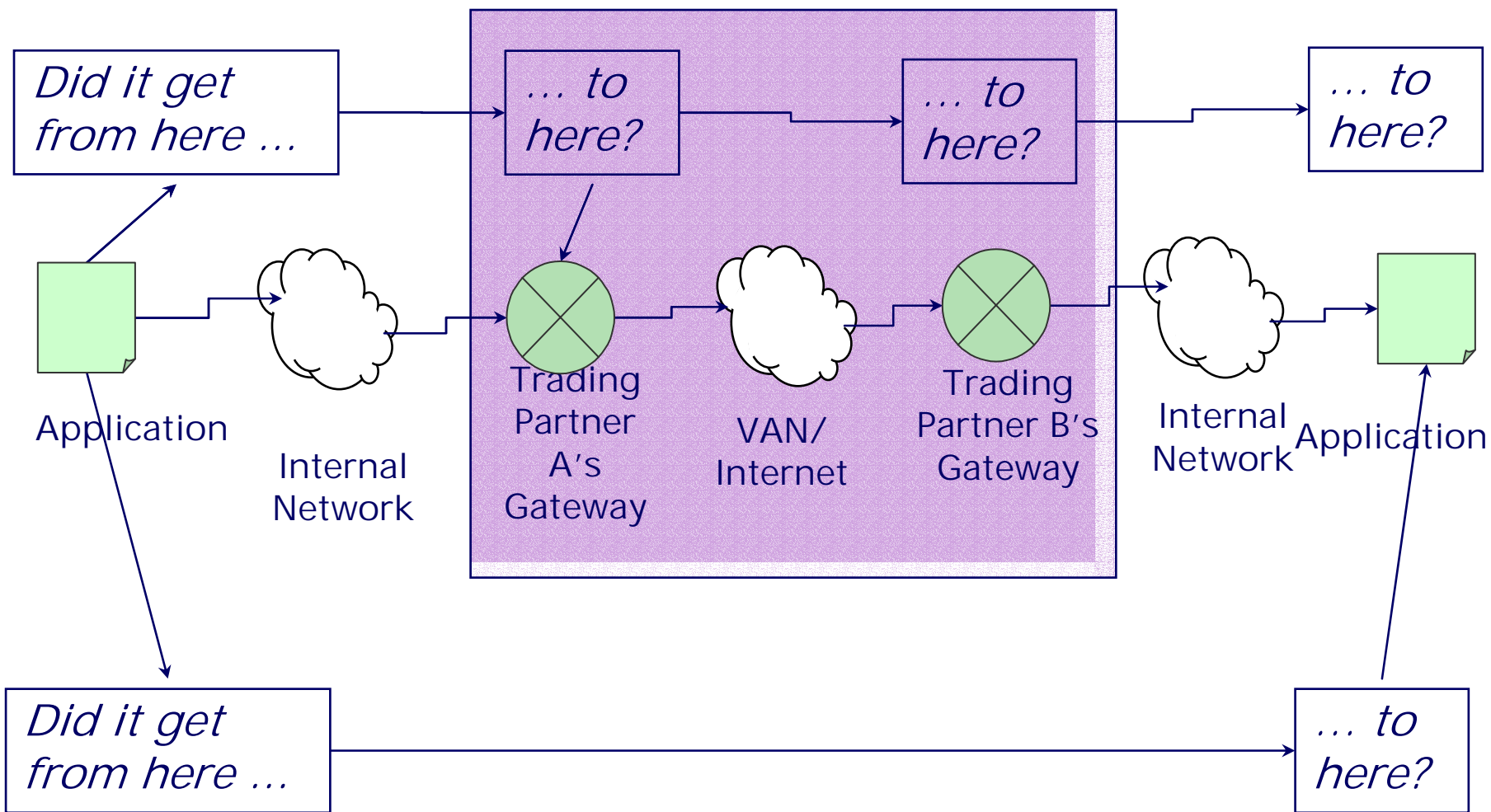


Application to Gateway



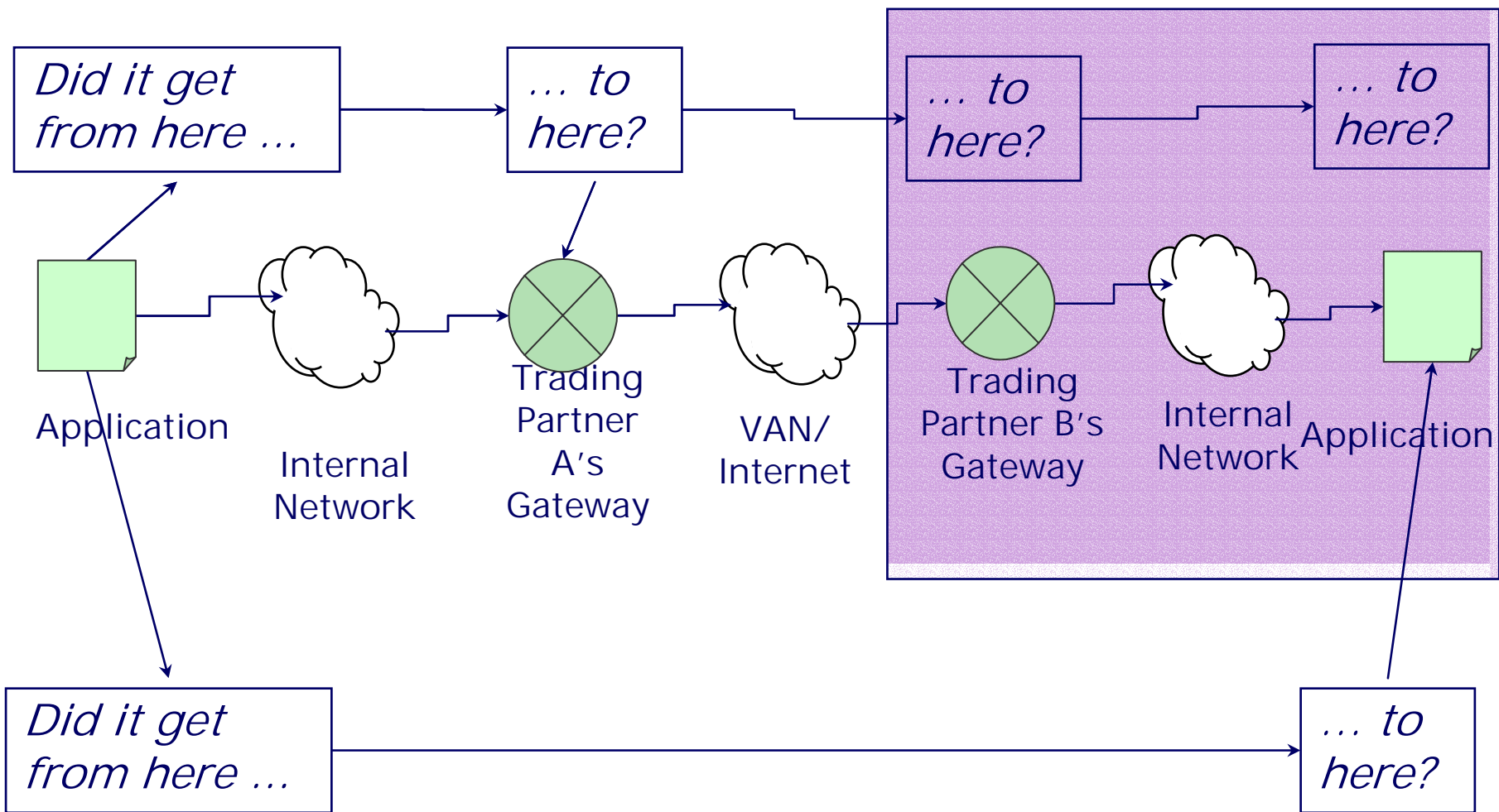


Gateway to Gateway





Gateway to Application





Is Transmission Verification Needed?

Can't we just trust the system?

Is Transmission Verification Needed?

When deciding whether a transmission verification method is needed, one has to evaluate:

- 1) The number of **incidences of lost files**, and
- 2) The **risk** to the to the trading partners if the data is not received.

Evaluating Incidences of Lost Files

- ▶ Is the incidence of files being lost between the gateway and the internal applications high, low or in-between?
 - ▶ If the incidence of lost files is *not* negligible, transmission verification of some kind is recommended.
 - ▶ If the incidence of lost files is negligible, then evaluate the risk associated with lost files.

Evaluating Risks of Lost Data

- ▶ Risk of inbound PO Acknowledgment not getting to recipient's application:
 - ▶ For most trading partners: Low - PO can be manually acknowledged

- ▶ Risk of inbound Ship Notice not getting to recipient's application:
 - ▶ For one trading partner: Low - Goods can be received without ASN
 - ▶ For another trading partner: High - ASN has critical application and/or lost file might cause considerable manual workaround

- ▶ Risk of inbound Purchase Order not getting to recipient's application:
 - ▶ Usually high - Could cause stock out for sender if not processed; but wait ... see "Application to Application" for possible alternative method.

Application to Gateway or Gateway to Application

Two methods for verifying successful transmission:

1. Send something from the recipient back to the sender telling the application the data was received.
 - a. If gateway is sender, application sends something back to gateway
 - b. If application is sender, g/w sends something back to application
2. Have the sender assign control numbers and have recipient do control number sequence checking files received from the sender.

The specific method chosen is between the gateway and the internal application, and does not directly involve trading partners.

Gateway to Gateway

Three methods for verifying success :

1. Send something from the recipient back to the sender telling the sender the data was received.
 2. Have the sender assign control numbers and have recipient do control number sequence checking files received from the sender.
 3. Review VAN/ISP reports which track whether partner has received transmissions.
- ▶ X12 users typically use Method 1, and almost never use Method 2. Just to confuse things, Method 1 employs control numbers (but not for sequence checking).
 - ▶ EDIFACT users typically use Method 3.

Transmission Verification

Let's talk about Method 2 First

Have the sender assign control numbers and have recipient do control number sequence checking files received from the sender.

This will require a little background on ISA, GS and UNB segments.

X12 ISA Interchange Control

```
ISA*00*           *00*           *01*123456789
*12*2025551234   *970504*0848*U*00200*000029281*0*P:*
```

- ISA01 I01 Authorization Information Qualifier**
- ISA02 I02 Authorization Information**
- ISA03 I03 Security Information Qualifier**
- ISA04 I04 Security Information**
- ISA05 I05 Interchange ID Qualifier**
- ISA06 I06 Interchange Sender ID**
- ISA07 I05 Interchange ID Qualifier**
- ISA08 I07 Interchange Receiver ID**
- ISA09 I08 Interchange Date**
- ISA10 I09 Interchange Time**
- ISA11 I10 Interchange Control Standards Identifier**
- ISA12 I11 Interchange Control Version Number**
- ISA13 I12 Interchange Control Number**
- ISA14 I13 Acknowledgment Requested**
- ISA15 I14 Test Indicator**
- ISA16 I15 Component Element Separator**

Qualifier indicates whether the ID following is a DUNS (01), Telephone Number (12), etc.

ISA versions are 00200 and 00300 and are independent of X12 versions (e.g. 00200 can still be used with X12 version 3020).

Recommended: Don't use (always send '0'); many TP's don't support interchange level acknowledgments in X12

X12 ISA Interchange Control

- ▶ Every element is mandatory
- ▶ Every element is fixed length (min = max)
- ▶ “Not used” elements are padded with spaces
- ▶ Character encountered after “ISA” tag tells recipient what data element separator is being used
- ▶ Character encountered at end of segment tells recipient what segment terminator is being used

Using ISA and IEA to Verify Data Integrity

```
ISA*00*      *00*      *01*123456789    *01*987654000
*970504*0848*U*00200*000029281*0*P**
```

- ISA01 I01 Authorization Information Qualifier
- ISA02 I02 Authorization Information
- ISA03 I03 Security Information Qualifier
- ISA04 I04 Security Information
- ISA05 I05 Interchange ID Qualifier
- ISA06 I06 Interchange Sender ID
- ISA07 I05 Interchange ID Qualifier
- ISA08 I07 Interchange Receiver ID
- ISA09 I08 Interchange Date
- ISA10 I09 Interchange Time
- ISA12 I11 Interchange Control Version Number
- ISA13 I12 Interchange Control Number
- ISA15 I14 Test Indicator
- ISA16 I15 Component Element Separator

- IEA01 I16 Number of Included Functional Groups
- IEA02 I12 Interchange Control Number

Determining if the interchange arrived intact:

- 1) IEA01 is count of number of GS/GE envelopes (groups) included
- 2) Control Number in IEA02 must match Control Number in ISA13

```
IEA*1*000029281
```

Assigning ISA Control Number

- ISA01 I01 Authorization Information Qualifier
- ISA02 I02 Authorization Information
- ISA03 I03 Security Information Qualifier
- ISA04 I04 Security Information
- ISA05 I05 Interchange ID Qualifier
- ISA06 I06 Interchange Sender ID
- ISA07 I05 Interchange ID Qualifier
- ISA08 I07 Interchange Receiver ID
- ISA09 I08 Interchange Date
- ISA10 I09 Interchange Time
- ISA12 I11 Interchange Control Version Number
- ISA13 I12 Interchange Control Number
- ISA15 I14 Test Indicator
- ISA16 I15 Component Element Separator

These elements form a key for sequential assignment of control number. This is the **recommended key**. However ...



Assigning ISA Control Number

- ISA01 I01 Authorization Information Qualifier
- ISA02 I02 Authorization Information
- ISA03 I03 Security Information Qualifier
- ISA04 I04 Security Information
- ISA05 I05 Interchange ID Qualifier
- ISA06 I06 Interchange Sender ID
- ISA07 I05 Interchange ID Qualifier
- ISA08 I07 Interchange Receiver ID
- ISA09 I08 Interchange Date
- ISA10 I09 Interchange Time
- ISA12 I11 Interchange Control Version Number
- ISA13 I12 Interchange Control Number
- ISA15 I14 Test Indicator
- ISA16 I15 Component Element Separator

Some trading partners only use one ID (or one ID + qualifier) as key. If the receiving trading partner is not using the same key, they can't perform sequence checking on inbound interchanges.



Assigning ISA Control Number

- ISA01 I01 Authorization Information Qualifier
- ISA02 I02 Authorization Information
- ISA03 I03 Security Information Qualifier
- ISA04 I04 Security Information
- ISA05 I05 Interchange ID Qualifier
- ISA06 I06 Interchange Sender ID** ←
- ISA07 I05 Interchange ID Qualifier
- ISA08 I07 Interchange Receiver ID** ←
- ISA09 I08 Interchange Date
- ISA10 I09 Interchange Time
- ISA12 I11 Interchange Control Version Number
- ISA13 I12 Interchange Control Number
- ISA15 I14 Test Indicator
- ISA16 I15 Component Element Separator

Many trading partners only use IDs (without taking qualifiers into account) as key. If the receiving trading partner is not using the same key, they can't perform sequence checking on inbound interchanges.

X12 GS Functional Group Header

- ▶ Every element is mandatory
- ▶ Elements are variable length
- ▶ No “Not used” elements
- ▶ Sender/Receiver ID’s do not need to be the same as ISA Sender/Receiver ID’s
- ▶ Sender/Receiver ID’s do not need to be DUNS numbers, etc - can be any value that is unique within the ISA ID. Dictionary description of GS Sender ID/Receiver ID’s: “Codes agreed to by trading partners”.

Using GS and GE to Verify Data Integrity

GS01 479 Functional Identifier Code
GS02 142 Application Sender's Code
GS03 124 Application Receiver's Code
GS04 29 Date
GS05 30 Time
GS06 28 Group Control Number
GS07 455 Responsible Agency Code
GS08 480 Version / Release / Industry Identifier Code

GE01 97 Number of Transaction Sets Included

GE02 28 Group Control Number

Determining if the functional group arrived intact:

1) GE01 is count of number of ST/SE envelopes (transactions) included

2) Control Number in GE02 must match Control Number in GS06

Assigning GS Control Number

ISA07	I05	Interchange ID Qualifier (Partner's)
ISA08	I07	Interchange Receiver ID (Partner's)
GS01	479	Functional Identifier Code
GS02	142	Application Sender's Code
GS03	124	Application Receiver's Code

GS04 29 Date
GS05 30 Time
GS06 28 Group Control Number
GS07 455 Responsible Agency Code
GS08 480 Version / Release / Industry Identifier Code

These elements form a key for sequential assignment of control number. This is the recommended key.



Assigning GS Control Number

ISA07	I05	Interchange ID Qualifier (Partner's)
ISA08	I07	Interchange Receiver ID (Partner's)
GS01	479	Functional Identifier Code
GS02	142	Application Sender's Code
GS03	124	Application Receiver's Code
GS04	29	Date
GS05	30	Time
GS06	28	Group Control Number
GS07	455	Responsible Agency Code
GS08	480	Version / Release / Industry Identifier Code



Many trading partners do not include ISA level ID and/or Functional ID as part of key for sequential assignment of control number. If the receiving trading partner is not using the same key, they can't perform sequence checking on inbound groups.

Assigning GS ID

ISA07	I05	Interchange ID Qualifier (Partner's)
ISA08	I07	Interchange Receiver ID (Partner's)
GS01	479	Functional Identifier Code
GS02	142	Application Sender's Code
GS03	124	Application Receiver's Code

GS ID is unique *within* ISA; but GS ID is not unique in and of itself. The same GS ID could be used by more than one trading partner; the GS ID must be paired with ISA ID to make unique key.

Trading Partner A

```

ISA Qualifier : 01
ISA ID       : 987654300
GS Func. ID  : PO
GS ID        : OP0001
    
```

Trading Partner B

```

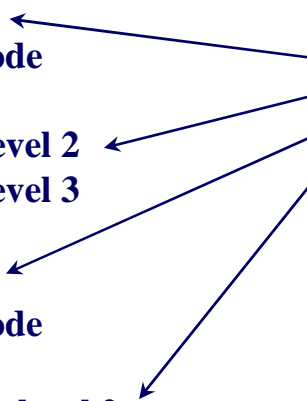
ISA Qualifier : 01
ISA ID       : 012345689
GS Func. ID  : PO
GS ID        : OP0001
    
```

EDIFACT UNB Interchange Header

UNB+UNOA:1+123456789:1:LOC001+987654321:1+970506:0651+291'

- S001** **Syntax identifier**
 - 0001** Syntax Identifier
 - 0002** Syntax Version Number
- S002** **Interchange sender**
 - 0004** Sender identification
 - 0007** Partner identification code
qualifier
 - 0008** Sender identification - level 2
 - 0085** Sender identification - level 3
- S003** **Interchange recipient**
 - 0010** Recipient identification
 - 0007** Partner identification code
qualifier
 - 0014** Recipient identification - level 2
 - 0087** Recipient identification - level 3

Three levels of ID for each party; only first level uses qualifier



EDIFACT UNB Interchange Header

UNB+UNOA:1+123456789:1:LOC001+987654321:1+970506:0651+291'

- S004** **Date/time of preparation**
- 0017 Interchange Date** ←
- 0019 Interchange Time**
- 0020** **Interchange control reference**
- S005** **Recipients reference password**
- 0022 Recipient password**
- 0025 Recipient password qualifier**
- 0026** **Application reference**
- 0029** **Processing priority code**
- 0031** **Acknowledgement request**
- 0032** **Communications agreement id**
- 0035** **Test indicator**

Date is YYMMDD; will remain 6 bytes (EDIFACT has no plan to change to add century).

ISA vs. UNB

ISA Interchange Header

- ▶ Every element is mandatory
- ▶ Every element is fixed length (min = max)
- ▶ “Not used” elements are padded with spaces
- ▶ Only one set of identifiers for each party

UNB Interchange Header

- ▶ Not every element is mandatory
- ▶ Elements are variable length
“Not used” elements are not padded with spaces
- ▶ Allows for level 2 and level 3 sender/recipient IDs

ISA vs. UNB

- ▶ ISA Interchange Header
 - ▶ Character encountered after "ISA" tag tells recipient what data element separator is being used
 - ▶ Character encountered at end of segment tells recipient what segment terminator is being used
- ▶ UNB Interchange Header
 - ▶ Data delimiters (data element separator, subelement separator, segment terminator) are UN/ECE defaults unless overridden with a preceding UNA segment
 - ▶ Element Separator: +
 - ▶ Subelement Separator: :
 - ▶ Segment Terminator: /

UNB Interchange Header

- ▶ Not every element is mandatory
- ▶ Elements are variable length
- ▶ Data delimiters (data element separator, subelement separator, segment terminator) are UN/ECE defaults unless overridden with a preceding UNA segment
- ▶ Allows for level 2 and level 3 sender/recipient IDs

Using UNB and UNZ to Verify Data Integrity

S001	Syntax identifier
S002	Interchange sender
	0004 Sender identification
	0007 Partner identification code qualifier
	0008 Sender identification - level 2
S003	Interchange recipient
	0010 Recipient identification
	0007 Partner identification code qualifier
	0014 Recipient identification - level 2
S004	Date/time of preparation
0020	Interchange control reference
S005	Recipients reference password
0026	Application reference
0029	Processing priority code
0031	Acknowledgement request
0032	Communications agreement id
0035	Test indicator

0036	Interchange control count
0020	Interchange control reference

Determining if the interchange arrived intact:

- 1) UNZ.0036 is count of number of UNH/UNT envelopes (messages) included
- 2) Control Number in UNT.0020 must match Control Number in UNB.0020



UNB - Three Levels of ID

- S001 Syntax identifier
- S002 Interchange sender
 - 0004 Sender identification
 - 0007 Partner identification code qualifier
 - 0008 Sender identification - level 2
- S003 Interchange recipient
 - 0010 Recipient identification
 - 0007 Partner identification code qualifier
 - 0014 Recipient identification - level 2
- S004 Date/time of preparation
- 0020 Interchange control reference
- S005 Recipients reference password
- 0026 Application reference
- 0029 Processing priority code
- 0031 Acknowledgement request
- 0032 Communications agreement id
- 0035 Test indicator

Instead of using GS equivalent, UNB allows three levels of ID; GS equivalent IDs can be carried as level 2 ID's (formerly called "Reverse Routing").

NOTE: These examples do not illustrate the level 3 ID's.

UNB - Allows many exceptions

S001	Syntax identifier
S002	Interchange sender
	0004 Sender identification
	0007 Partner identification code qualifier
	0008 Sender identification - level 2
S003	Interchange recipient
	0010 Recipient identification
	0007 Partner identification code qualifier
	0014 Recipient identification - level 2
S004	Date/time of preparation
0020	Interchange control reference
S005	Recipients reference password
0026	Application reference
0029	Processing priority code
0031	Acknowledgement request
0032	Communications agreement id
0035	Test indicator

One company needs different qualifier (for themselves) for each trading partner. Use of DE0007 code list not mandatory.

EDIMAN requires that message type be included on UNB. We still don't know the implications.

Assigning UNB Control Numbers

S001	Syntax identifier
S002	Interchange sender
	0004 Sender identification
	0007 Partner identification code qualifier
	0008 Sender identification - level 2
S003	Interchange recipient
	0010 Recipient identification
	0007 Partner identification code qualifier
	0014 Recipient identification - level 2
S004	Date/time of preparation
0020	Interchange control reference
S005	Recipients reference password
0026	Application reference
0029	Processing priority code
0031	Acknowledgement request
0032	Communications agreement id
0035	Test indicator

These elements form a key for sequential assignment of control number.



Assigning UNB Control Numbers

S001	Syntax identifier
S002	Interchange sender
	0004 Sender identification
	0007 Partner identification code qualifier
	0008 Sender identification - level 2
S003	Interchange recipient
	0010 Recipient identification
	0007 Partner identification code qualifier
	0014 Recipient identification - level 2
S004	Date/time of preparation
0020	Interchange control reference
S005	Recipients reference password
0026	Application reference
0029	Processing priority code
0031	Acknowledgement request
0032	Communications agreement id
0035	Test indicator

Many trading partners only use level 1 IDs (without taking qualifiers or level 2 IDs into account) as key. If the receiving trading partner is not using the same key, they can't perform sequence checking on inbound interchanges.

Just as with ISA and GS segments in X12, there are many other variations that make it impossible for partners to do control number checking.



Sequence Checking vs. Duplicates

- ▶ Some gateways can distinguish between control number sequence checking and identifying duplicates.
 - *Some trading partners can only detect duplicate control numbers by doing sequence checking. Since most trading partners' methods of assigning control number differ from each other, many don't do control number checking on inbound documents and can't detect duplicate control numbers.*
 - *Many trading partners can't distinguish between duplicate and out-of-sequence, and their software may stop non-duplicate data due to control number sequence error.*

- ▶ Then there's the issue of what to do if you do detect a duplicate control number: Need to be able to configure to either pass data through anyway and issue a warning, or stop the transmission (put it into an error state).
- ▶ Even where control number checking is done, many trading partners have no ability to stop the data.

Example: Checking Control Numbers

```

ISA*00*                *00*                *01*123456789
*01*987654000          *970504*0848*U*00200*000029281*0*P**
    
```

Compare control number sent by trading partner to table. Check ISA control number independently from GS control number.

Partner Name ABC Company
 Check Control Numbers: Y

ISA Control Number	000029281	Partner's ISA Code	123456789
--------------------	-----------	--------------------	-----------

Example: Checking GS Control Numbers

```

ISA*00*                *00*                *01*123456900        *01*
987654000 *970504*0848*U*00200*000029281*0*P**
GS*PS* 123456900FST*987654321*970504*1600*56*X*002002
    
```

Compare control number sent by trading partner to table.

Note: GS ID not unique by itself - GS ID only unique within an ISA ID.

Partnership Identifier:	EDI0123	Partner Name	ABC Company
Check Control Numbers:	Y		
GS Control Number	000000 56	Partner's ISA Code	123456900
Functional ID Code	PS	Partner's GS Code	123456900FST

Example: Checking Internal Batch Numbers

In flat file sent to gateway:
 Partnership Identifier
 Last_Batch_Number
 Current_Batch_Number

Read Last and Current
 in Flat File (okay) or
 Check against table
 (better)

Flat file record:
 EDI0123000000000350000000036

Partnership Identifier:	EDI0123	Partner Name	ABC Company
Check Control Numbers:	Y		
GS Control Number	000000036	Functional ID Code	PO
Current Batch #	000000036	Partner's GS Code	987654321
Last Batch #	000000035	Our GS Code	012345689

Control Number Checking: Limitations

If multiple “batches” (files drawing from the same pool of control numbers) are created and transmitted together, and the network used cannot guarantee that the files will arrive at the destination in the same sequence in which they left the origin, control number checking will not be feasible.

Files generated and transmitted within seconds of each other:

```
EDI0123   Control #1
EDI0123   Control #2
EDI0123   Control #3
EDI0123   Control #4
```

Might arrive at the destination in this order:

```
EDI0123   Control #2
EDI0123   Control #4
EDI0123   Control #3
EDI0123   Control #1
```


Control Number Checking: Gotchas

- ▶ From time-to-time, it is necessary for a trading partner to break the sequence of control numbers, e.g.:
 - ▶ Maximum value reached – TP needs to reset to '1'
 - ▶ New gateway software installed
 - ▶ Complication due to a company split or merger
- ▶ Notify trading partners if you are going to experience a break in control numbers
- ▶ Unless it's unavoidable, EIDX recommends that control numbers be reset to '1' *no more often than annually*

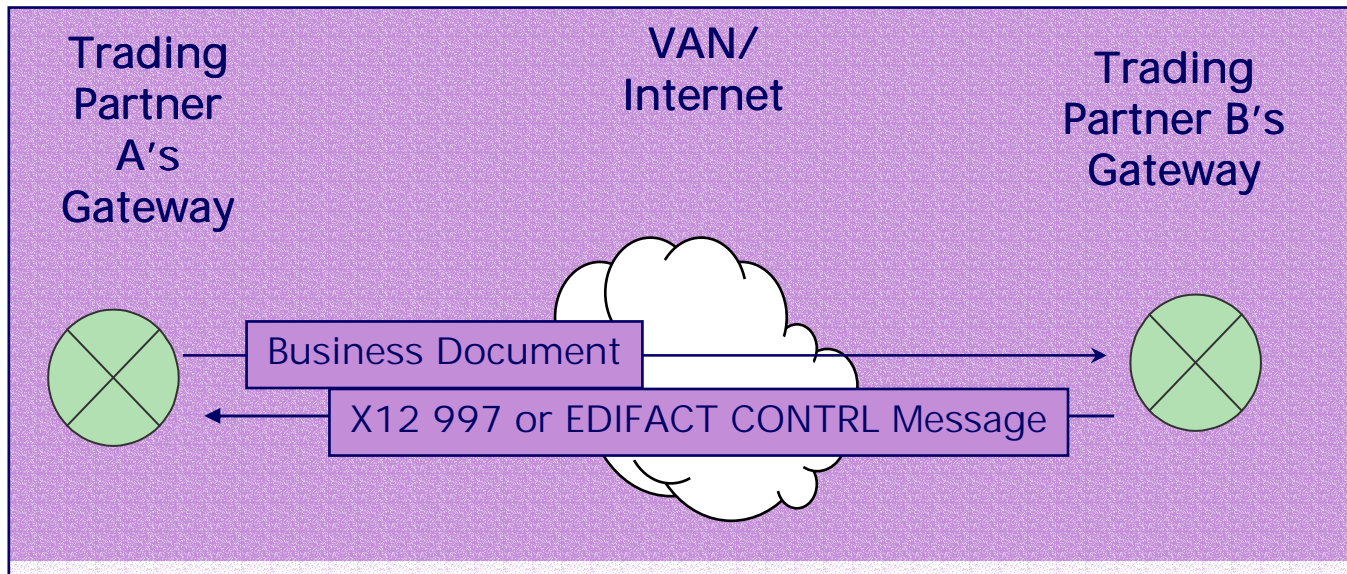


Transmission Verification

Now Let's Talk About Method 1

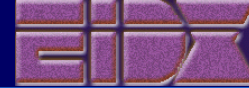
Send something from the recipient back to the sender telling the sender the data was received.

Acknowledging Transmission Receipt



X12's 997 and EDIFACT's CONTRL indicate that a transmission was received, whether it was intact, and whether or not it passed syntax checks.

Most EDIFACT users do not use Method 2 (the CONTRL message) but we are still including information in this class for the benefit of those who want to consider using it and for comparison to X12.



Matching X12 997 to Sent Group

Document from Buyer to Supplier

```
ISA*00*          *00*          *01*123456900    *01*987654321
*970504*0848*U*00200*000029281*0*P**
GS*PS*123456900FST*987654321*970504*1600*56*X*002002
```

FA from Supplier to Buyer

```
ISA*00*          *00*          *01*987654321    *01*123456900
*970505*1600*U*00200*000101071*0*P**
GS*FA*987654321*123456900FST*970505*1600*33*X*002002.
ST*997*0001.
AK1*PS*56.
AK2*830*000000001.
AK5*A.
...
AK9*A*20*20*20.
SE*44*0001.
GE*1*33.
IEA*1*000101071.
```

ISA and GS Sender/Receiver ID's on 997 are
reverse of sent group;

Functional ID and GS control number of sent
group are in AK1 segment of 997

What else is in a 997?

Document from Buyer to Supplier

```
ISA*00*          *00*          *01*123456900    *01*987654321
*970504*0848*U*00200*000029281*0*P**
GS*PS*123456900FST*987654321*970504*1600*56*X*002002
```

FA from Supplier to Buyer

```
ISA*00*          *00*          *01*987654321    *01*123456900
*970505*1600*U*00200*000101071*0*P**
GS*FA*987654321*123456900FST*970505*1600*33*X*002002.
ST*997*0001.
AK1*PS*56.
AK2*830*000000001.
AK5*A.
...
AK9*A*20*20*20.
SE*44*0001.
GE*1*33.
IEA*1*000101071.
```

Group Acknowledgment Code
 A = Entire Group Accepted
 E = Accepted but Errors Noted
 P = Group Partially Accepted
 R = Entire Group Rejected

You said you sent 20 (from your GE01),
 I received 20, I accepted 20



What else is in a 997?

Document from Buyer to Supplier

```
ISA*00*      *00*      *01*123456900  *01*987654321
*970504*0848*U*00200*000029281*0*P**
GS*PS* 123456900FST*987654321*970504*1600*56*X*002002
```

FA from Supplier to Buyer

```
ISA*00*      *00*      *01*987654321  *01*123456900
*970505*1600*U*00200*000101071*0*P**
GS*FA*987654321*123456900FST*970505*1600*33*X*002002.
ST*997*0001.
AK1*PS*56.
AK2*830*000000001.
AK5*A.
...
AK9*A*20*20*20.
SE*44*0001.
GE*1*33.
IEA*1*000101071.
```

Set Acknowledgment Code
 A = Transaction Set Accepted
 R = Transaction Set Rejected

AK3 and AK4 can be used to identify specific segments and elements that contained errors; few trading partners process these segments.

Matching EDIFACT CONTRL to Sent Interchange

Document from Buyer to Supplier

UNB+UNOA:1+123456789:1:LOC001+987654321:1+
970506:0651+291

CONTRL Message from Supplier to Buyer

UNB+UNOA:1+987654321:1+12346789:1:LOC001+970507:1232+235'
 UNH+10001+CONTRL:1'
 UCI+291+ 123456789:1:LOC001+987654321:1+5'
 UNT+3+10001'
 UNZ+1+235'

UNB sender/receiver ID's of sent group are in UCI segment of CONTRL message;
 UNB control number of sent group is also in UCI segment of CONTRL message.



Matching CONTRL to Sent Interchange

EDIFACT standards allow CONTRL sender/receiver IDs to be something other than the reverse of the sender/receiver IDs on the message being acknowledged, because IDs of message being acknowledged are in UCI.

UNB+UNOA:1+ 123456789:1:LOC001+987654321:1+
970506:0651+291'



CONTRL Message from Supplier to Buyer

UNB+UNOA:1+001236900:1+123456789:1+970507:1232+235'
UNH+10001+CONTRL:1'
UCI+291+123456789:1:LOC001+987654321:1+5'
UNT+3+10001'
UNZ+1+235'

Trading partners should not try to match CONTRL to sent interchange by reversing the sender/receiver ID's of the sent interchange's UNB segment!

What Else is in CONTRL?

Document from Buyer to Supplier

```
UNB+UNOA:1+123456789:1:LOC001+987654321:1+
970506:0651+291'
```

CONTRL Message from Supplier to Buyer

```
UNB+UNOA:1+987654321:1+12346789:1:LOC001+970507:1232+235'
UNH+10001+CONTRL:1'
UCI+291+ 123456789:1:LOC001+987654321:1+5'
UNT+3+10001'
UNZ+1+235'
```

Action Code
 4 = negative acknowledgment (rejection)
 5 = positive acknowledgment

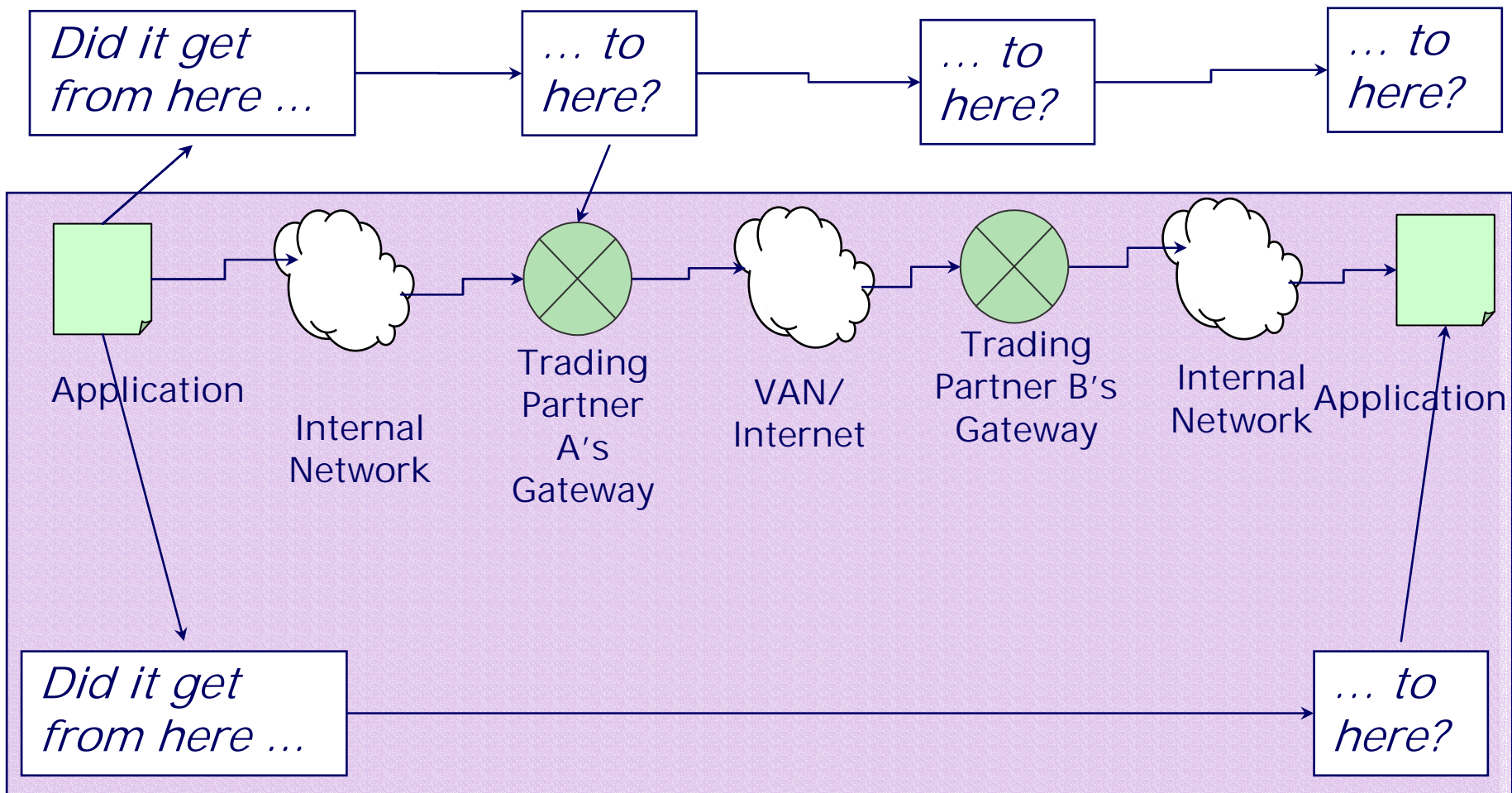


997 vs. CONTRL

- ▶ In X12, convention is to acknowledge receipt at the functional group level
- ▶ Reference made to Functional ID of transaction being acknowledged
- ▶ Sender/receiver IDs of transmission being acknowledged are assumed by reversing sender/receiver IDs on the 997
- ▶ 4 choices of acknowledgment code
- ▶ Number of transaction sets sent, received and accepted noted.
- ▶ In EDIFACT, the convention is to acknowledge receipt at the interchange level.
- ▶ No reference to message type of message being acknowledged
- ▶ Sender/receiver IDs of message being acknowledged are explicitly sent in UCI segment of CONTRL
- ▶ 2 choices of acknowledgment code
- ▶ Number of messages acknowledged not noted.



Application to Application



Application to Application

An FA or CONTRL message tells the sender that the data got to the recipient's gateway, but not necessarily that it got from the recipient's gateway to their application.

Trading partners need to discuss how it will be verified that data got all the way to the receiving application.

Example: A buyer might require the seller to send back acknowledgments (855/ORDRSP) to all PO's (850/ORDERS).



Partnership Identification

Or ... More About
Sender/Receiver ID's

We didn't say we were going to make this seem
easier ...



Trading Partnerships

ISA07 I05 Interchange ID Qualifier (Partner's)
ISA08 I07 Interchange Receiver ID (Partner's)
GS01 479 Functional Identifier Code
GS02 142 Application Sender's Code
GS03 124 Application Receiver's Code

These elements form a key for registration of partnerships.

Many systems cannot set up two partnerships with the same key.

- ▶ Can't register Planning Forecast (PS) and Supplier-Managed Inventory Forecast (PS) to same trading partner (TP) using same set of sender/receiver IDs
- ▶ Can't register inbound Order Status (RR) from TP and inbound Request for Order Status (RR) from same TP using same set of sender/receiver IDs (could be a problem when dealing with Contract Manufacturers, who are both customer and supplier)



Using Forecast as Example

When we need 2 partnerships for documents that have the same functional ID/direction.

One solution: Different forecast type codes

```
ISA*00*          *00*          *01*123456900    *01*987654321
*970504*0848*U*00200*000029281*0*P**
GS*PS*123456900FST*987654321*970504*1600*56*X*003020
BFR*FCST0504*DL*A*970504*971104*970504
```

Schedule Type Qualifier = DL for Planning Forecast, Delivery Based

```
ISA*00*          *00*          *01*123456900    *01*987654321
*970504*0848*U*00200*000029281*0*P**
GS*PS*123456900FST*987654321*970504*1600*56*X*003020
BFR*FCST0504*ZS*A*970504*971104*970504
```

Schedule Type Qualifier = ZS for SMI Forecast, Shipment Based

Trading Partnership Alternatives

When we need 2 partnerships for documents that have the same functional ID/direction.

- ▶ Use different document type codes on a beginning segment with the transaction/message
 - ▶ Not all TP's can handle this - means using one map and having the map look inside the data determine how to process the data; especially difficult if different specific types of documents need to go to different applications, e.g. planning forecast to one application, and SMI forecast to another application
 - ▶ Not all senders can send this way, especially if different types of documents are generated by different applications

Using Forecast as Example

When we need 2 partnerships for documents that have the same functional ID/direction.

Alternative solution: Different GS or Level 2 codes

```
ISA*00*          *00*          *01*123456900   *01*987654321
*970504*0848*U*00200*000029281*0*P**
GS*PS*123456900PF*987654321*970504*1600*56*X*003020
BFR*FCST0504PF*DL*A*970504*971104*970504
```

GS ID = 123456900PF for Planning Forecast, Delivery Based

```
ISA*00*          *00*          *01*123456900   *01*987654321
*970504*0848*U*00200*000029281*0*P**
GS*PS*123456900SMI*987654321*970504*1600*56*X*003020
BFR*FCST0504MR*SH*A*970504*971104*970504
```

GS ID = 123456900SMI for SMI Forecast, Shipment Based

Partnership Key Alternatives

When we need 2 partnerships for documents that have the same functional ID/direction.

- ▶ Use different GS or Level 2 ID codes
 - ▶ Some TP's cannot handle a different GS/Level 2 ID for identifying themselves nor for their trading partner
 - ▶ Some TP's can maintain multiple mailboxes (multiple ISA/Level 1 IDs) which can be used; this can be a costly solution
 - ▶ If different GS/Level 2 or different mailbox can't be used, the TP's would have to choose to exchange only one type of forecast

Partnership Key Alternatives

When we need 2 partnerships for documents that have the same functional ID/direction.

- ▶ Remember how 997's have to be sent to the same set of IDs, with sender/receiver reversed on ISA and GS (which is not true for CONTRL)?
- ▶ Some TP's can only have one 997 registration for another TP. Some can't have multiple CONTRL registrations for a TP.
 - ▶ Even if the TP could receive a different ISA/Level 1 ID or GS/Level 2 ID, they would not be able to handle the FA/CONTRL that has to be sent back to the other TP.



The End

Questions? Comments?