

# WebSphere Data Interchange v3.2 Data Transformation Transaction Store Implementation Guide

## Data Transformation Transaction Store CSD Overview

WebSphere Data Interchange(WDI) development has created this CSD in response to the immediate demand to support Transaction Store functions with Data Transformation processing.

This document is not a formal IBM publication. It is a technical document written by WDI development to help customers implement Data Transformation Transaction Store functions using this CSD.

### WDI Utility Interface Introduction

The WebSphere Data Interchange Utility provides command-level access to WebSphere Data Interchange services. The service categories discussed in this document are listed below:

- Performing general data translation. Data Transformation processing.
- Processing outbound EDI documents. Translating to EDI standard format using a send map, Enveloping, and Sending.
- Processing inbound EDI documents. Receiving, Deenveloping, and Translating to data format using a receive map.
- Managing data. Updating, removing, and status updates for transaction store records.
- Reporting and extracting data. Formatting and printing reports. Extracting data.

### Data Transformation processing:

Translating data from any EDI, XML, or data format to any other EDI, XML, or data format using a data transformation map. Any-to-any translation is a WebSphere Data Interchange feature that allows you to translate data from any supported source document type to any supported target document type. Supported document types include data formats, EDI standards, and XML data. The Utility command PERFORM TRANSFORM is used for any-to-any translation. The Transaction Store CSD addresses Data Management and reporting with data transformation processing. **NOTE:** The new transaction store functionality is described in this document.

### Outbound and Inbound EDI processing:

The outbound and inbound EDI processing services were used prior to the data transformation processing services. Data management and reporting functions are included in all currently supported WDI releases. Documentation on data management and Transaction Store may be found in WebSphere Data Interchange for z/OS Administration Guide v3.2 and WebSphere Data Interchange Programmer's Reference.

## Managing data and Reporting:

The *Transaction Store* is a collection of electronic data interchange transaction images and the control information that is needed to track the progress of transactions. Transactions enter the Transaction Store when they are translated for sending or when they are deenveloped after being received. All images are in standard format, without envelope header and trailer segments.

The Transaction Store Facility z/OS is an interactive interface to the transaction store services. With these services, you can:

- Envelope transactions
- Send transactions
- Receive and deenvelope transactions
- Translate received transactions to application, XML or standard format
- View or print transaction status information and functional acknowledgments
- View or print envelope status information and functional acknowledgments
- Retrieve acknowledgments from the network to update the network status of transactions
- Update the Transaction Store status by transaction, functional group, or interchange envelope

WDI Client also provides transaction store view and print capabilities for reporting purposes.

As an alternative to the WDI Client and Transaction Store Facility z/OS, you can use the WebSphere Data Interchange Utility, the command-level interface to WebSphere Data Interchange services.

## Installation

### Updating the WebSphere Data Interchange database

As part of the transaction store changes for CSD3 some new columns have been added to one of the tables in the WebSphere Data Interchange database.

If you are doing the initial set up of your database **after** installing CSD3, then you do not need to take any special action. You can just follow the steps in the WebSphere Data Interchange for Multiplatforms User's Guide.

If you are installing CSD3 and **already have** an existing database for WebSphere Data Interchange, then you need to update it by doing the following steps.

Note: If you have not previously installed CSD1, you must make the database updates associated with CSD1 before making the CSD3 database changes. The steps for CSD1 are outlined in the "Special installation instructions for CSD1" in the CSD1 details section of the **readme.txt** file. If you have installed CSD1, but have not installed CSD2, you do not need to do any special CSD2 updates, and can just follow the steps below.



Perform the following steps as a user with DB2 administrator authority.

1. If you are installing on Windows, using your DB2 administrator user ID, select Start -> Programs -> IBM DB2 -> Command Window to open the DB2 Command window. If you are installing on AIX, log in as a user with administrator authority. You should use the same id that was initially used to create the EDIEC32E database.

The remaining database setup steps use this command window or login session.

2. Change to the ddl directory under the installation directory.

Issue the following command:

```
db2 -tf csd3.ddl -l csd3.log
```

This adds new columns to the EDITSTH transaction store table and updates the views associated with that table.

3. Change to the ixf directory under the installation directory.

Issue the following command:

```
loadmsgs
```

This updates the message text for error and other messages that are written to the print file.

For AIX only, issue the additional command:

```
db2 -tf loadcsd3.ddl -l loadcsd3.log
```

This replaces the 20 rows in the Profile Definition table, EDIPSPD, correcting a problem noted with the E Envelope Profile.

4. Change to the bind directory under the installation directory.

Invoke the command to rebind the WebSphere Data Interchange DB2 packages.

```
db2 -tf bindgrnt.fil -l bind.log
```

You have now finished updating the WebSphere Data Interchange database for CSD3.

## Data Transformation Transaction Store Implementation

Data Management and reporting for Data Transformation processing uses the existing Transaction Store interfaces. The Transaction Store Facility is supported on the z/OS platform. For data transformation, the facility functions supported includes reporting functions and status update. The WDI Utility interface is supported for all platforms and includes the primary transaction store functions.

The primary Transaction Store functions for EDI messages that are translated using the PERFORM TRANSFORM processing include:

- Deferred or Delayed translation - Develope an inbound EDI transaction with one command, then translate it later. This is similar to the existing receive processing commands PERFORM DEENVELOPE with a separate PERFORM TRANSLATE TO APPLICATION.
- Deferred or Delayed enveloping - Translate a document to create an EDI transaction with one command, then envelope it at a later time. This is similar to the existing send processing commands PERFORM TRANSLATE TO STANDARD with a separate PERFORM ENVELOPE.
- Status Tracking/reporting capability - The existing transaction store interface (WDI Client, z/OS Administrator's Menu and Utility PERFORM commands) will be used to report the status of transactions in the transaction store. Documentation on WDI Transaction Store may be found in WebSphere Data Interchange for z/OS Administration Guide v3.2 and WebSphere Data Interchange Programmer's Reference.
- Duplicate envelope detection - Optional checking to see if inbound EDI envelopes have already been received. If a duplicate exists, appropriate Functional Acknowledgments will be generated. Duplicate envelopes may be reported with the CONTRL Functional Acknowledgment.

### Activating Transaction Store

The WDI Client Application Defaults profile identifies your business applications to WebSphere Data Interchange and controls certain processing features. An application defaults profile mainly determines whether copies of transactions are stored in WebSphere Data Interchange's Transaction Store database and the log files in which processing messages are stored.

When an application invokes WebSphere Data Interchange, it can provide an application ID (APPLID). WebSphere Data Interchange then searches for an application defaults profile to match the application ID. The default application defaults profile is EDIFFS. For WDI z/OS, application ID can be provided as an invocation parameter **APPLID = value**. For AIX/WINDOWS, application ID can be provided with a **SET APPLID(value)** in the STDIN command file or using the **SetAppl(char\* pszVal)** class with the WDI C++ API.

There are three flags in the application defaults profile that control transaction store activity. These flags may contain the Values: Y(es), N(o), E(rror), S(uccessful). Y(es) specifies transaction store is active, N(o) specifies transaction store is inactive, E(rror) specifies that only translations with errors are written to the store, and S(uccessful) specifies that only successful translation are written to the store. The following application defaults profile flags control transaction store activity:

- Transaction
- Transaction Image
- Func. Ack Image

The Transaction Store flag must be a value other than N for transaction store to be active. If the Transaction Store flag value is N, the store is inactive and the image flag values are ignored.

The Transaction Image flag specifies if inbound or outbound transaction images are to be written to the store. This includes inbound functional acknowledgment images.

The Functional Ack. Image flag specifies if outbound functional acknowledgment images are to be written to the store. A value of E(rror) specifies the outbound functional acknowledgment images are written to the store only for inbound transactions that fail standard compliance checks. A value of S(uccessful) specifies the outbound functional acknowledgment images are written to the store only for inbound transactions that passed standard compliance validation successfully. This flag does not control transaction store activity for inbound functional acknowledgment images. Inbound functional acknowledgment images are controlled by the Transaction Image flag.

Only transactions that are translated to or from EDI data may be written to the Transaction Store. The EDI images may be written to the store for these transactions. Data Format(DF) or XML images will not be written to the store for these transactions. Other translations that do not involve EDI, such as XML-DF, DF-DF, and XML-XML, will not be written to the Transaction Store.

## WDI Utility Interface

### New keywords for PERFORM TRANSFORM

The following keywords have been added to the PERFORM TRANSFORM command:

- **BATCHSET**

User-defined ID assigned to the transaction. This is used the same way as the current BATCHSET keyword.

- **DEENVONLY(Y/N)**

Default is N. If set to Y, this means the data will only be deenveloped and put in transaction store, and will not be translated. (Functionally equivalent to PERFORM DEENVELOPE.)

- **DIR(S/R)**

This keyword is required if INTYPE(ST) is specified. It indicates whether source EDI transactions (DIR=R for “Receive”) or target EDI transactions (DIR=S for “send”) are to be read from transaction store. Only transactions that were processed by a previous PERFORM TRANSFORM command will be retrieved.

Transactions that were placed in the store by a PERFORM DEENVELOPE or PERFORM TRANSLATE TO STANDARD command would not be retrieved for the PERFORM TRANSFORM processing.

Transactions that were added to the store using the DT processing will be identified by translate type “DT”.

- **DUPENV(Y/N)**

Default is Y. Indicates if duplicate inbound envelopes should be processed. If N, duplicate inbound envelopes cause a severity 8 validation error. If Y, duplicates are processed normally. This is used the same way as the current DUPENV keyword.

- **EENVDATE**

Earliest envelope date (only used for outbound EDI). This is used the same way as the current EENVDATE keyword.

- **ENVELOPE(Y/N)**

New keyword. Default is Y. Indicates if the EDI data should be enveloped (only used for outbound EDI). If N, the transactions are just added to the store for later enveloping.

- **FADELAY(Y/N)**

Default is N. This is functionally equivalent to the current FADELAY keyword. If set to Y, then the Enveloper does not envelope the Functional acknowledgments (FAs). Instead, the FAs are just added to the store for later enveloping.

- **INTYPE**

New value “ST”. This value indicates that the data is read from transaction store instead of from a file or queue and will override the INFILE keyword. The DIR keyword is required when using this keyword.

- **PURGINT**

Number of days in TS before marked for purge. This is used the same way as the current PURGINT keyword

- **ONEMSG**

Indicates whether all MQSeries messages are read from an MQSeries queue or only one message is read from the queue. The queue can be either a receive file or an application send file. Applies only to MQ. An MQSeries message is defined as a logical set of MQ records with the same MSGID. This keyword also controls MQ message descriptor propagation when you can set it on the TRANSLATE AND SEND and RECEIVE AND TRANSLATE commands where the application file and the send or receive file are both MQ queues. If you want MQ message descriptors to propagate during processing, set this keyword to Y. Valid values are:

Y Reads one MQSeries message (one MSGID value) from an MQSeries queue  
N Reads all MQSeries messages from an MQSeries queue (default)

**NOTE: This keyword is not directly related to Transaction Store but has been made available.**

- **RECOVERY(T/E)**

Indicates the unit of work. The default is environmentally dependent. Valid values are:

E Issues a database commit after each envelope (default for CICS)  
T Issues a database commit after each transaction (default for z/OS and MP)

## **Delayed Translation and Enveloping**

EDI target - Outbound EDI processing includes the following functions:

- Translating application, XML, or EDI data into an EDI standard format and placing it in the Transaction Store
- Enveloping standard transactions or messages so they are ready to be sent (**Delayed Enveloping**)
- Sending enveloped data to trading partners

EDI source - Inbound EDI processing includes the following functions:

- Receiving data from trading partners
- Deenveloping interchanges and placing the EDI standard transactions or messages into the Transaction Store
- Translating EDI standard transactions or messages into application, XML, or EDI formats (**Delayed Translation**)

Commands are supplied to do each of these three steps independently or in combination.





Below are some Utility command examples of Outbound/Inbound processing for send/receive maps and the corresponding PERFORM TRANSFORM processing for any-to-any maps:

***OUTBOUND EDI Target Delayed Enveloping:***

**PERFORM TRANSFORM WHERE .... ENVELOPE(N)**

Similar to PERFORM TRANSLATE TO STANDARD.

Data will be transformed, but outbound EDI data will not be enveloped. It will only be written to the Transaction Store.

**PERFORM TRANSFORM WHERE .... INTYPE(ST) DIR(S)**

Similar to PERFORM ENVELOPE.

Outbound EDI data will be retrieved from the Store based on Transaction Store selection criteria, and enveloped. Only transactions from the DT translator will be retrieved.

***OUTBOUND EDI Target Combination (Translate and Envelope):***

**PERFORM TRANSFORM WHERE .... ENVELOPE(Y)**

Similar to PERFORM TRANSLATE AND ENVELOPE.

This is the same as the current data transformation processing.

***INBOUND EDI Source Delayed Translation:***

**PERFORM TRANSFORM WHERE .... DEENVONLY(Y)**

Similar to PERFORM DEENVELOPE

The inbound EDI data is deenveloped, validated, and a functional acknowledgment is generated (if requested), but the data is not translated. The EDI transactions are written to the Store for later processing.

**PERFORM TRANSFORM WHERE .... INTYPE(ST) DIR(R)**

Similar to PERFORM TRANSLATE TO APPLICATION.

Inbound EDI data that has been deenveloped but not translated will be retrieved from the Store based on Transaction Store selection criteria and translated. Only transactions from the DT processing will be retrieved.

### ***INBOUND EDI Source Combination (Deenvelope and Translate):***

#### **PERFORM TRANSFORM WHERE .... DEENVONLY(N)**

Similar to PERFORM DEENVELOPE AND TRANSLATE.

This is the same as the current data transformation processing.

When **INTYPE(ST)** is specified with PERFORM TRANSFORM, the following selection criteria keywords also apply:

ACFIELD	- Application control field
APPLID	- Application ID
BATCH	- Batch ID assigned via BATCHSET
DIR	- Direction. Valid values: 'S' (Send/Target) and 'R' (Receive/Source)
EPURDATE	- Date transaction is to be purged from TS
FORMAT	- Application data format of the transaction
HANDLE	- Transaction Store key of the transaction
NETID	- Network profile ID
STDTRID	- Standard transaction ID
TPNICKN	- Trading partner nickname
TRXDATE	- Date transaction added to TS
TRXSTAT	- Transaction store status
TRXTIME	- Time transaction added to TS

### **Re-Processing Data**

The **TRXSTAT** keyword along with **INTYPE(ST)** and **DIR(S/R)** can be used with PERFORM TRANSFORM to indicate reprocessing. A list of values for TRXSTAT keyword can be found in the WebSphere Data Interchange for z/OS Administration Guide v3.2 and WebSphere Data Interchange Programmer's Reference.

### ***OUTBOUND EDI Target:***

#### **PERFORM TRANSFORM WHERE INTYPE(ST) DIR(S) TRXSTAT(30) ENVELOPE(Y)**

Similar to PERFORM REENVELOPE WHERE TRXSTAT(30).

Outbound EDI data that have been previously enveloped (TRXSTAT = 30) will be retrieved from the Store based on Transaction Store selection criteria, and reenveloped. Only transactions from the DT translator will be retrieved.

Additional selection criteria keywords supported (from REENVELOPE command): Note: Selection criteria keyword TRERLVL will not be supported for Data Transformation processing.

APPRECID	- Application receiver ID
APPSNDID	- Application sender ID
ENVDATE	- Date transaction enveloped (outbound only)
ENVTIME	- Time transaction enveloped (outbound only)
ENVTYPE	- Type of envelope used
FUNACKP	- Specifies if FA is pending (outbound only)
GRPCTLNO	- Group control number
INTCTLNO	- Interchange control number
INTRECID	- Interchange receiver ID
INTSNDID	- Interchange sender ID
NETACKP	- Specifies if network ack is pending (outbound only)
NETSTAT	- Network status of the transaction (outbound only)
SNDDATE	- Date transaction sent (outbound only)
SNDTIME	- Time transaction sent (outbound only)
TRXCTLNO	- Transaction control number

### ***INBOUND EDI Source:***

#### **PERFORM TRANSFORM WHERE INTYPE(ST) DIR(R) TRXSTAT(73)**

Similar to PERFORM RETRANSLATE TO APPLICATION.

Inbound EDI data that has been deenveloped and have status receive translate error (TRXSTAT = 73) will be retrieved from the Store based on Transaction Store selection criteria and translated. Only transactions from the DT processing will be retrieved.

## **Managing Data and Reporting**

To assist in data management and reporting, a new Utility selection criteria keyword has been introduced to select transactions that were processed using outbound (send maps) or inbound (receive maps) transactions, transactions that were processed using data transformation maps or both:

**TRANSTYPE** - Values: DT to indicate Data Transformation translations or SR to indicate Send/Receive translations, blank for both

A new field **Transaction Type** has also been added to the WDI Client and z/OS Transaction Store Facility under Transaction Criteria (panel TF03).

The **TRANSTYPE** keyword may be used on the following commands:

ENVELOPE - TRANSTYPE value is always SR  
ENVELOPE AND SEND - TRANSTYPE value is always SR  
ENVELOPE DATA EXTRACT  
HOLD  
PRINT ACKNOWLEDGMENT IMAGE  
PRINT ACTIVITY SUMMARY  
PRINT EVENT LOG  
PRINT STATUS SUMMARY  
PRINT STATUS SUMMARY2  
PRINT TRANSACTION DETAILS  
PRINT TRANSACTION IMAGE  
PURGE  
QUERY  
REENVELOPE - TRANSTYPE value is always SR  
REENVELOPE AND SEND - TRANSTYPE value is always SR  
RELEASE  
REMOVE TRANSACTIONS  
RETRANSLATE TO APPLICATION - TRANSTYPE value is always SR  
TRANSACTION DATA EXTRACT  
TRANSLATE TO APPLICATION - TRANSTYPE value is always SR  
UNPURGE

**Data management includes the following functions:**

- Rebuilding interchanges from transactions in the Transaction Store.
- Updating status of transactions in the Transaction Store.
- Removing transactions from the Transaction Store.
- Updating management reporting statistics. Not supported for Data Transformation.
- Removing management reporting statistics. Not supported for Data Transformation.

**Examples:**

**PERFORM RECONSTRUCT WHERE TPNICKN(IBM) INTCTLNO(35) DIR(R)**

Reconstruct inbound EDI data with interchange control number 35 that for trading partner IBM were stored in Transaction Store using Data Transformation or Inbound EDI processing.

**NOTE:** TRANSTYPE keyword is not needed for this command since TPNICKN, INTCTLNO, and DIR are required keywords.

## **PERFORM PURGE WHERE INTCTLNO(5) DIR(R) TRANSTYPE(DT)**

Mark for Purge inbound EDI data with interchange control number 5 that were stored in Transaction Store using the Data Transformation process. Only transactions from the DT processing will be retrieved (TRANSTYPE = DT).

### **Data reporting and extraction includes the following functions:**

- Generating formatted reports containing data from the Transaction Store.
- Extracting data from the Transaction Store for further processing outside WebSphere Data Interchange.
- Extracting data from the management reporting statistics tables for further processing outside WebSphere Data Interchange. Not supported for Data Transformation.

WebSphere Data Interchange provides two mechanisms for producing reports:

- Management Reporting data extracts. Not supported for Data Transformation.
- Transaction Store data extracts.

### ***Examples:***

## **PERFORM PRINT ACTIVITY SUMMARY WHERE BATCH(121497) DIR(R) TRANSTYPE(DT)**

Print Transaction Activity Summary for inbound EDI documents with batch ID 121497 that were stored in Transaction Store using the Data Transformation process. Only transactions from the DT processing will be retrieved (TRANSTYPE = DT). Place the results in the RPTFILE file.

## **PERFORM TRANSACTION DATA EXTRACT SELECTING INTERCHANGE(Y), GROUP(Y) TRANSACTION(Y) WHERE TPNICKN(PISCES) DIR(S) INTCTLNO(000008888) TRANSTYPE(DT)**

Retrieve the interchange, group, and transaction information for all outbound transactions with trading partner PISCES and interchange control number 000008888. Place the results in the EDIQUERY file.

Transaction Store Reports that are supported with data transformation:

- Activity Summary
- Acknowledgment Image
- Transaction Details
- Transaction Image
- Status Summary

## Status Summary2

Data Extracts supported with data transformation:

Envelope Data Extract  
Transaction Data Extract

## Considerations

### *INBOUND EDI Source Delayed Translation:*

Some Envelope properties that may have been mapped are not available with delayed inbound translation. These include: All envelope specific properties (for example: ISA04) and the following envelope generic properties:

**IchgSndrQI** Interchange sender qualifier  
**IchgRcvrQI** Interchange receiver qualifier  
**IchgPswd** Interchange password  
**IchgUsgInd** Interchange usage indicator  
**IchgAppRef** Interchange application reference  
**IchgVerRel** Interchange version/release  
**GrpDate** Group date  
**GrpTime** Group time  
**GrpPswd** Group password  
**GrpVer** Group version  
**GrpRel** Group release  
**TrxCode** Transaction code  
**TrxVer** Transaction version  
**TrxRel** Transaction release

Global variables with Interchange or Group scope, that may have been mapped, are not available with delayed inbound translation.

Transactions that are in an “Orphan” state will be filtered and not translated. Orphan entries are usually created when the first transaction of a translation process does not complete normally or for some reason does not have an interchange, group, or image record in the Transaction Store database. These entries cannot be accessed in any way and are only shown when removed.

They are removed whenever the REMOVE TRANSACTIONS command is Executed.

### *OUTBOUND EDI Target Delayed Enveloping:*

Transactions that are in the Transaction Store database but have no images will be filtered and not enveloped.

End of Document