

XML backend "Narisphere".

IDOC backend "SAP". [SAP - Systems, Applications & Products].

standard - It is a language format.

Transaction - collection of segments.

Element - Piece of information (or) Individual data field within each segment.

Segment - Group of elements.

Envelop - Enclosed by pair session.

Syntax Token: It enables the specified range of characters and numbers.

ANSIX12 - * ; ~

Element delimiter - *

Segment " " - ~

Sub-element " " - :/x

GITS: Gentran Integration Suite.

SI: Sterling Integrator (It is a middleware tool, it is used to translate data from one standard format to another standard format).

* When EDI is in Input side (Source side) it is called Inbound

* When EDI is in Output side (Target side) it is called Outbound

VAN is also called as Repository. It is used to store files in it.

B2B

- Business-to-Business transactions
- * It is a transaction or business conducted b/w one business and another business, such as wholesaler and retailer.
 - * B2B transactions tend to happen in the supply chain, where one company will purchase raw materials from another to be used in the manufacturing process.

* B2B transactions are also commonplace for auto industry companies, as well as property management, housekeeping and industrial cleanup companies.

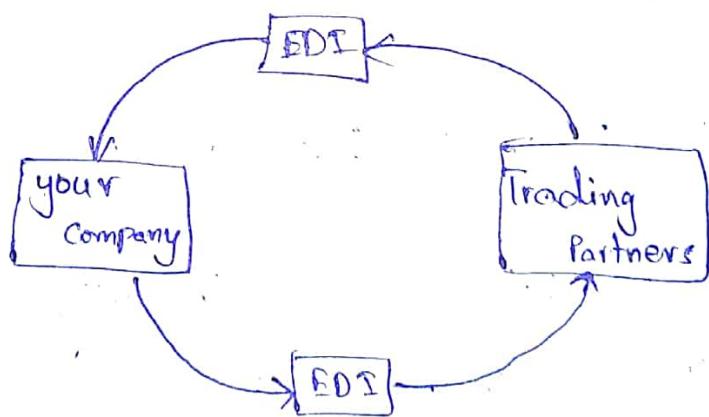
* Meanwhile, business-to-consumer transactions (B2C) are those made b/w a company and individual consumers.

EDI (Electronic Data Interchange):

→ The process of electronically exchanging business documents in a standard format.

* It is the computer-to-computer exchange of business documents b/w the companies in standard format using networks, such as VAN or the internet.

- * The EDI documents can be Invoices, Purchase orders, Transaction Bills, claims, Inventory documents and Payment Confirmation etc.
- * It replaces the paper-based exchange of business documents.
- * A standard format is used in EDI so that the computer will be able to read and understand the documents.
- * EDI documents can flow directly out of a sender's computer system directly into a receiver's computer system without any human intervention.



ANSI X12 :- American National Standard Institute (ANSI)
X12 - Sub Committee.

EDIFACT :- EDI For Administration, commerce, and Transport.

- Advantages of EDI when compared to other data formats.
- * EDI is automated.
 - * Less human intervention.
 - * Less operating cost.
 - * Eliminates time delays.
 - * Eliminates data entry errors.
 - * No transaction fees.
 - * Maintain use ~~reduces~~ volumes of transactions.
 - * Less time and higher security.
 - * Reentering the repeating data can be avoided by using looping concepts.

→ Components of EDI Systems:

The 3 main components required to send or receive EDI messages are:

- (1) EDI Standards or Application.
- (2) EDI Translation
- (3) EDI Communication

→ What are the different communication channels for EDI.

- * VAN [Value Added N/w]
- * ISP [Internet Service Provider]
- * P-T-P [Point to Point]
- * Proprietary N/w * Email.

* what is EDI - INT?

- A set of standards that define a common method for sending EDI msgs over Internet.
- EDI - INT provides guidelines for encryption, decryption and confirmations of through msgs.
- Disposition Notifications (MDN).
- Internet EDI consists of three established standards (AS1, AS2, AS3) to securely transport EDI documents over the Internet.

* Mention some of the EDI standards.

- 1) ASC X12 - Accredited Standards Committee X12
- 2) ANSI X12 - American National Standards Institute X12
- 3) UN/CEFACT - United Nations / Center for Trade Facilitation and Electronic Business.
- 4) EDIFACT - Electronic Data Interchange for Administration, Commerce and Transport.
- 5) HIPPA - Health Insurance Portability & Accountability Act.
- 6) TDCC - Transportation Data Coordinating Committee

* what are the different EDI Communication protocols.

- HTTP - Hyper Text Transfer protocol.
- SMTP - Simple Mail Transfer protocol.
- FTP - File Transfer protocol.

* Which industry is first to use EDI.

→ The transportation industry was one of the first to use EDI concepts.

→ The organization also named as Transportation Data Coordinating Committee (TDCC).

* What does 12 represents in X12.

→ The number 12 represents a subcommittee of twelve.

* What is standard.

→ A standard is a usable, well-defined generic language format that enables organizations to communicate documents without human interpretation or intervention.

* What is transaction.

→ A transaction refers either one Valid or original document. A transaction is a collection of segments in a standard sequence.

* What is Segment?

→ A segment is a record in a standard transaction or message. Each segment is identified with two or three characters identifier at the beginning of the segment.

* EDI Data formates :-

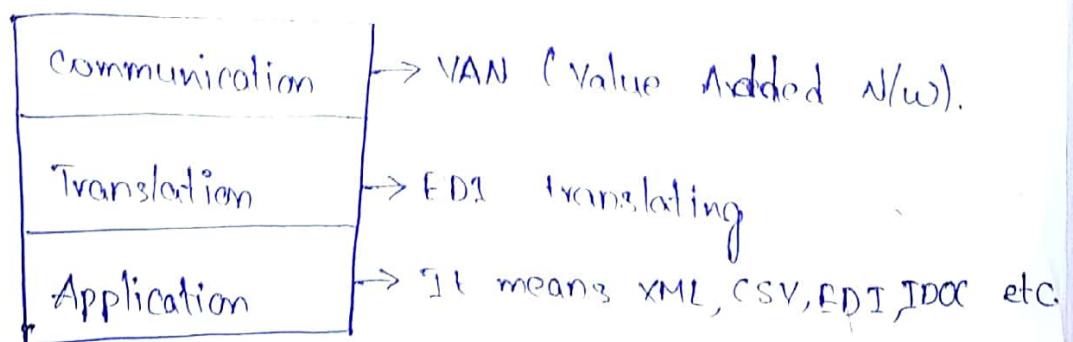
→ EDI - Electronic Data Interchange [It is in ANSI X12]

→ XML - Extensible Markup language.

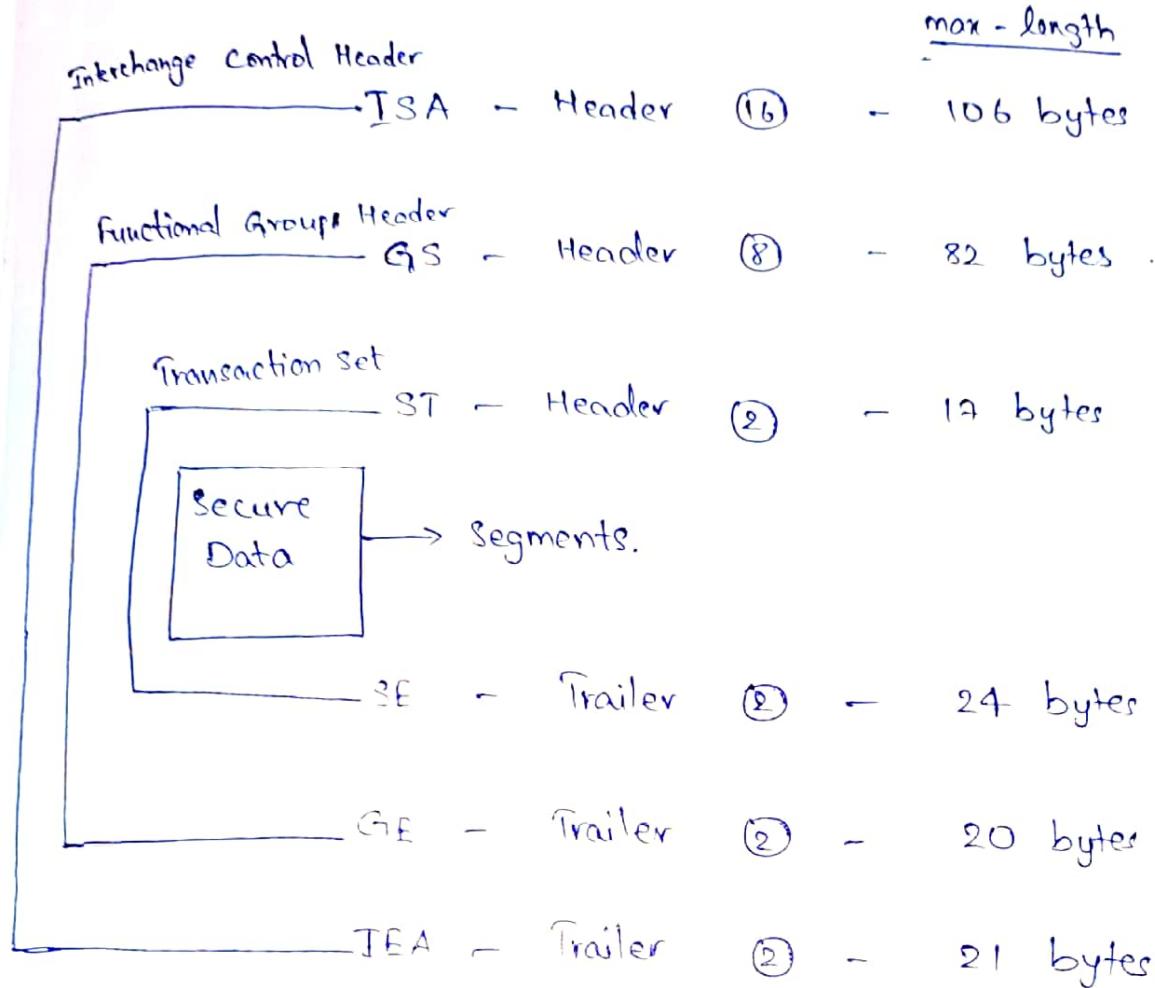
→ CSV - Comma Separated Value.

→ IDOC - Intermediate document [It has backend "SAP"]

* EDI Process In 3 steps:



* Structure of ANSI X12 :



→ max length of ISA Segment is 106 bytes.

→ max length of GS " " " " " " " " " " 82 bytes.

→ " " " " " " " " " " 17 bytes.

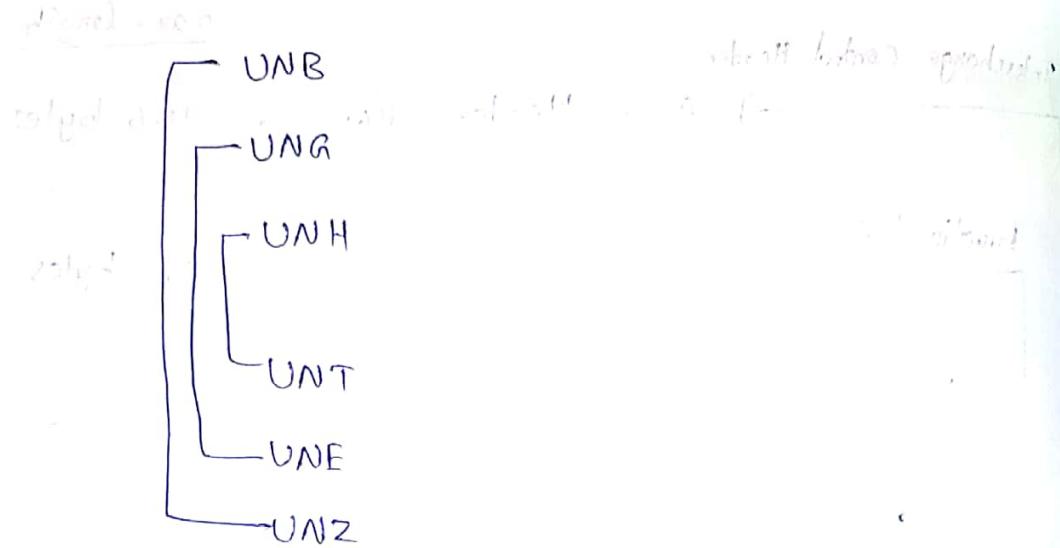
→ " " " " " " " " " " 24 bytes.

→ " " " " " " " " " " 20 bytes

→ " " " " " " " " " " 21 bytes.

etc.

* EDI FACT :



* Snail mail : Before EDI.

□ → in → CPU → out → mail → post → mail → office
in → CPU → Out → □

* First EDI used organization named as "TDCC".

* What does X12 represents in X12.

→ The number 12 represents a sub committee of twelve.

* ISA is used for?

→ The ISA Segment identifies sender and receiver of the EDI transmission.

ISA : [Interchange Control Header]

Sequence

Description

ISA01	Authorization Qualifier.
ISA02	Authorization Qualifier Information.
ISA03	Security Information Qualifier.
ISA04	Authorization Password.
ISA05	Interchange ID Qualifier
ISA06	Interchange Sender ID.
ISA07	Interchange ID Qualifier
ISA08	Interchange Receiver ID.
ISA09	YYMMDD Date.
ISA10	HH:MM Time.
ISA11	Repeating Terminator
ISA12	Interchange Control Version
ISA13	Interchange Control Number
ISA14	Acknowledgement Indicator.
ISA15	Usage Indicated (I > Information, Test Data, P > Product Data)
ISA16	Component Separator.

IEA : [Interchange Control Trailer]

IEAO1	Number of functional Groups exists.
IEAO2	Interchange Control Number.

- * Max no. of functional groups can exists b/w ISA and IEA is 99,999
- * The control number in both ISA and IEA must be same

Functional Group Header : [GS]

<u>Sequence</u>	<u>Description</u>
GS01	Functional Identifier, which defines related group of Transactions.
GS02	Applic Sender Id
GS03	Applic Receiver Id.
GS04	Date
GS05	Time
GS06	Group Control number.
GS07	Responsible Agency X= ANSI X12, T= TDCC.
GS08	Version / Release / Sub release Number.

Functional Group Trailer : [GE]

GE01 The Functional grp Contains 35 Transaction.

GE02 This tells that it belongs to the which Functional Control it belongs to.

* The control number in both GS and GE must be same.

* What is an element? what is the max no. of elements in a segment.

→ An element is an individual data field with in each segment the element represents one piece of information in a document. An element may be also thought as a field. An Segment can max no. of 24 elements.

* What is composite elements?

→ If an element consists of component element or sub elements then it is a composite element.

* What does an element contains?

→ An element contains a numeric ID, a data type and minimum and max length.

* Mention the datatypes of EDI.

→ Alphabetic - A

→ Alphanumeric - AN

→ Binary - B

→ Composite Data element - CD.

→ Character - CH.

→ Date - DT

→ Fixed length string - FS.

- Identifier - ID
- Interchanging value - IV.
- Numeric - Nn
- Password - PW
- Decimal Number - R
- Time - TM.

* What is delimiter? (Delimiter ameddi root lo untodki file properties lo).

→ A delimiter is an element separator character.

The separate can be any character that is not used in the data or control character.

* What is terminator? (Input file open chesinapudu. ISA segment last character ni "˜" Terminator antaru).

→ A terminator is the last character in the segment that indicates the end of the segment.

* What are the most commonly separators used in the ANSI X12.

- Simple Data element - asterisk (*)
 - Composite Data element - backslash (\)
 - Repeating Data element - caret (^)
 - Segment terminator - tilde (~)
- } Even delimiters to use chetharu.

* What are the Standard Separator Characters used in the UN/EDIFACT.

- 1) Data element separator - plus (+)
- 2) Component element separator - colon (:) → [850 to BEG oneili main, so a segment mandatory (M) avi unnapduku elements kudar mandatory avi untagyi].
- 3) Segment Terminator - Apostrophe (')

* What are the element requirement designators in EDI?

→ [850 to BEG oneili main, so a segment mandatory (M) avi unnapduku elements kudar mandatory avi untagyi].
1, M - Mandatory - If the segment is present, this element must be present.

2, O - Optional - (C - Conditional in EDIFACT) - (This use not based on if this element is at the discretion of the trading partners) - If A is C, then B depends upon A.

3, X - Relational - (Not used in EDIFACT) -

The requirement for this element is dependent upon the absence or presence of other element.

* Parts of transaction:

The transaction can be divided into three parts.

- Header (Table 1)
- Detail (Table 2)
- Summary (Table 3).

* Envelope: An envelope contains Control or Service segments.

* Transaction Envelope: A transaction envelope contains transactions that are enclosed in segment pairs.

* Group Envelope: A group envelope contains the same set of transactions grouped in a pair of segments that are set to a trading partner.

* Interchange Envelope:

It is a collection of groups and transactions destined for one trading partner.

* Interchange Acknowledgement: (TA1) TA1 reports receipt by the recipient of an interchange and syntactical analysis of the interchange control header and trailer only.

* Grade of Service Request (ISB):

ISB request a priority higher or lower than normally provide

* Deferred Delivery Request (TSE): It specifies the earliest time the interchange can be delivered.

* Interchange Delivery Notice (TAZ): TAZ reports successful or unsuccessful delivery and retrieval of interchange b/w service request handlers.

* functional Acknowledgement: A functional acknowledgement is an acknowledgement generated by the receiver's translator.

→ functional acknowledgement and what does it contains:

* It provides end to end acknowledgement of receipt of functional groups.

* It must be co-ordinated with trading Partner.

- (1) Functional groups
- (2) Transaction set
- (3) Segment errors
- (4) element errors
- (5) Transaction set Errors
- (6) Functional group errors.

* Difference b/w ANSI X12 and EDIFACT.

Differences b/w ANSI X12 & EDIFACT.

↓ It is used in the US & It is used in Europe, Asia, Africa, Australia, South America, etc.

(2) A group of segments is said to be transaction set. (2) A group of segments is said to be message.

(3) ANSI X12 is developed by Accredited standards Committee (ASC X12 Committee).

(3) EDIFACT is internationally agreed upon standards.

(4) Terminators (386) various separators.

(5) ISA / IEA (386) (386)

(5) UNB / UNZ

(6) GS / GE (388) (388)

(6) UNG / UNE

(7) ST / SE (388) (388)

(7) UNH / UNT

(8) The Functional Acknowledgment for EDIFACT is CONTRL.

* C Error:

The C Error function raises a Compliance error.

Eg: ① Mandatory Data Missing.

② Block Data unknown.

③ Incorrect Data Format.

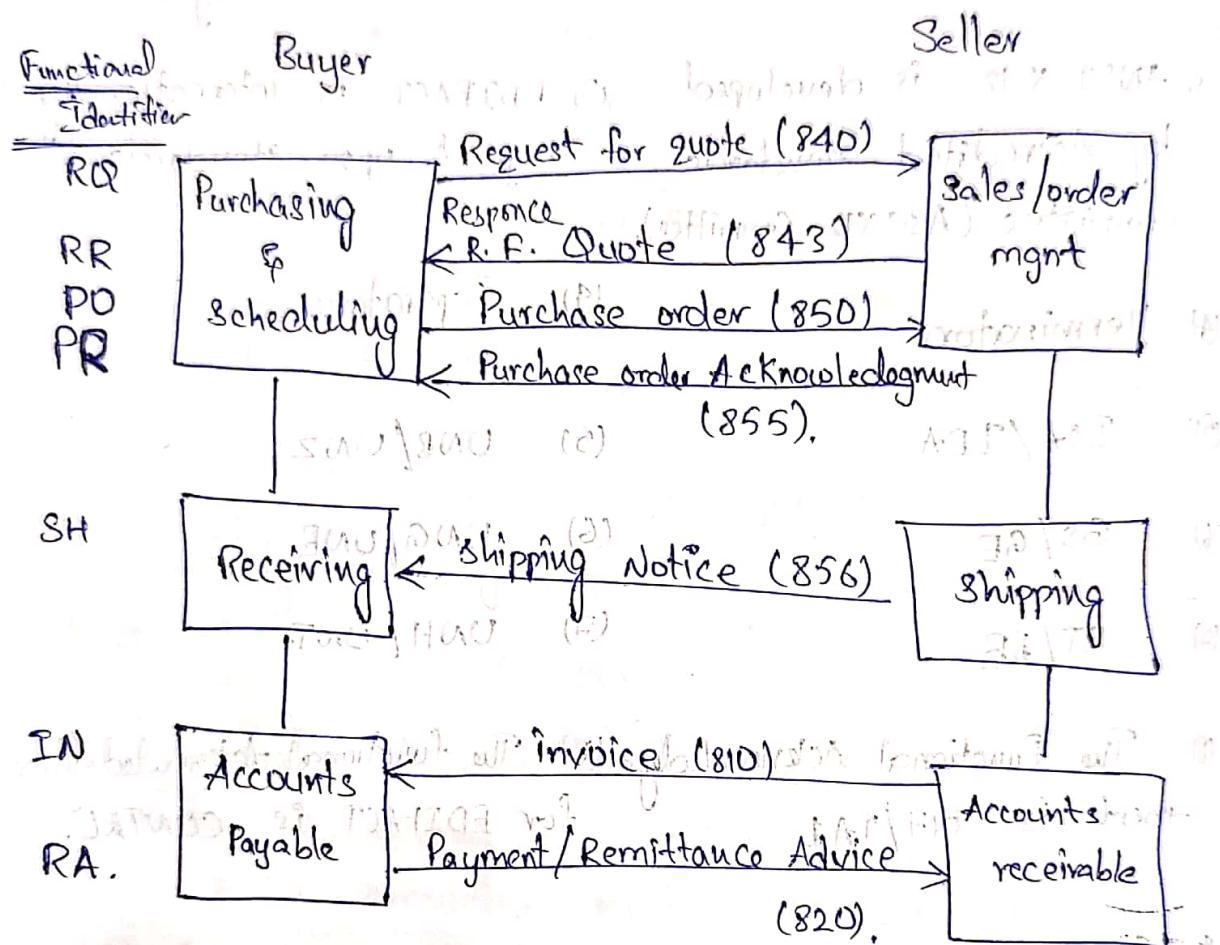
④ Invalid Data.

⑤ Mandatory Block missing.

⑥ Mandatory Group missing.

Retail flow:

The various process which help customer to procure the desired merchandise from the retail stores for their end user refer to as retail management.



- * Retail flow is transaction b/w Buyer and seller
- * Buyer wants some items from seller. So, Buyer sends 840 transaction to seller, if the items are available with seller then seller sends 843 transaction. Then buyer sends 850 purchase order to seller. Seller receives 855 purchase order. Seller sends 855 purchase order acknowledgement to buyer.

- * As well as sellers 856 for shipment notice to buyer for the tracking purpose of an item.
- * When the item reaches to buyer 810 transaction send by seller to buyer when buyer receives 810 invoice. Buyer Pay either amount by sending 820 payment order.

Transaction: exchange of information or movement of

(1) Request for quote (840):

A buyer has a need for merchandise and creates a request for quote (RFQ) document.

Mandatory Segments: [BQT, PO1, CTT]

* BQT: Beginning Segment for request for quotation.

Purpose: To indicate the start of request for quotation

transaction set and to transmit identifying numbers and dates.

* PO1: Purchase order baseline item data.

Purpose: To specify basic and most frequently used purchase order line item data.

CTT: Transaction Totals.

Purpose: To transmit a hash total for a specific element in the transaction set.

(2) Response to request for quotation (843)

→ seller sends this to buyer.

Mandatory Segments: [BQR, CTT]

→ seller sends this to buyer.

* BQR: Beginning Segment for response to request for quotation.

Purpose: To indicate the start of the response to request for quotation transaction set and to transmit identifying numbers and dates.

* CTT: Transaction (Totals) ~~not taught~~

Purpose: To transmit a hash total for a specific element (in the transaction set).

(3) Purchase [Order (850)]

→ Upon acceptance of the quote the buyer sends a purchase order (PO) document to the seller requesting goods at quoted prices.

Mandatory Segments: [ST, BEG, PO1]

* BEG: Beginning segment for purchase order.

Purpose: To indicate the beginning of the purchase order transaction set and transmit identifying numbers and dates.

BEG-03: Purchase order number.

PO1: Base line Item Data.

Purpose: To specify basic and most frequently used item data.

↳ ~~item data~~ ~~with the help of~~ ~~which~~ ~~is~~ ~~used~~ ~~in~~ ~~the~~ ~~order~~ ~~to~~ ~~specify~~ ~~basic~~ ~~and~~ ~~most~~ ~~frequently~~ ~~used~~ ~~item~~ ~~data~~.

↳ ~~item data~~ ~~with the help of~~ ~~which~~ ~~is~~ ~~used~~ ~~in~~ ~~the~~ ~~order~~ ~~to~~ ~~specify~~ ~~basic~~ ~~and~~ ~~most~~ ~~frequently~~ ~~used~~ ~~item~~ ~~data~~.

4. Purchase Order Acknowledgement (855):

→ ~~PO1: [855]~~ ~~is~~ ~~simple~~ ~~protocol~~ ~~?~~

→ The seller responds acknowledging that goods can be delivered at the prices specified in the time frame indicated.

Mandatory Segments [BAK, CTT, ST, SE]

* BAK: Beginning Segment purchase order Ack

Purpose: To indicate the beginning of the purchase order acknowledgement Transaction Set and transmit identifying number and dates.

BAK-03: Purchase order number

* CTT: Transaction Totals.

Purpose: To transmit a hash total for a specific element in the transaction set.

* ST: Transaction set Header

* SE: Transaction set Trailer.

ASN:— Advance Shipment Notice.

5) Shipment Notice / Manifest (856):
[BSN, HL, DTM]

→ The seller ships the goods and sends a shipping notice (SH) providing details regarding the carrier, product identification, dates, and other information about the shipment.

Mandatory Segment: [BSN, HL, DTM]

* BSN: Beginning Segment for Ship Notice

Purpose: To indicate the beginning of the shipment transaction set and transmit identifying numbers and date.

(Or)
To transmit identifying numbers, dates, and other basic data relating to the transaction set.

* DTM: Date/Time Reference.

Purpose: To specify pertinent dates and times.

* HL: Hierarchical level.

Purpose: To identify the dependencies among and the context of hierarchically related groups of data segments.

Hierarchical level codes:

S - Shipment

O - Order

P - Package

I - Item.

Hierarchical levels:

HL01 - Hierarchical ID no.

HL02 - Hierarchical parent ID no.

HL03 - Hierarchical level Code.

HL04 - Hierarchical child code.

* BSN-PRF = 01 purchase order number

6) Invoice (810):

The seller sends an invoice document, billing the buyer for the goods. There is also an EDI document (Sales Billing) shown at an and ~~combined~~ ~~billing notice~~ ~~and~~ ~~combined~~ ~~billing notice~~

Mandatory Segment: [BIG, TDS]

* BIG: Beginning Segment for invoice.

Purpose: To indicate the beginning of an invoice

transaction set and transmit identifying number and dates.

BIG-02 Contains invoice number

BIG-04: Purchase order number.

* TDS: Total monetary value summary.

Purpose: To specify the total invoice amounts and amounts.

Purpose : To specify the total invoice discount amounts.

TDS-01: Total invoice amount.

CTT : Transaction Total.

6. Payment Order / Remittance Advice [820]:

→ The buyer notifies his bank that payment is to be made (payment order) and used another form of the same document (remittance advice) to inform the seller that payment is being made.

Mandatory Segments: [BPR]

* BRR : Beginning Segment for payment order / Remittance Advice.

BPRO2 : Payment amount.

RFQ : (Request for Quotation)

BOT:-

M Segment : Beginning segment for request for quotation.

Elements:

BQTO1 - Transaction set purpose code.

BQTO2 - RFQ reference no.

BQTO3 - Request Date

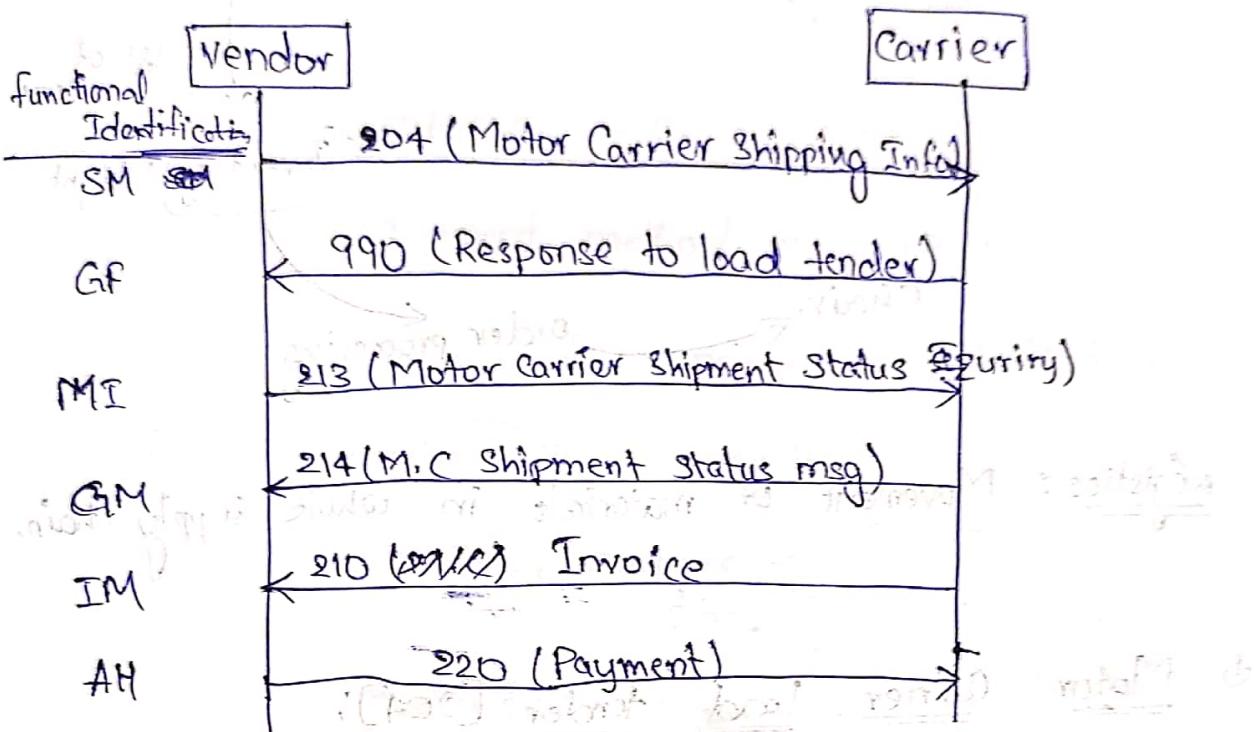
BQTO4 - Date/time Qualifier

BQTO5 - Date.

BQTO6 - Purchase order Type Code

Logistic Flow

Transactions used in logistic flow:



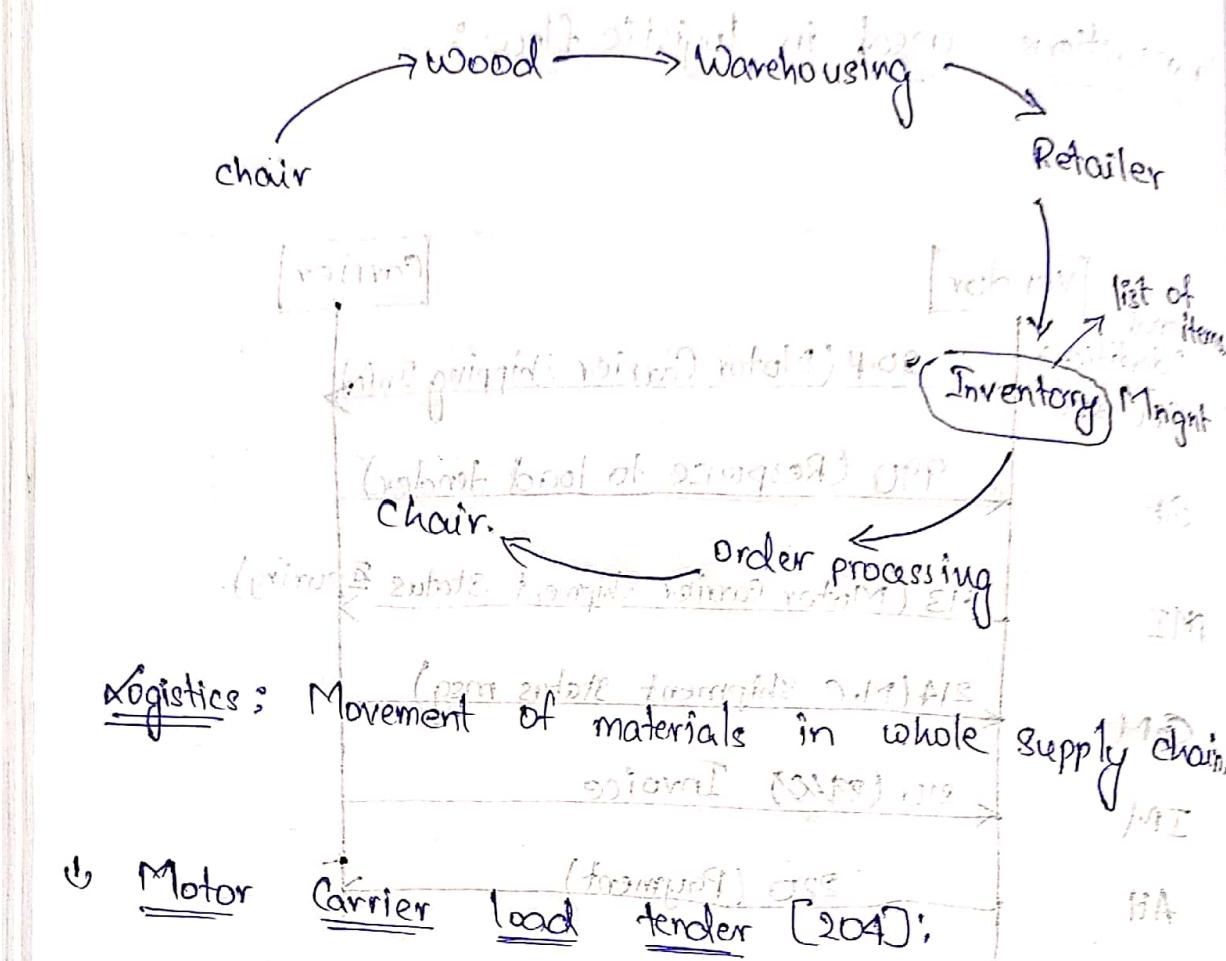
Logistic Management:

It is the power of Supply chain management to plan, implement and control the efficient and effective forward flow (and) receives reverse flow and storage of goods, services and related information.

Supply chain Management (SCM):

Transforming raw materials into products and getting it to customers.

Ex: chair



④ Motor Carrier load tender [204]:

→ This transaction set can be used to allow shipper or other interested parties to offer (tender) a shipment of full load (truck load) motor carrier including detailed scheduling equipment requirements, commodities and shipping instructions pertinent to a load tender.

Mandatory Segments:

* B2: Beginning Segment for Shipment information.

Purpose : To transmit basic data relating to shipping
shipment information.

B201 : Tariff Service code (TSC) [990]

B202 : Standard carrier Alpha Code (SCAC)

B204 : Shipment Identification number (SID)

B206 : Shipment method of payment Code (SMP)

* B2A : To allow for positive identification
of transaction set purpose.

Set purpose →
① 00 - create.
② 01 - cancel.
③ 02 - update.
④ 04 - Confirmation.

* S5 : Stop off details

S501 : Stop sequence number.

S502 : Stop reason code

② Response to load tender [990]:
→ Whether it will pick up a particular shipment
previously offered by the shipper vendor.

* [Shipper accepting a shipment offer] →
* Kakigirattha appudu 990 send chestam (Accept or Decline).

Mandatory Segments:

* B1: Beginning Segment for booking or pickup/Delivery

Purpose: To transmits identifying numbers, dates and other basic data relating to the transaction set.

(212) → B101: Standard Carrier Alpha Code

(213) → ~~B102: Shipment Identification number.~~

~~B104: Reservation Action Code~~

(remark)

* K1: To transmit information in a free-form format; for comment or special instruction

~~K101: free form msg~~

~~K102: free form msg~~

(3) Motor Carrier Shipment Status Inquiry [213]:

→ The transaction set can be used to request shipment status from a motor carrier. The requests may be a shipper or consignee.

Mandatory Segments:

* B110: Beginning Segment for shipment status inquiry.

↳ To cancel off charges and manipulate

Purpose : To transmit identifying numbers, dates and other basic data relating to shipment information.

B1101 : ID5, Coding Qualifier.

B1102 : ID5, Code.

(Remark) B1103 : Date (format - 5X)

* K2 : To transmit information in a free-form format, if necessary, for comment or special instruction.

K201 : free form msg.

(4) Transportation Carrier Shipment Status [214]:

→ 214 transaction set represents a transaction Carrier Shipment status msg. It is typically used by transportation carriers, such as trucking Companies to provide shippers and consigned with the status of their shipments.

** [Carrier nuchi vendor ki 214 aane ki send chaste by using some code like, AA, AB etc]

ATM [Shipment status codes].

and abhi aage jankari - 2002

* AT7 : Shipment status Codes.

AA - Appointment Set for pickup location

AB - Drop location

X3 - Arrived pickup location

CP = Complete loading.

AF - Starting truck.

X6 - Tracking location.

X1 - Arrived drop location

D1 - Complete unloading.

CD - Departure from the drop location.

Mandatory Segments :

* B103 : Beginning segment for Transportation

Carrier Shipment status message of example

Purpose : To transmit identifying no's and other

Basic data relating to shipment information

B1001 - Reference Identification

B1002 - Shipment Identification number

B1003 - Standard Carrier alpha Code

N₁ - Address Name. (Close) ~~Required~~ ✓

N₂ - Address Information (Additional Name info)

N₃ - Address Information

N₄ - Geographic Location.

(b) Invoice: [210] ~~info forage pertaining to P&R~~

→ It is used both as a motor carrier invoice to request payment or as details pertaining to ~~request~~ motor freight shipments(s) charge

[* To send invoice from carrier to vendor].

Mandatory Segments: [B301 : E108]

* B3 : Beginning Segment for Carriers' invoice.

Purpose : To transmit basic data relating to the carriers' invoice.

B301 - Shipment Qualifier

B302 - Invoice no.

B303 - Shipment Identification no.

B304 - Shipment method of Payment
Code.

6. Payment [220]: ~~Vendor~~ ~~220~~ → ~~Carrier~~

After 210 transaction, 220 will be send from vendor to carrier to pay the invoice / payment.

Mandatory Segments:

* B9: Beginning Segment for "Logistics Services".

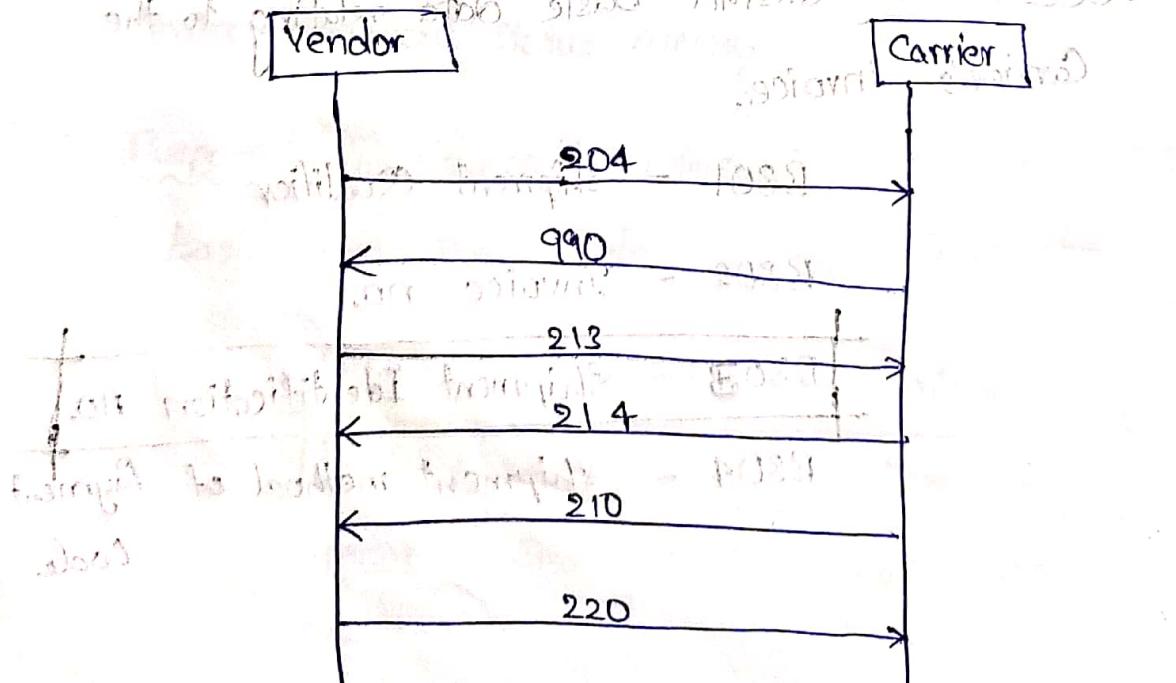
Purpose: To indicate the beginning of a logistic service transaction set.

B901 : Reference Identification

B902 : Transaction set purpose code.

B903 : Shipment method of payment

Xogistic flow:



- * Logistic flow is a third party transaction. Transaction b/w vendor and carriage firm [Carrier].
- * Vendor has some goods to send buyer. So transportation of goods vendor needs some third party help.
- * Vendor send 204 transaction to Carriage firm for truck are available to transport goods to a particular location at particular time. If it's available carriage sends accept by sending 990.
- * Then vendor send 213 for shipment ~~request~~ ^{Inquiry} after that carrier send 214 Shipment message for vendor like AA, AB, X3, CP, AF, XI, DI, X6, CD.
- * Carrier send 210 ^{Invoice} ~~reverse~~ when the items sends to a particular destination.
- * When items received by the destination then vendor pays amount to carrier (Carriage firm).

Pre-session rule:

In this we can declare and initialize the variables and also we can write the conditions as per the business rules. It will get executed at the beginning of the map translation. The variables we declare in the pre-session rule are having global (scope) variables.

Extended Rule: relevant with busi. conditions
we can write business rules using functions available in the SI map editor and we can declare the variables. The variables we declare in the extended rule are having local scope.

Standard Rules:

Below we can apply standard rules to the field level using predefined functions available in the standard rule. Those predefined functions are,

1 Select:

If it will enables you to access the internal Server database.

2 Update:

To move data in a map to process data and to update the process data.

3 Use System Variable:

It will print the current date and time of the system.

A) Use constant:

It is used to move a constant value

to field/element, all other nodes will be identified with the previous step like #1, value constant

(S) Use accumulator:

It gives access to a set of numeric variables that you can manipulate by using numeric operations and then transfer to field.

To count the no. of iterations of a group

(6) Loop Count:

To count the no. of iterations of a group

(7) Use Code:

Used for validating the constant of the field.

brackets don't affect mapping price

Code list: It is a set of codes, we can get required values based upon a key pair value.

i) Internal code list - used within the map

ii) External code list - use any map (SI, Server)

Query: Select Text1 into DB Returnvalue from Codelist;

where name = "codelistname" & Sender Code / Receiver Code

which return code = # Sender Code / Receiver Code.

Post-Session Rules: It is same as Pre-sessions but only diff. is pre-session rules will get executed at the beginning of the translation but post-session rules will get executed after the translation.

Extended Rules:

A refactoring of the code

It will populate the temporary records.

- * Add a conditional logic.
- * Perform Arthemetic operations.
- * Assign and Retrieve values from a variable.

→ Extended Rules enable us to use a Gentran proprietary programming language to perform virtually any mapping operation.

→ Extended rule consists of a declarations section followed by a statements section.

→ Declarat

Standard rule:

(5) Use accumulator:

Primary Access to a set of numeric variables can manipulate by using numeric operations, then transfer to fields.

* Count of line items.

* Calculating hash totals.

* Resetting and calculating a value total.

Operations Types:

rotation of opened primary part

→ Increment primary part

→ Decrement

environmental to system after transaction

→ Sum in primary

note: standard by default not use

→ Hash sum in primary

→ load primary

→ use primary

→ zero primary

→ etc.

23/07/2023

Functions in Maps

Function: To convert one data type to another data type.
Three types of functions in Maps:

i) Numeric functions

ii) String functions.

iii) Date functions

iv) Numeric Functions: To convert one data type to another data type.

(i) len

(ii) atoi

(iii) atof

(iv) ntoa

(v) Count

(vi) Delete

v) len: The len function counts and returns the number of characters in a string.

Syntax: `int len(string str);`

Example: `a = len("hello");`

vi) atoi: This function converts string into integer.

Syntax: `int atoi(string s);`

Ex: `integer a;`

"a" contains the value 5. \Rightarrow `String [20] S;`
 \Rightarrow `s = atoi(s);`

(iii) aton: Converts string into real number
Syntax: `real aton(string)`

Ex: `real b;`

`String [20] s;`

`s = "5.5";`

`b = aton(s);`

Output: `"b" contains the value 5.5, which is`

(iv) ntoa: Converts real number to string
Syntax: `ntoa(real, string);`

Ex: `real b;`

`String [20] s;`

`b = 5.5;`

`ntoa(b, s);` [\because s contains "5.5"]

Output: `s contains "5.5"`

Q Count: This function returns the number of

definition: iterations of a group.

* * * [Count func is used to find no. of iterations of a group]

Note: When a count extended rule is performed

on an empty group, the value of -1 is returned from count

returned from count

in a group

`2 (0.8) print < (2) foto > 2 & 2 sub with wished`

Syntax: Count (\$GroupName [*]);

Ex: integer i;

i = Count (\$GroupName [*])

[*: [*] → wildcard that count the no. of iterations
of the group name].

Ex: Return the total iterations of the groupname
group with in the first third iteration of the
parent group.

works out to be Count (\$GroupName[3][*]);

vii) Delete: The delete function delete the

specified occurrence of a repeating record
or group.

Syntax: Delete (\$GroupName [Occurrence]);

Without specifying occurrence of the group that you
want to delete, it will delete all the occurrences.

Ex: Delete (\$GroupName[2]);

// Delete the second occurrence of the
GroupName group.

(2) String Functions: Comparison b/w two strings & in register var

(i) Left

(ii) Right

(iii) mid

(iv) strdate

(v) Concat

(vi) strstr

(vii) trim

(viii) trim left

(ix) trim right

(x) Date

i) Left: This function extracts ~~way~~ specified number of characters from the left of the string

Var (or) field and returns the result as

a string, ~~soft~~ without ~~soft~~ field as var.

↳ Syntax: ~~String-variable = left(String-var, num-char)~~

where: num-char = integer variable

↳ Example: ~~String-var = left(S, 3);~~

ii) Right: This function extracts a specified number of characters from the right of the string

Var (or) field and returns the result

as a string

~~soft~~ ~~variables~~ ~~variables~~ ~~soft~~ field as

iii) mid: This function extracts from the specified position in the string.

* Either to the end of the string or for a specified number of characters, and returns the return^(the number of characters) the resultant string. This function is "zero-based function".

Example: String [10] S; will be about work with

String [3] S₁; left operator overload with

String [3] S₂; right operator overload with

String [4] S₃; mid operator overload

String S = "Bhaskar"; here work with IT

S₁ = left(S, 3)

S₂ = right(S, 3)

S₃ = mid(S, 3, 4);

Syntax:

string var = mid(string-var, start-pos, num-char);

start-pos = integer variable

num-char = integer variable

(iv) strdate: This func converts a datetime type into a string using a format.

Syntax: strdate(datetime, "format", string);

where :

datetime - datetime variable (month specified as 1-12).

format - date format

String - string variable

Ex: `datetime d;`

`snwfile [String[8]], S;` be a file for saving
`strdate(d, "%Y-%m-%d", S);` generate with
current date.

(v) Strstr: (0-based function)

This func finds a substring ^{inside} a string

This functions returns the position of the first instance of the specified substring with in the specified string.

* If Strstr does not find the specified substring in the string.

It returns a value of -1.

Syntax: `integer = strstr ("String", "substring");`

where `integer` = integer variable

`(read, write, readwrite, print, printf) file = ray - print`

`String = String,`

`substring = part of the string.`

`Substring = part of the string.`

Ex: `integer d; // finds "is" in "mississippi"`

`d = strstr ("mississippi", "is");`

// finds the first instance of the substring "is" inside the string.

// "mississippi" and returns the position of that first substring.

// `d=1` because this function is zero-based

(vi) Trim: The trim func enables to remove white space from both sides of a string by default.

* we can also specify a characters, specify for the trim func to remove before and after a string.

* If we want to remove a characters, specify the character in the appropriate case (upper-lower case).

Example: Trim removes all white spaces from a string.

Syntax: String = trim (String-to-trim) [or]

String = trim (String-to-trim, Character-to-

trim).

(vii) Date: String to date time type.

Syntax: Datetime = date ("format", string);

Ex: datetime d;

d = date (1999,4,6);

(viii), Trim left: It will remove the left side of a white

space.

String = trimleft (String-to-trim, character-to-trim);

(ix) Trim right:

It will remove the right side of a white space.

String = trimright (String-to-trim, character-to-trim);

(X) Concat:

It prints a string after concatenating more.

This func concatenates a specified number of

characters from one string onto the end of another string.

Method signature of concat with args:

Syntax: Concat (String1, String2, num-char);

Where String1 & String2 are strings of type `String`.

Where num-char = no. of characters from the second

[iv] `String1.charAt(0).concat(String2)` prints `String2` on to the end of the first string.

[v] `String1.substring(0, num-char).concat(String2)` prints `String2`

Ex: `String [10]: S1, S2;`

Concat (S1, S2, 5);

Output: `S1S2S2S2S2`

Numeric Function:

Cerror: It is a compliance error. It mainly uses for debugging.

* If the condition is true then we can use `cerror` to print some message.

* It is used to crack the flow of execution.

* It is valid on the input side of a map only. It is a user-defined error.

Syntax: `cerror(100, $segment_name.#field-name)`

where `$segment_name` is the segment name and `#field-name` is the field name.

enter error info

:`(charAt(0).concat(charAt(1)))` if print = print2

The error function raises a Compliance error.

Ex: all mandatory Data Missing.

1. Block Data Unknown.

2. Incorrect Data format.

3. Invalidate.

4. Mandatory Block missing.

5. Mandatory Group missing.

6. Mandatory Group missing.

Order to Cash: It is normally refer to business

process for receiving and processing customer sales.

1. Customer order documentation

2. Order Fulfillment

3. Order Delivery.

4. Invoice.

5. Payment Order.

Procure to pay: It is a process of obtaining for management of raw material for manufacturing the product.

1. Determination of requirements

2. Source determination. (6. Goods Receipt.)

3. Vendor Selection. (7. Invoice Verification.)

4. P.O. Processing

5. Order monitoring.

GET Function:

This function enables us to access individual components of a Date Time variable.

Valid date time components are:

- * Years
- * Months
- * Days
- * Hours
- * Minutes
- * Seconds.

Syntax:

integer-variable = get datetime - Component
(datetime-variable)

SET Function:

This func enables us to define individual components of a Date Time variable.

Valid date time component are:

- * Years
- * Months
- * Days
- * Hours
- * Minutes
- * Seconds.

Syntax:

Set datetime - Component (datetime-variable)
integer-variable;

integer-variable;

Code list :

: function L

- * It is a key value pair.
- * Based on keys we will get values. NAV
- * We have to create the huge no. of data.
- * we can get it into any field. → 73
- * Code list are 2 types.
 - Internal Code list.
 - External Code list.
- * first, we have to create a code list and we have to deploy it into dashboard then we can use it.

Syntax :

Select age into str from codelist where

name = "miracle" and Receiver id;

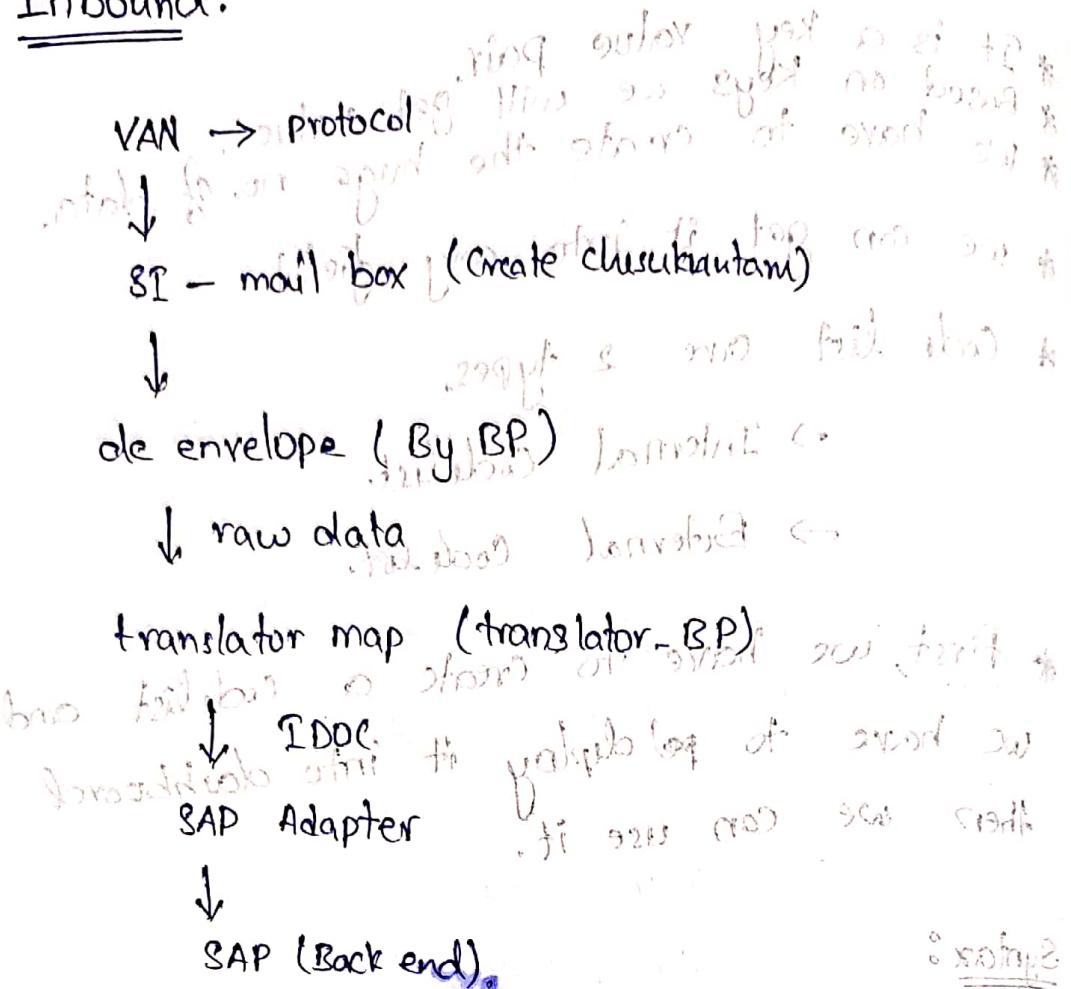
Select Sender Id/Receiver Id into str from codelist where

name = "codelist name" and SenderId/ReceiverId;

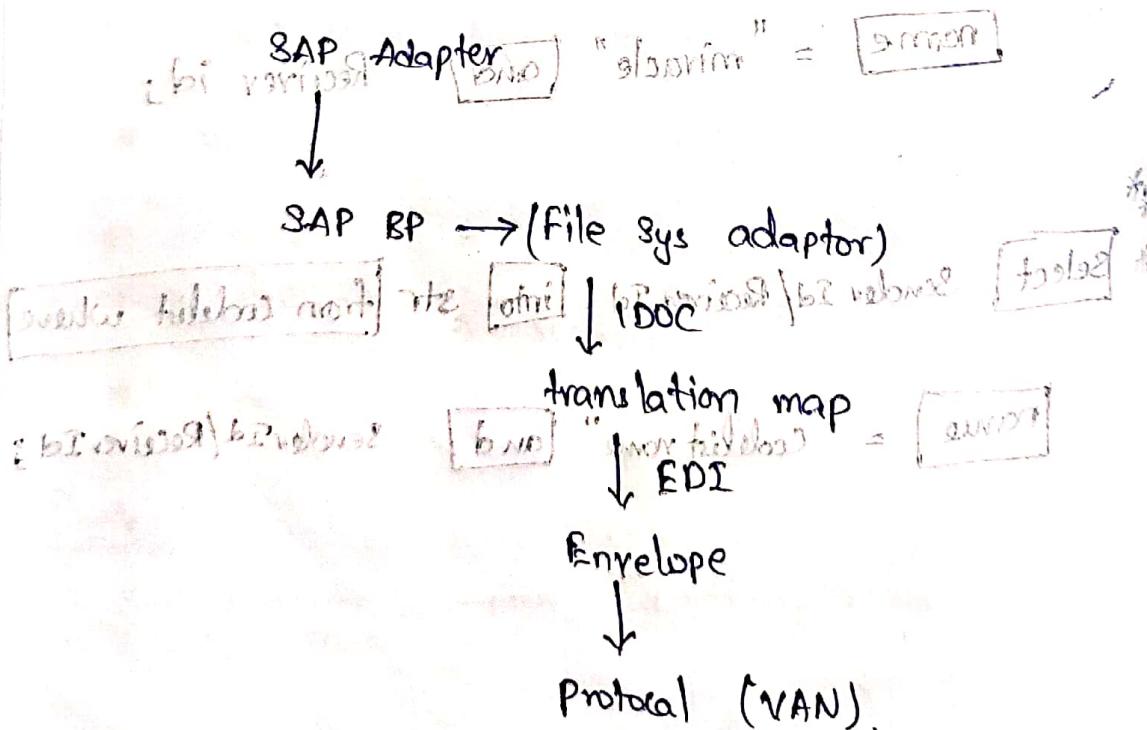
query result

(UAV) location

Inbound:



Outbound:



AS2 Cong b/w Org & Trading Partner
(3rd time) : Creation of Org.

- * S.I Dashboard → Trading partner → System → Create a self signed certificate → GO → Identity → Name = Bhaskar-1
Organization name = Miracle.
→ Specification → Confirm → finish.

- * System certificate → Search → Bhaskar-1 → check out
→ download the certificate → Base → GO.

2nd time: Creation of partner.
→ Digital certificates → System → Create a self signed certificate with partner name and organization → next

Name - Sai-1
Organization - Miracle
Partner. Crf
→ Specification → Serial No - 1 → next → finish.
Duration day - 30.

- * System certificate → Search → Sai-1 → check out →
→ download the certificate → Base 64 → GO.

- * Trusted → new certificate → GO → filename to upload the certificate (Bhaskar-1) → next → naming - Certif. Name - Bhaskar-1
→ next → finish

- * Trusted → new certificate → GO → filename to upload the certificate (Sai-1) → next → naming - Certif. Name - Sai-1
→ next → finish

Go to AS2 → Create AS2 partner (or) Org → Org

→ Create new Identity → new profile - Identity Name:- **Bhaskar**
AS2 Identifier :- Identifiers

Profile Name - Koppineedi

Wish to have a bank's Exchange Cert - Bhaskar-1 → Save

Signing cert = Bhaskar - 1 → Save

→ finish.

1-1082 1 mated ♀

→ Create AS2 partner (or) org → partner → Create new Identity

→ Identity Name :- Sci-1

AS2 identifier : Identifier 1

→ Profile Name: Ammolopanwas [Koppineedi]

Final point is: <http://192.168.51.813/a82>

Response Time out - 30s IX 942 of (108, 373)

శ్రీ పట్టణ గీత శబ్దాలు అనుమతి లేకపై ఉండ్చుకోవాలి.

Key Certificate - Sai-1 → Save

OA Confidential *

performed according to the following procedure:

* Messages → Exchange Certif - Sai-1 → Save

Report of the Board of Appeals dated 18th of June

↳ (536) Self (also signing) Config - [Sci-1] → if save

With spite of 1,920 in 2008 (abg) ending → finish.

The next morning I took a short walk around the lake.

* AS2 trading relationship →

Organization : U - Bhaskar [koppineedi]

Partner : Sai - I [koppineedi]

→ next → next → next → next → Finish

* Search Relationships →

Organization : I - Bhaskar

Partner : Sai - I

→ Go → Test → Pass → Return.

I - IIS : sample application

B2B Landscape

SI pushes the data through HTTP Protocol (IDOC-XI) for 855, 865, 810, 856, and Flat-XML for 850, 860, 830, 862, 846, 861) to SAP XI System.

* SAP XI System and SAP SCM sends the data to SAP SCM System and it is to ECC System.

Maxim business receives the data in various formats from Trading Partner and it converts FTP excel to SI. EDW system used forecasting purpose and it sends the POS (Point of sale) files (867) & Inventory Inquiries (846). BODS is used to store the EDW database tables and it is an EDW SAP System.

Currency. Usually ISO 4217 Code.

CUR segment, CUR=02 | CUX seg, CUX-6345. | EDI CUR

Reference

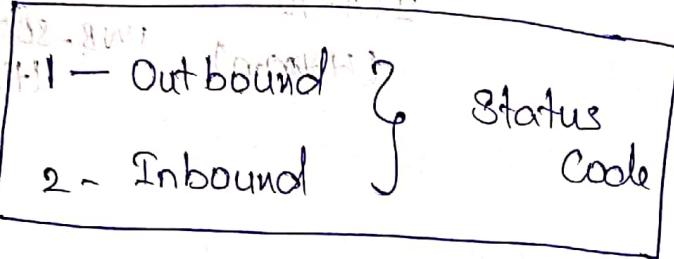
RBF Seg

REF grp (REF) | (REF+DTM) | EDI record

EDI: 850 :

IDOC or Basic type - ORDERS05.

message type - ORDERS.



004010 - using Version

005010 - Latest Version

EDI 810 : IDOC (or) Basic type - INVOICE01

message type - INVOICE.

EDI 812 :

IDOC (or) Basic type - PEXR2001 & PEXR2002.

message type - CREADV (or) DEBADV.

EDI Code 820:

EDI Code 820 is for payment order and credit advice.

Payment Order { IDOC type - PEXR2001 & PEXR2002.
msg type - PAYEXT & REMADV.

Credit { IDOC type - ASVERFO1
advice { msg type - ASVBRF.

EDI 830 [Delivery schedule].

msg type - DELFOR1
IDOC type - DELFOR01.

EDI 855 (also 865).

It is for ordering or order response.

msg type - ORDERSP1

IDOC type - ORDERSO1 to ORDERSO5.

EDI 865:

It is for ordering or order response.

msg type - ORDERSP1

IDOC type - ORDERSO1 to ORDERSO5.

EDI 856:

EDI 856 is for transport and shipping notification (ASN - Advanced ship Notification).

- * For transport in the SAP application.
 - The logical msg is SHPMNT or SHPADV.
 - The IDOC types - SHPMNT01 to SHPMNT03.
- * For delivery in the SAP application.
 - msg type - DBSADV.
 - IDOC type - DELVRY01 & DBLVRY02.

EDI 860: (also 876).

It is for Ordering or order modification

msg type - ORDCHg (same as EDI 860)

IDOC type - ORDER01 to ORDER05.

EDI 862:

It is for delivery schedule

msg type - DELINS or DELJIT

IDOC type - DELFOR01.

EDI 997

Functional Acknowledgement.

* 22 shipping OK, ≤ 100 m shipping bags

* [16:11] Functional Acknow (positive)

* 17 " shift on verb (negative)

* 14 - 15 exchanging ACK;

BDI 846

Transfer of stock and sales data

msg type - PROACT.

~~PROACTO~~ PROACTO 1.

EDIFACT → **INVRPT**

EDT 867:

Point of sales report.

msg type - SLSRPT

EPI 876

It is for ordering or order modification.

msg type → ORDCHG

Type - ORDER S01 to ORDER S05.