IBM Software Group

IBM WebSphere® Data Interchange V3.3

Data Routing

@business on demand.

© 2007 IBM Corporation

This presentation describes data routing capabilities of IBM WebSphere Data Interchange.

Agenda

- Routing in WebSphere Data Interchange (WDI)

- Profile relationships

- Command Chaining

Data Routing

© 2007 IBM Corporation

The presentation will review routing within WebSphere Data Interchange (WDI), WDI profile relationships, and command chaining.

Routing

EDI Input

EDI data can be bundled coming from a value added network (VAN) mailbox or file.

WDI has the ability to separate the resulting record format data into files of a specific format. .

3

WDI has several techniques for splitting transformed data. Electronic data interchange (EDI) data can be bundled coming from a value added network (VAN) mailbox or file. For example, input from multiple trading partners and multiple transactions can be present in the same input.
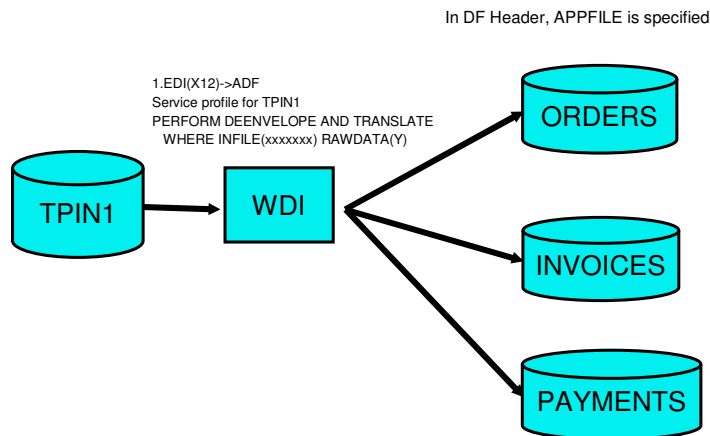
WDI has the ability to separate the resulting record format data into files of a specific format. It also as a special format, called C&D records, that allow record data of different formats to coexist in a single physical file.

IBM Software Group

Multiple Data Format outputs

```
isa
gs
st* 850  - map1 adf(orders)
se
ge
gs
st*810  - map2 adf(invoices)          In DF Header, APPFILE is specified
se
ge
gs
st*820  - map3 adf(payments)
se
ge
iea
```

1.EDI(X12)->ADF
Service profile for TPIN1
PERFORM DEENVELOPE AND TRANSLATE
   WHERE INFILE(xxxxxxx) RAWDATA(Y)

TPIN1 → WDI → ORDERS, INVOICES, PAYMENTS

see footnote

In this example, a single EDI file (or MQ message) is received. The input contains one or more EDI Envelopes, each of which contains one or more groups, and in turn each group contains multiple transactions.

The translator (process engine) is a context based tool which determines the map needed to translate the transaction presented. As data changes from one transaction to the next, WDI changes to the appropriate map to transform the data.

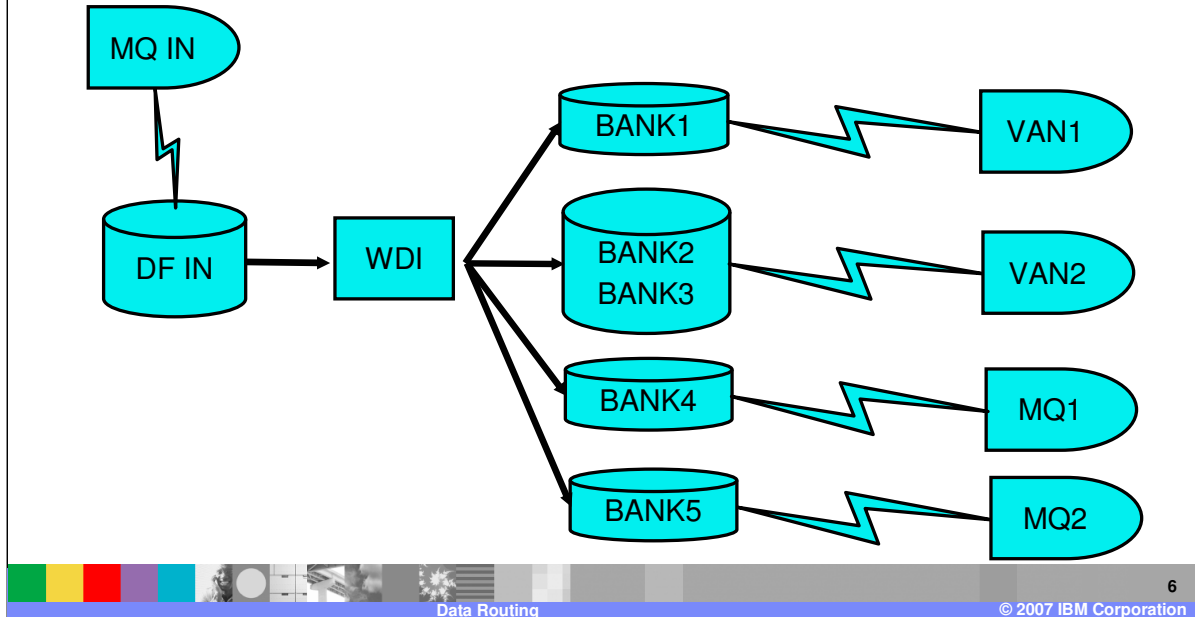The logical name of the file into which data of a specific format is contained in the data

Routing

- Record Oriented Input – Raw data

  - Ability to route the data to specific trading partners, or specific networks - depending on the customers choice.

  - Data for trading partners using a VAN are grouped together.

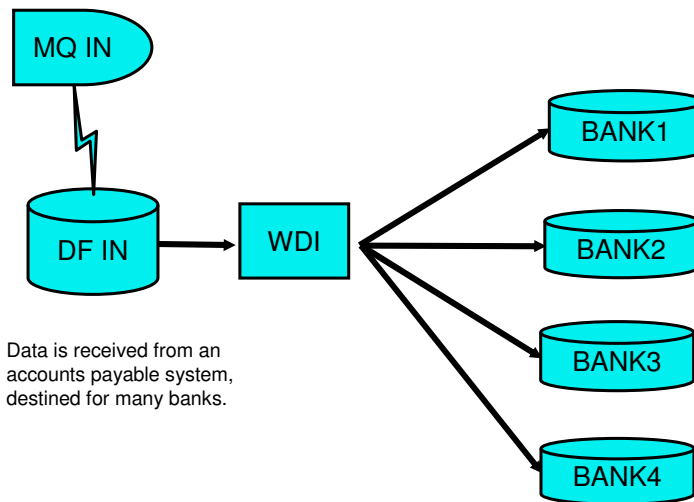 - Data can be separated by Trading Partner and sent using WMQ. .

5

When WDI receives record format data, also called raw data, it has the ability to route the data to specific trading partners, or specific networks depending on your choice. If you are sending to a value added network (VAN), then data for trading partners using that VAN are grouped together. If you are using WebSphere MQ (WMQ) to send to each trading partner individually, then data can be separated by Trading Parnter and sent using WMQ. Data can also be presented to WebSphere Partner Gateway (WebSphere Partner Gateway) for other methods. The gateway is accessed using WMQ in these cases.

Routing by Network


```
MQ IN

DF IN  →  WDI  →  BANK1 ──⚡── VAN1
                   BANK2
                   BANK3 ──⚡── VAN2
                   BANK4 ──⚡── MQ1
                   BANK5 ──⚡── MQ2
```

In this example, data is received from an accounts payable system, destined for many banks.

Routing by Trading Partner

MQ IN

DF IN → WDI → BANK1, BANK2, BANK3, BANK4

Data is received from an accounts payable system, destined for many banks.

A batch of Application data is input to the translator. The data is translated to EDI and separated by Trading Partner, such as a Bank.

Profile Relationships

- Mailbox profile defines the network

- Trading Partner Profile defines the network preferred by the Trading Partner

- Network Profile defines the Network to be used

- Queue Profile defines the WMQ Queue to be used

- Network Commands define the language interface for VAN connection

The definition of the destination from which data is received or to which it is written is defined with a series of administrative profiles. The Mailbox profile defines the network preferred by a requester during a Receive or Input process. The Trading Partner Profile defines the network preferred by the TP during a Send or Output process. The Network Profile defines the Network to be used and the programs for communicating with the Network. The Queue Profile defines the WebSphere MQ (WMQ) Queue to be used. Network Commands define the language interface for value added network (VAN) connection, in IBM these are the Expedite commands necessary to communicate with Information Exchange.

Queue Profile Relationship

Normally used in RECEIVE
operations

MAILBOX
profile

Network ID

Normally used in SEND
operations

TRADING
PARTNER
profile

Network ID
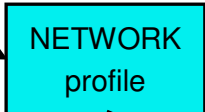
NETWORK
profile

Communication Routine
    VANIMQ
Network Parameters
    points to Queue Profile
    Send - SENDMQ
    Receive - RECEIVEMQ
    Replyto - REPLYTO

QUEUE
profile

Full Queue name
Queue Manager name

WMQ
Queue

Data Routing

9

© 2007 IBM Corporation
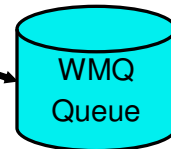
This diagram shows how the various WDI profiles relate to each other and to WMQ.
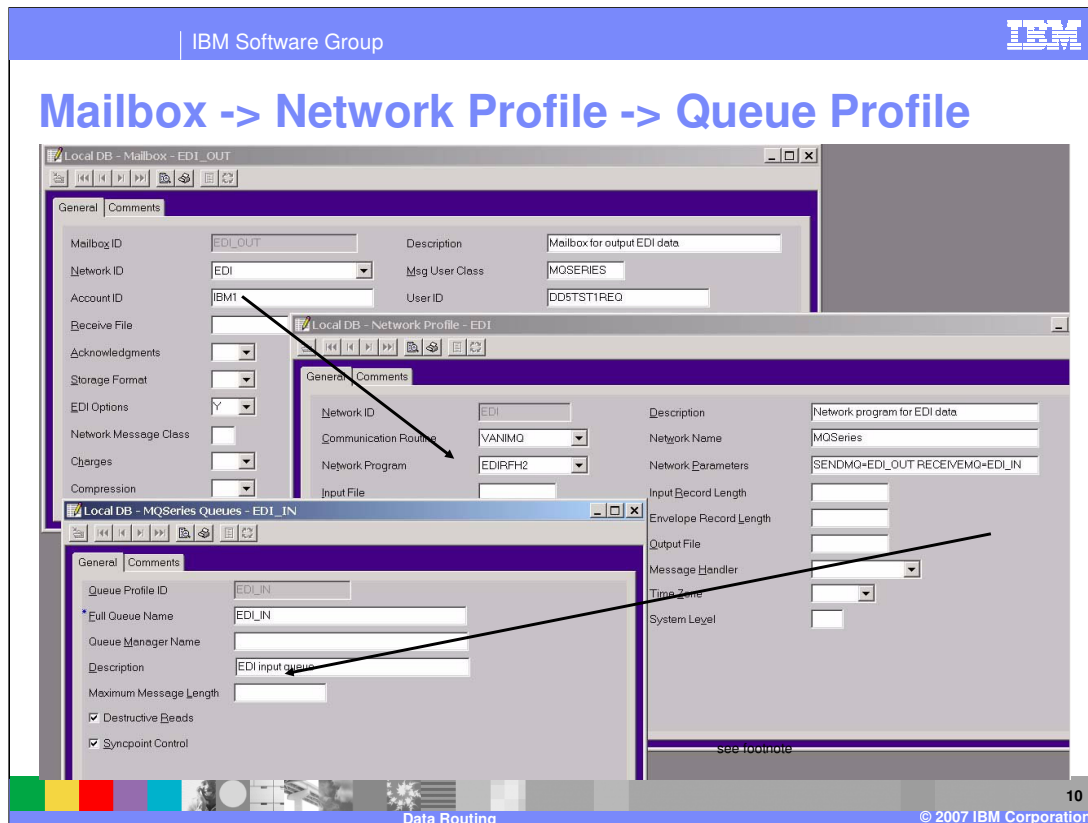
The Mailbox profile - Network ID field - identifies the Network profile which should be used to receive data. The RECEIVEMQ=EDI_IN value in the Network Parameters field identifies the Queue profile to be used. The Queue Profile identifies the name of the WMQ queue which should be read.

When presented with the name of the Mailbox profile in the REQID parameter on a PERFORM command, WDI will connect and read a WMQ message, place it into a file, and then call the translator to process the data.

Command Chaining

- Command chaining allows the output of one command to be chained to a second command.
  - ‣ Double translation
  - ‣ Combining commands

- Service profiles are used to identify the sequence of the commands

- The chain is terminated when no Service profile exists

11

Command chaining allow the output of one command to be chained to a second command. This allows for techniques such as double translation, required to go from XML->EDI with the Send/Receive translator, or combination commands with the PERFORM TRANSFORM to couple transmission with a translation, for example TRANSFORM followed by a SEND command. Service profiles are used to identify the sequence of the commands. The key for the chain is the logical name of the output file. The chain is terminated when no Service profile exists, no output file is created, or the return code of a command indicates an error.

Command Chaining Example

The logical name of the output file is shown in this field

Local DB - Data Format - DUTCHTAXADF DUTCHEMP

General | Details | Overview | Raw Data | Comments

Data Format Name: DUTCHEMP   Where Used

Dictionary Name: DUTCHTAXADF

Description: Dutch employee record

Format
Record ID Information: DUTCHRID
Record Delimiter: Carriage Return/Line Feed

Field Format
○ Fixed Length   ○ Delimited

Delimiter Information
○ Comma   ○ Tab   ○ Semicolon   ○ Other
Text Qualifier:

Code Page: Character Size: 1 Byte

Document Destination
Type:
Name:

This is the General tab for a Data Format definition. Data Formats are located in the WDI Client Data Formats functional area. The logical name of the output file is shown in the fields for Document Destination. Blanks in the Type field indicate a file is being used rather than a WMQ or a CICS output type.

Command Chaining Example

EDI Data

RECEIVE

EDI file

EDI->ADF

DEENVELOPE
AND
TRANSLATE

ADF->XML
TRANSLATE
AND
SEND

DF1 file

DF2 file

DF3 file

SENDFILE -
Acknowledgement

Output
Queue 1

see footnote
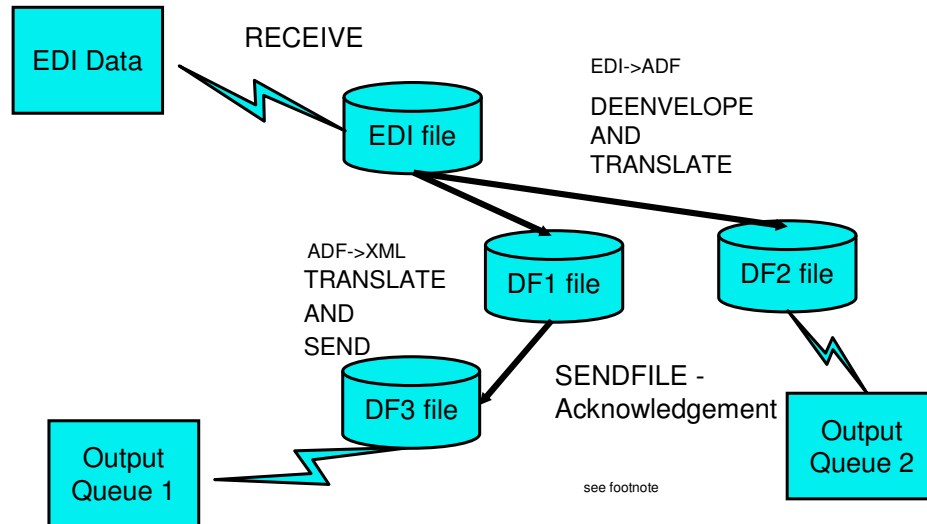
Output
Queue 2

This is a common use for chaining particularly to send the generated Functional Acknowledgement in case of failure when other outputs are rolled back.

In this example, EDI data is received from a queue, the subsequent RECEIVE AND PROCESS creates a file with a logical name of EDIFILE.

The EDIFILE Service Profile specifies a DEENVELOPE AND TRANSLATE command. The results of the translation creates two outputs, DF1 and DF2.   The DF1 Service profile has a command to chain the data in the DF1 file as input to a TRANSLATE AND SEND, this creates the
DF3 file

Chaining Example 2

Input Data

```
isa
gs
st* 850  - map1  adf(outfil01)
se
ge
gs
st*810   - map2 adf(outfil02)
se
ge
gs
st*820   - map3  adf(outfil03)
se
ge
iea
```

MQA

3.
Service profile for OUTFIL01
PERFORM SENDFILE

2.EDI(X12)->ADF
Service profile for TPIN1
PERFORM DEENVELOPE AND TRANSLATE

OUTFIL01 — MQ1

TPIN1 → editr

4a.
ADF->EDIFACT

OUTFIL02 → OUTFIL05 → MQ2

1.
PERFORM RECEIVE AND PROCESS
4. ADF->XML
Service profile for OUTFIL03
PERFORM TRANSLATE AND ENVELOPE WHERE
    APPFILE(OUTFIL03)   FILEID(OUTFIL04)
    TPID(AIDA) RECOVERY(T) RAWTEST(Y) DUPENV(Y)
    RAWFMTID(HI270V41T2)   &FORMAT
....

5.
Service profile for OUTFIL04 and OUTFIL05
PERFORM SENDFILE
PERFORM SEND

OUTFIL03 → OUTFIL04 → MQ3

1. The WDI Adapter issues a PERFORM RECEIVE AND PROCESS with a REQID of TPIN1. MQ Message is read into TPIN1 file which contains the X12 EDI data (ISA) .

2. The TPIN1 Service Profile has a PERFORM DEENVELOPE AND TRANSLATE command, which translates the ED data in TPIN1 to OUTFILE01, OUTFILE02, and OUTFILE03. The different files are required because the 850 Purchase Orders have a different record layout than the 810 Invoices and the 820 Payments.

3. The OUTFIL01 Service profile has

IBM

Trademarks, copyrights, and disclaimers