

Cleo ERP Connector for JDE User Guide

Version 1.2



December 2017

RESTRICTED RIGHTS

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (C)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227 - 7013.

Cleo

4949 Harrison Avenue, Suite 200
Rockford, IL 61108 USA
Phone: +1.815.654.8110
Fax: +1.815.654.8294
Email: sales@cleo.com
www.cleo.com

Support

International: +1 815-282-7894
US Toll-free: 1-866-444-CLEO (2536)
UK: 02038653439
Email: pasupport@cleo.com

Cleo reserves the right to, without notice, modify or revise all or part of this document and/or change product features or specifications and shall not be responsible for any loss, cost or damage, including consequential damage, caused by reliance on these materials. This document may not be reproduced, stored in a retrieval system, or transmitted, in whole or in part, in any form or by any means (electronic, mechanical, photo-copied or otherwise) without the prior written permission of Cleo.

© 2017 Cleo. All rights reserved. Cleo is a trademark of Cleo. Cleo Clarify is a registered trademark of Cleo.

All marks used herein are the property of their respective owners.

Contents

- Cleo ERP Connector for JDE..... 5**
 - How the ERP Connector integrates with JD Edwards E1..... 5
 - Supported Transactions 5
 - Supported product versions and other system requirements 7
 - Related Help and Support 7
- What's Included 8**
 - com.cleo.jde.di.baseobjects..... 8
 - com.cleo.jde.di.mypartner 8
- Getting Started 9**
 - Download the ERP Connector contents 9
 - Customize EDI Integrations 9
 - Test and Refine EDI Integrations..... 9
 - Deploy and Run EDI Integrations 10
 - Configure ERP Connector for JDE Integrations with EnterpriseOne 11
- Inbound Mapping..... 13**
 - Inbound 850 Purchase Order Mapping..... 14
 - Inbound 810 Invoice Mapping 18
 - Inbound 820 Payment Order Mapping..... 21
 - Inbound 860 Purchase Order Change Mapping 22
 - Inbound 945 Warehouse Shipping Advice Mapping 24
 - Inbound 867 Product Transfer & Resale Report Mapping 27
 - Inbound 823 Lockbox Mapping 30
 - Inbound 856 Advance Ship Notice Mapping..... 31
- Outbound Mapping..... 33**
 - Outbound 810 Invoice Mapping..... 33
 - Outbound 856 Advance Ship Notice Mapping..... 38
 - Outbound 855 PO Acknowledgement Mapping 41
 - Outbound 940 Warehouse Shipping Order Mapping 44
 - Outbound 850 Purchase Order Mapping..... 47
 - Outbound 846 Inventory Advice Mapping..... 50
 - Outbound 852 Product Activity Mapping 51
- General Recommendations & Best Practices 52**
- Additional Information 54**
 - Identify Numeric Fields with Implied Decimal Positions 54
 - JD Edwards Table Naming Conventions 56
 - Field Types..... 57

General Recommendations & Best Practices 59

Additional Information 61

 Identify Numeric Fields with Implied Decimal Positions 61

 JD Edwards Table Naming Conventions 63

 Field Types 65

Cleo ERP Connector for JDE

The Cleo ERP Connector for JDE provides customizable implementation solutions that can be used to integrate Cleo Clarify and JDE using the JD Edwards EnterpriseOne interface and ASC X12 EDI transactions. The ERP Connector includes a large collection of reusable Clarify resources, such as Rulesets for each of the supported inbound and outbound transactions, sample Business Processes, pre-configured data sources and database adapters (for integration with the staging database used by the JD Edwards system), and source and target Schemas.

The ERP Connector delivers a centralized configuration and control solution for all communications and data integration activity, allowing you to:

- Deliver EDI Order-to-Cash and Procure-to-Pay transactions by customizing the included sample integration projects;
- Create custom business integration projects for other transaction types, from scratch.

How the ERP Connector integrates with JD Edwards E1

JD Edwards EnterpriseOne exposes an EDI interface called the *Data Interface for Electronic Data Interchange*. This interface consists of a set of database tables (identified by the code prefix “F47”) that are used to stage input and output data for the EnterpriseOne system, as illustrated below. Data staging is a safe interface approach, because EnterpriseOne controls the process of importing and exporting data between the F47 tables and its live database.

You can also use Clarify and the ERP Connector to read and write user-defined table content, identified using the code prefixes “F55” through “F59”. For example, you can use these tables for temporary storage of inbound EDI data that must be turned around in subsequent outbound transactions.

Communication with the JDE Data Interface occurs by accessing and updating staging tables directly, or less commonly, through a derived flat file interface. The ERP Connector uses the former approach (but can be customized to use the latter), reading and updating the F47 staging tables directly.

Supported Transactions

The ERP Connector for JDE includes pre-built integrations for common Order-to-Cash transactions. These integrations provide a starting point for creating customized implementations that utilize trading partner-specific EDI documents, customized data transformations and process automations, and partner-specific communication adapters.

Modify these integration projects to create integrations to meet your business needs, or refer to them as examples when creating new kinds of integrations. A complete list of pre-built integrations are as follows:

Inbound Transactions

EDI Transaction	JD Edwards Target Tables
810 Invoice	F47041 / F47042 / F47044 / F4706
820 Payment Order	F47111 / F47112 / F47113 / F47114
823 Lockbox	F47111 / F47112 / F47113 / F47114
850 Purchase Order	F47011 / F47012 / F4717 / F4715 / F4706
856 Advance Ship Notice (S-O-P-I)	F47031 / F47032 / F4706 / F4714
856 Advance Ship Notice (S-O-T-P-I)	F47031 / F47032 / F4706 / F4714
860 Purchase Order Change	F47131 / F47132 / F4706
867 Product Transfer & Resale	F47011 / F47012 / F4714 / F4715 / F4706
945 Warehouse Shipping Advice	F47131 / F47132

Outbound Transactions

EDI Transaction	JD Edwards Source Tables
810 Invoice	F47046 / F47047 / F470461 / F4706 / F4714
850 Purchase Order	F47016 / F47017 / F47065 / F4714 / F4715
852 Product Activity	F47126 / F47127
855 PO Acknowledgement	F47026 / F47027 / F4714
856 Advance Ship Notice S-O-I	F47036 / F470371 / F4706
856 Advance Ship Notice S-O-I-P	F47036 / F470371 / F4706
856 Advance Ship Notice S-O-P-I	F47036 / F470371 / F4706
856 Advance Ship Notice S-O-T-P-I	F47036 / F470371 / F4706
940 Warehouse Shipping Order	F47026 / F47027
846 Inventory Advice	F47126 / F47127

You can implement integrations for additional EDI transactions by using the EDI standards and EDI resources built into the Clarify Studio. These include the ability to generate and customize document Schemas, Adapters, Rulesets, and other integration objects from metadata and other external inputs.

For more information, please refer to the *Designing Integration Solutions* section in the *Cleo Clarify v4 User Guide*.

Supported product versions and other system requirements

This product has been designed and tested to work with Cleo Clarify v4 and above. All references within this document cite this version only. The following versioning and requirements apply:

- Clarify Studio v4 or later must be installed;
- Clarify Server v4 or later must be installed;
- ERP Connector for JDE supports JDE EnterpriseOne*.

* The ERP Connector can be customized to work with JDE World. Please contact Cleo Support with additional questions on these requirements.

JD Edwards Configuration

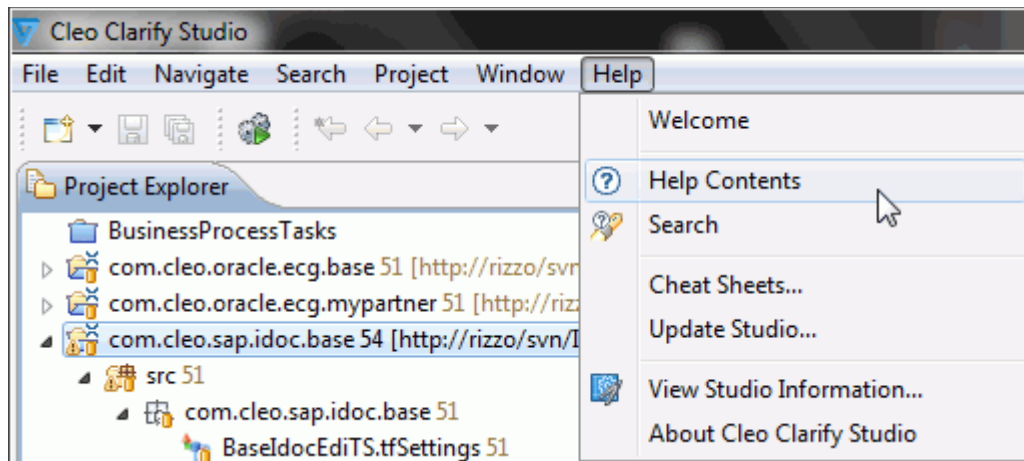
The ERP Connector was built using the following configuration:

- Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 – 64bit Production
- PL/SQL Release 11.2.0.1.0 – Production
- CORE 11.2.0.1.0 Production
- TNS for 64-bit Windows: Version 11.2.0.1.0 – Production
- NLSRTL Version 11.2.0.1.0 – Production

Note: To determine which database your JD Edwards system is using, type `SELECT * FROM V$VERSION` from a SQL command line.

Related Help and Support

This document refers to the Studio's product Help (User Guide) for detailed instructions on how to complete certain tasks that are not described here. To view the Help, select Help Contents from the main menu in your Studio.



What's Included

The ERP Connector contains several Clarify Projects containing multiple packages, objects, sample data, and other resources. These include:

- `com.cleo.jde.di.baseobjects` (Project)
- `com.cleo.jde.di.mypartner` (Project)

Note: The Projects must be set up and configured in each Clarify Studio environment where you will be creating or maintaining integrations for JDE.

`com.cleo.jde.di.baseobjects`

This base Project contains several Packages containing multiple objects. These objects may be referenced, or shared, by other Projects; in this case, the other Project, *com.cleo.jde.di.mypartner*.

Clarify supports sharing the same objects between different Projects, and sharing the same objects and Projects between different users. Shared objects within this Project include:

- Application Interfaces
- Business Processes
- Code Tables
- Control Number Generators
- Events
- File Adapters
- File Monitors
- Formulas
- Global Variables
- Rulesets
- Ruleset Data Objects
- Schemas
- Transformation Settings

`com.cleo.jde.di.mypartner`

This Project contains Packages and objects that are specific to the sample Trading Partner used throughout the sample EDI and JDE transactions within the ERP Connector. This includes inbound and outbound EDI Routes, EDI Envelopers, Rulesets, and Trading Partner objects.

Using the Clarify Project Generator

An important aspect to this Project is the use of the generic “mypartner” in the object names. This facilitates use of the Clarify Project Generator, which allows you to easily replicate a template Project and its resources, updating new resource names, re-configuring enveloping and trading partner info, all while maintaining resource dependencies and other attributes and values to automatically reflect a newly generated Project.

For more information, please refer to the *Wizards and other Tools* topic in the *Clarify v4 User Guide*.

Getting Started

Before you can use this product, it must be set-up and configured correctly. Because the ERP Connector comprises customizable integration objects, it must be set up on each Clarify Studio workspace environment where you plan to create and configure EDI integrations for your JD Edwards system.

Download the ERP Connector contents

The ERP Connector product must be downloaded from the Cleo-hosted SVN repository. If you do not already have access, please contact Support for initial registration and setup.

Once you have access to the hosted SVN repository, follow this process to download the integrator contents.

1. From the Clarify Studio, create a repository location using the *SVN Repository* perspective. **This location URL is provided by Support.**
2. Check out the two JDE Projects from the *ERP Connector* repository location. This can be done by right-clicking the Project, and selecting *Check Out* from the menu. This establishes the necessary resources in your Workspace.
3. Disconnect the Projects from the hosted repository. Select a Project from the *Project Explorer* perspective, right-click, and choose *Team/Disconnect*. When prompted, delete all SVN meta-data. Your Projects can now be shared and committed to your own repository.
4. You must perform the above step for *each* Project.

Customize EDI Integrations

The ERP Connector for JDE consists of pre-built Clarify objects that allow you to build custom integrations to meet the specific EDI integration needs of your business. Creating EDI integrations with the Connector requires an understanding of how to manage Clarify objects using the Clarify Studio.

Once you have configured an EDI integration, you can test, debug, deploy, and monitor its execution using Clarify Studio and the Clarify Dashboard.

Custom Integrations using the ERP Connector for JDE

The following process describes an outline of steps needed to configure new instances of integrations based on the integrations that are included in the ERP Connector product – for example, onboarding a new partner for an included transaction type.

If you plan to configure integrations for EDI transaction types that are not included in the ERP Connector, you will need to create or copy and modify Business Processes, Database Adapters, EDI and Database Schemas, and Rulesets. For more information, please refer to the *Designing Integration Solutions* section in the *Clarify v4 User Guide*.

Test and Refine EDI Integrations

The Clarify Studio provides several tools for testing integrations, verifying execution status and results, troubleshooting execution exceptions, and reporting on execution activity, over time:

- **Local Test Server:** Each Clarify Studio environment supports rapid iterative testing of integrations, as you build and refine them. This feature allows you to manually invoke integrations and verify execution results, before you deploy them to a Test or Production server environment.
- **Business Process View:** The Business Process View is an interactive tool for launching, monitoring, and managing Business Processes.
- **Auditor View:** The Auditor View is an interactive tool for querying, filtering, browsing, and viewing log data produced during execution of Clarify integrations. You can use the Auditor while testing integrations to verify the results of execution and to troubleshoot the nature of runtime exceptions and errors.

For more information on these tools, see *Navigating in Clarify Studio/Clarify Perspectives and their Views/Admin Console in the Clarify v4 User Guide*.

Deploy and Run EDI Integrations

After testing the integrations in your local test Server, you can deploy and configure them for execution in an Clarify Test or Production environment and monitor ongoing execution results, using the following features and tools:

Deployment: Clarify Studio supports the deployment of integrations from one Clarify environment (e.g., Clarify Studio) to another (e.g., Clarify Test / Clarify Production). This deployment is actually a two-step process. First, Clarify bundles all Clarify objects contained in integration and copies them to a server; a final installment process make the integration project available in the different server environment. For more information, see *Designing Integration Solutions/Deploying the Solution*.

Global Variables: *The ERP Connector for JDE* uses a Global Variable JDEEnvironment to bind the SQL Actions used in the included transformation rulesets to your Test or Production database environment. By setting its value at deployment time, you can configure your EDI integrations to use the correct SQL Actions, automatically. For more information, see *Designing Integration Solutions / Sending and Receiving Data / Adapters/Global Variables*.

Cleo Dashboards: The Clarify Dashboard is a web-based application for viewing, analyzing, and reporting on Clarify system status and integration activities. For more information, see the *Cleo Dashboards User Guide* (available within the Dashboard product).

Configure ERP Connector for JDE Integrations with EnterpriseOne

The ERP Connector must be configured to the specific DBMS, version, and location of your JDE EnterpriseOne system. Cleo provides installation and configuration services to ensure compatibility with your JDE environment, so for initial installation, you do not need to master these details.

If you plan to upgrade your JDE E1 system, however, you might need to repeat some configuration steps. To reconfigure ERP Connector, review the following steps.

1. **Download, install, and configure the Java Database Connectivity (JDBC) Driver** for the DBMS used by your JD Edwards system. In order to connect with the DBMS used by your JD Edwards system, the integrations you create using ERP Connector require a JDBC driver. JDBC drivers are supplied by your DBMS vendor, and can differ between releases of the same DBMS. Your database administrator should be able to provide assistance with identification, downloading and installation, of the correct JDBC driver.

Once installed, the JDBC driver must be registered and configured for Cleo Clarify. See *Designing Integration Solutions/Sending and Receiving Data /Adapters/Database Adapters/Data Sources*.

2. **Update Clarify Data Source objects so that they reference the JDBC driver:** Once you have downloaded, installed, and configured the correct JDBC driver for your DBMS, the data source objects included with the ERP Connector for JD Edwards product must be updated so that they point to the JDBC driver.

Clarify uses Data Source objects to implement database adapter connections to your JD Edwards database(s). The ERP Connector for JDE product includes two data sources, one for your production database (identified as *Pd*) and the other for test (identified as *Py*).

To update your Clarify Data Source objects, navigate to the test or production Data Source in Clarify Studio, open the Data Source editor, and follow the instructions documented in the *Designing Integration Solutions/Sending and Receiving Data /Adapters/Database Adapters/Data Sources*.

Once the data source(s) are updated to point to the correct JDBC driver you will not need to update them again unless you upgrade the database used by your JD Edwards system.

3. **Refresh the Clarify Schemas used by ERP Connector for JDE to define JDE table contents.** If you are using a database that is different from the reference version used by ERP Connector for JDE (see Supported product versions and other system requirements), the Clarify database schemas included with the product must be refreshed, using the Clarify Studio's *Database Refresh Wizard*. The wizard updates the Clarify database schemas so that they reflect the database table definitions in your E1 database catalog.

If the E1 version you are using is older than the ERP Connector for JDE reference configuration, the schema refresh process will remove fields in the Clarify schemas that do not exist in your E1 database. If the E1 version you are using is newer than the default configuration, the schema refresh process will add any fields that exist in the E1 database but are not included in the default Clarify database schemas. The refresh also updates field properties that differ between the reference and production database schemas.

Among the properties updated by the schema refresh is the representation of JDE E1 numeric fields in the Clarify schemas. JDE E1 expresses numeric precision for some numeric fields in a separate dictionary, rather than the database catalog. Because the refresh function is driven by metadata in

the catalog, implied decimal point information must be restored in a separate step. Cleo performs both schema refresh and decimal point resetting as part of the installation and configuration service.

Once the Clarify database schemas are refreshed to reflect the database used by your JDE E1 version, you will not need to refresh them again unless you upgrade the JDE database.

Note: You can run the first step in the Database Refresh Wizard to view a list of changes before performing the update step. Due to the impact of schema refresh on JDE numeric field definitions, ***you should not complete the refresh process (update) without first contacting Cleo Support.***

4. **Update Rulesets impacted by schema refresh, if any:** If you are using a version of JDE E1 that is older than the reference version used by ERP Connector for JDE, the Rulesets included with ERP Connector for JDE might refer to some fields that do not exist in your E1 database version. The Clarify Studio's Ruleset editor flags such rules automatically, but they must be removed before using. When needed, Cleo will perform this update as part of the installation and configuration service.
5. **Update Database Adapters:** Clarify uses the Data Bindings section in the Database Adapter to map from the logical view of data represented in Clarify Schemas to the physical Database Schema represented in your database catalog. As part of the configuration process, the Data Binding settings included with ERP Connector for JDE must be updated to reflect the Database Schema name and login credentials needed to access your JDE database.
6. **Update SQL Actions:** ERP Connector for JDE includes a number of SQL Action objects that are invoked by data transformations (maps) to perform lookups and other database actions. These objects reference database schema names that must be updated to reflect your local E1 database.
7. **Create database views for outbound Advance Ship Notices (X12 856):** To support the generation of shipment / order / tare / pack / item levels in Advance Ship Notices, the following SQL views must be created in your production (*Pd*) and test (*Py*) database environments:

```
CREATE VIEW VW_F470371S AS SELECT * FROM PRODDTA.F470371 WHERE SZHL03 = 'S';
CREATE VIEW VW_F470371O AS SELECT * FROM PRODDTA.F470371 WHERE SZHL03 = 'O';
CREATE VIEW VW_F470371T AS SELECT * FROM PRODDTA.F470371 WHERE SZHL03 = 'T';
CREATE VIEW VW_F470371P AS SELECT * FROM PRODDTA.F470371 WHERE SZHL03 = 'P';
CREATE VIEW VW_F470371I AS SELECT * FROM PRODDTA.F470371 WHERE SZHL03 = 'I';
```

For the Test environment, substitute "TESTDTA.F470371" in place of "PRODDTA.F470371".

Inbound Mapping

The sections in this topic identify the fields in the JD Edwards interface tables that are mapped in the Inbound EDI integrations included with the ERP Connector. Each integration typically maps to more than one interface table. The Prefix that appears above each table identifies fields that belong to that table. For example: fields in SYEDOC (Header table) use prefix "SY", and fields in SZEDOC (Detail table) use prefix "SZ".

The JD Edwards interface requires the "Mandatory Fields" listed in each table to be populated. The "Additional Fields" are commonly mapped from the corresponding EDI transactions, but are not required by the JD Edwards interface. The Rulesets included in the ERP Connector product reflect common usage, but may not reflect the exact mapping requirements for your trading partners.

Use the Clarify Studio's Ruleset editor to add or modify mappings based on the EDI specifications provided by your trading partners. For more information, please refer to the *Designing Integration Solutions/Transforming the Data* topics in the *Clarify User Guide*.

Inbound 850 Purchase Order Mapping

Header – F47011

Prefix SY

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	EDI Order Type (DCTO)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Transaction Set Purpose (TPUR)	Bill-to Address Number (AN8)
Ship to Address (SHAN)	
Additional Fields	
Batch Number (EDBT)	Promised Delivery Date (PDDJ)
EDI – Transmission Date (EDDT)	Requested Date (RSDJ)
Trading Partner ID (PNID)	Carrier Code (CARS)
Purchase Order Number (VR01)	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Transaction Date (TRDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)

Detail – F47012

Prefix SZ

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Key Company
KEY: EDI Document Number (EDOC)	KEY: EDI Line Number (EDLN)
EDI Order Type (DCTO)	EDI Transaction Set (EDST)
Send/Receive Flag = R (EDER)	Transaction Set Purpose (TPUR)
Bill-to Address Number (AN8)	Ship to Address (SHAN)
Short Item Number (ITM)	Order/Transaction Quantity (UORG)
Additional Fields	
Batch Number (EDBT)	EDI Transmission Date (EDDT)
Trading Partner ID (PNID)	Item Description (DSC1)
Purchase Order Number (VR01)	Unit of Measure (UOM)
Line Type (LNTY)	Carrier Code (CARS)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	

Notes:

F47012.LNTY is usually 'S' (Stock Item) but it might be 'N' (Non-Stock Item)

If trading partner needs order UOM returned then additional fields might be needed, e.g., F47012.UOM4. This is essential since F47012.UOM on the outbound ASN/Invoice is usually the pricing UOM.

F47012.CITM does NOT get passed to F4211. F4211.CITM does not exist.

Header Notes – F4714

Prefix ZT

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Line Number (EDLN)
KEY: EDI Document Number	KEY: File Name (FILE)
KEY: EDI Document Key Company	KEY: Line Number Work Order
EDI Order Type (DCTO)	
Additional Fields	
Batch Number (EDBT)	Work Station Id (JOBN)
User ID (USER)	Date Updated (UPMJ)
Program Id (PID)	Time of Day (TDAY)
Transaction Originator (TORG)	Order Key Company (KCOO)
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	

Notes:

F4714.TXLN – Field is only 60 characters in length

Detail Notes – F4715

Prefix ZD

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: File Name (FILE)	KEY: Line Number Work Order (LINS)
EDI Order Type (DCTO)	File Name (EDST) – 'F47011'
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	
Additional Fields	
Batch Number (EDBT)	User Id (USER)
Program Id (PID)	Work Station Id (JOBN)
Date Updated (UPMJ)	Time of Day (TDAY)

Notes:

F4715.TXLN – Field is only 60 characters in length

Address – F4706

Prefix ZA

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: File Name (FILE) – 'F47011'	EDI Order Type (DCTO)
KEY: Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	

Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)
Program Id (PID)	Address Line 2 (ADD2)
Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
City (CTY1)	State (ADDS)

Notes:

SQL Access Action: JdeF0005SelectDL01Sy/rt/ky from Table F0005

*SQL statement goes out to the F0005 table using the input parameters listed below. The query returns the DRDL01 value from table F0005 and this output value is mapped to the SYASN field (Price and Adjustment Schedule) in Table F47011.

Parameters:

Description (DRDL01)–Output (SYASN)

Product Code (DRSY)-Input (For example, hard code “5547”) User

Defined Code (DRRT)-Input (For example, hard code “FT”)

User Defined Code (DRKY)-Input (For example the value from element FOB01: Shipment Method of Payment)

SQL Access Action: JdeF0005SelectKyfrom Table F0005

* SQL statement goes out to the F0005 table using the input parameters listed below. The query returns the DRDL01 value from table F0005 and this output value is mapped to the SYFRTH field (Freight Handling Code) in Table F47011.

Parameters:

User Defined Code (DRKY)–Output (SYFRTH)

Product Code (DRSY)-Input (For example, hard code “42”)

Description (DRDL02)-Input (For example the value from element FOB01: Shipment Method of Payment)

User Defined Code (DRRT)-Input (For example, hard code “FR”)

SQL Access Action: JdeF0005SelectKy2 from Table F0005

*SQL statement goes out to the F0005 table using the input parameters listed below. The query returns the DRKY value from table F0005 and this output value is mapped to the SYMOT field (Mode of Transport) in Table F47011.

Parameters:

User Defined Code (DRKY)–Output (SYMOT)

Product Code (DRSY)-Input (For example, hard code “00”)

User Defined Code (DRRT)-Input (For example, hard code “TM”)

Special Handling Code (DRSPHD)-Input (For example the value from element TD504: Transportation Method/Type Code)

SQL Access Action: JdeF0005SelectKyUsingSy/rt from Table F0005

*SQL statement goes out to the F0005 table using the input parameters listed below. The query returns the DRKY value from table F0005 and this output value is mapped to the SYMCU field (Business Unit) in Table F47011.

Parameters:

User Defined Code (DRKY)-Output (SYMCU)

Product Code (DRSY)-Input (For example, hard code "5547")

User Defined Code (DRRT)-Input (For example, hard code "BU")

SQL Access Action: JdeF0111SelectWwan8 from Table F0111

*SQL statement goes out to the F0111 table using the input parameters listed below. The query returns the WWAN8 value from table F0111 and this output value is mapped to the SYSHAN field (Ship To Address Number) in Table F47011.

Parameters:

Address Number (WWAN8)-Output (SYSHAN)

Name (WWMLNM)-Input (For example, value from element N104: Identification Code)

Type Code (WWTYC)-Input (For example, hard code "Z")

WWREM1 (Remark 1)-Input (For example, (PNID) Trading Partner ID)

SQL Access Action: JdeF0101SelectAban81 from Table F0101

*SQL statement goes out to the F0101 table using the input parameters listed below. The query returns the ABAN81 value from table F0101 and this output value is mapped to the SYAN8 field (Bill To Address Number) in Table F47011.

Parameters:

Address Number (ABAN81)-Output (SYAN8)

Name (ABAN8)-Input (For example, value WWAN8 which was returned in previous SQL query)

SQL Access Action: JdeF4104SelectIvitm from Table F4104

*SQL statement goes out to the F4104 table using the input parameters listed below. The query returns the IVITM value from table F4104 and this output value is mapped to the SZITM field (Second Item Number) in Table F47012.

Parameters:

Short Item Number (IVITM)-Output (SZITM)

Address Number (IVAN8)-Input (For example, value ABAN81 from Table F0101 which was returned in previous SQL query)

Item Number (IVCITM)-Input (For example, value from element PO107 Product/Service ID)

SQL Access Action: JdeF4104SelectIvlitm from Table F4104

*SQL statement goes out to the F4104 table using the input parameters listed below. The query returns the IVLITM value from table F4104 and this output value is mapped to the SZLITM field (Second Item Number) in Table F47012.

Parameters:

Second Item Number (IVLITM)-Output (SZLITM)

Address Number (IVAN8)-Input (For example, value ABAN81 from Table F0101 which was returned in previous SQL query)

Item Number (IVCITM)-Input (For example, value from element PO107 Product/Service ID)

Inbound 810 Invoice Mapping

Header – F47041

Prefix SY

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	EDI Order Type (DCTO)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Order Number (DOCO)	Bill-to Address Number (AN8)
GL Date (DGJ)	Order Key Company (KCOO)
Invoice Number (VINV)	
Additional Fields	
Batch Number (EDBT)	Document Type – Voucher Only
EDI-Transmission Date (EDDT)	Invoice Date (DIVJ)
Trading Partner ID (PNID)	Company (CO)
Purchase Order Number (VR01)-	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Transaction Originator (TORG)	Work Station Id (JOBN)
Supplier Remark (RMK)	Date Updated (UPMJ)
Processed (Y/N) (EDSP)	Reference 2 (VR02)
Order Suffix (SFXO)	

Detail – F47042

Prefix SZ

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY:EDI Document Key Company
KEY: EDI Document Number (EDOC)	KEY:EDI Line Number (EDLN)
EDI Order Type (DCTO)	EDI Transaction Set (EDST)
Send/Receive Flag = R (EDER)	Order Number (DOCO)
Address Number (AN8)	Order Key Company (KCOO)

Open Amount (AAP)	Line Type (LNID)
Business Unit (MCU)	Customer Item Number (CITM)
Second Item Number (LITM)	Pay Status (PST)
Quantity Open (UOPN)	
Additional Fields	
Batch Number (EDBT)	Edi – Transmission Date (EDDT)
Trading Partner ID (PNID)	Item Description (DSC1)
Purchase Order Number (VR01)-	Unit of Measure (UOM)
Line Type (LNTY)	Carrier Code (CARS)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Document Type – Voucher Only
Record Sequence (EDSQ)	Record Type (EDTY)
Invoice Number (VINV)	Company (CO)
Transaction Originator (TORG)	Processed (Y/N) (EDSP)
Unit Price (UPRC)	Allowance or Charge Indicator (ACIN)
Order Suffix (SFXO)	Currency Code (CRCD)

Inbound 810 Invoice – Summary – F47044

Prefix SW

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	EDI Order Type (DCTO)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Order Number (DOCO)	Bill-to Address Number (AN8)
EDI Line Number (EDLN)	Order Key Company (KCOO)
Quantity Open (UOPN)	Open Amount (AAP)
Additional Fields	
Address Number (AN8)	Document Company (KCO)
Transaction Originator (TORG)	Time of Day (TDAY)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Currency Mode-Foreign or Domestic	Currency Code–From (CRCD)
Amount Open (AAP)	EDI-Translation Format (EDFT)
Order Suffix (SFXO)	

Order Address Information – F4706

Prefix ZA

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
KEY:File Name (FILE) – ‘F47011’	EDI Order Type (DCTO)

Order Number (DOCO)	Order Key Company (KCOO)
KEY: Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	
Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)
Program Id (PID)	Address Line 2 (ADD2)
Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
Record Sequence (EDSQ)	City (CTY1)
Record Type (EDTY)	State (ADD5)
Transaction Originator (TORG)	

Notes:

SQL Access Action: JdeF0111SelectWwan8 from Table F0111

*SQL statement goes out to the F0111 table using the input parameters listed below. The query returns the WWAN8 value from table F0111 and this output value is mapped to the SYSHAN field (Ship To Address Number) in Table F47046.

Parameters:

Address Number (WWAN8)–Output (SYSHAN)

Name (WWMLNM)-Input (For example, value from element N104: Identification Code) Type

Code (WWTYC)-Input (For example, hard code “Z”)

WWREM1 (Remark 1)-Input (For example, (PNID) Trading Partner ID)

SQL Access Action: JdeF0101SelectAban81 from Table F0101

*SQL statement goes out to the F0101 table using the input parameters listed below. The query returns the ABAN81 value from table F0101 and this output value is mapped to the SYAN8 field (Bill To Address Number) in Table F47046.

Parameters:

Address Number (ABAN81)–Output (SYAN8)

Name (ABAN8)-Input (For example, value WWAN8 which was returned in previous SQL query)

SQL Access Action: JdeF4104SelectIvitm from Table F4104

*SQL statement goes out to the F4104 table using the input parameters listed below. The query returns the IVITM value from table F4104 and this output value is mapped to the SZITM field (Second Item Number) in Table F47042.

Parameters:

Short Item Number (IVITM)–Output (SZITM)

Address Number (IVAN8)-Input (For example, value ABAN81 from Table F0101 which was returned in previous SQL query)

Item Number (IVCITM)-Input (For example, value from element IT107 Product/Service ID)

SQL Access Action: JdeF4104SelectIvlitm from Table F4104

*SQL statement goes out to the F4104 table using the input parameters listed below. The query returns the IVLITM value from table F4104 and this output value is mapped to the SZLITM field (Second Item Number) in Table F47042.

Parameters:

Second Item Number (IVLITM)-Output (SZLITM)

Address Number (IVAN8)-Input (For example, value ABAN81 from Table F0101 which was returned in previous SQL query)

Item Number (IVCITM)-Input (For example, value from element IT107 Product/Service ID)

Inbound 820 Payment Order Mapping

EDI Lockbox/Payment Header – F47111

Prefix Z1

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Alpha Name (ALPH)	
Additional Fields	
EDI Transmission Date (EDDT)	Processed – Y/N (EDSP)
Trading Partner ID (PNID)	Time of Day (TDAY)
Date Updated (UPMJ)	

EDI Lockbox Header – F47112

Prefix Z2

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Deposit Number (DEPN)	Total Transaction Amount (TTOT)
Additional Fields	
EDI Transmission Date (EDDT)	Processed – Y/N (EDSP)
Trading Partner ID (PNID)	Time of Day (TDAY)
Batch Number (EDBT)	Record Sequence (EDSQ)
Date Updated (UPMJ)	

EDI Lockbox/Payment Detail – F47113

Prefix Z3

Mandatory Fields

KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: Deposit Number (DEPN)	KEY: Line Number (LNID)
Check Number (DOCM)	Bank Transit Number (TNST)
Customer Bank Account Number	Currency Code (CRCD)
Total Transaction Amount (TTOT)	Address Number (AN8)
Additional Fields	
Check/Item Date (DMTJ)	Processed – Y/N (EDSP)
Batch Number (EDBT)	Time of Day (TDAY)
Record Sequence (EDSQ)	Alpha Name (ALPH)
EDI Transaction Set Number (EDST)	EDI Send/Receive Indicator (EDER)
Date Updated (UPMJ)	

EDI Lockbox/Payment Detail – F47114

Prefix Z4

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: Deposit Number (DEPN)	KEY: Line Number (LNID)
KEY: Sequence Number (SEQN)	Check Number (DOCM)
Document Number (DOC)	Document Type (DCT)
Discount Taken (ADSA)	Gross Amount (AG)
Additional Fields	
Document Company (KCO)	Processed – Y/N (EDSP)
Line Number (LNID)	Time of Day (TDAY)
Batch Number (EDBT)	Record Sequence (EDSQ)
Purchase Number (PO)	Remark (RMK)
Date Updated (UPMJ)	Currency Amount (ACR)
EDI Transaction Set Number (EDST)	EDI Send/Receive Indicator (EDER)

Inbound 860 Purchase Order Change Mapping

Header-F47131

Prefix SY

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	EDI Order Type (DCTO)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Transaction Set Purpose (TPUR)	Bill-to Address Number (AN8)
Ship to Address (SHAN)	Order Key Company (KCOO)
Order Number (DOCO)	
Additional Fields	

Batch Number (EDBT)	Order Suffix (SFXO)
Company (CO)	Transaction Originator (TORG)
Trading Partner ID (PNID)	Backorders Allowed (BACK) –Y/N
Purchase Order Number (VR01)	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Transaction Date (TRDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)

Detail – F47132*Prefix SZ*

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Key Company
KEY:EDI Document Number (EDOC)	KEY:EDI Line Number (EDLN)
EDI Order Type (DCTO)	Order Company (Order Number)
Send/Receive Flag = R (EDER)	Document (DOCO)
Bill-to Address Number (AN8)	
Additional Fields	
EDI Transaction Set (EDST)	Order Suffix (SFXO)
Trading Partner ID (PNID)	Change Code (CHGC)
Business Unit (MCU)	Unit of Measure (UOM)
Line Type (LNTY)	Unit of Measure – Primary (UOM1)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Transaction Originator (TORG)
Status Code – Next (NXTR)	Committed (H/S) (COMM)
Company (CO)	Backorders Allowed (BACK) – Y/N
Date – Actual Ship Date (ADDJ)	Line Number (LNID)
Short Item Number (ITM)	2 nd Item Number (LITM)
Units-Order/Transaction Quantity	Quantity Shipped (SOQS)
Transaction Set Purpose (TPUR)	

Address – F4706*Prefix ZA*

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
KEY:File Name (FILE) – 'F47011'	EDI Order Type (DCTO)
KEY:Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	
Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)
Program Id (PID)	Address Line 2 (ADD2)

Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
City (CTY1)	State (ADDS)

Notes:

SQL Access Action: JdeF0111SelectWwan8 from Table F0111

*SQL statement goes out to the F0111 table using the input parameters listed below. The query returns the WWAN8 value from table F0111 and this output value is mapped to the SYSHAN field (Ship To Address Number) in Table F47046.

Parameters:

Address Number (WWAN8)–Output (SYSHAN)

Name (WWMLNM)-Input (For example, value from element N104: Identification Code) Type

Code (WWTYC)-Input (For example, hard code “Z”)

WWREM1 (Remark 1)-Input (For example, (PNID) Trading Partner ID)

SQL Access Action: JdeF0101SelectAban81 from Table F0101

*SQL statement goes out to the F0101 table using the input parameters listed below. The query returns the ABAN81 value from table F0101 and this output value is mapped to the SYAN8 field (Bill To Address Number) in Table F47046.

Parameters:

Address Number (ABAN81)–Output (SYAN8)

Name (ABAN8)-Input (For example, value WWAN8 which was returned in previous SQL query)

SQL Access Action: JdeF4211SelectSdshanSdan8 from Table F4211

*SQL statement goes out to the F4211 table using the input parameters listed below. The query returns the SDAN8 and SDSHAN, values from table F4211 and the output values are mapped to the SYAN8 (Bill To Address Number) and SYSHAN (Ship To Address Number) fields in Table F47131.

Parameters:

Bill To Address Number (SDAN8)–Output (SYAN8) Ship

To Address Number (SDSHAN)–Output (SYSHAN)

Purchase Order Number (SDVR01)-Input (For example, value in element BCH03: Purchase Order Number)

Inbound 945 Warehouse Shipping Advice Mapping

Header – F47131

Prefix SY

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	EDI Order Type (DCTO)

EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Transaction Set Purpose (TPUR)	Bill-to Address Number (AN8)
Ship to Address (SHAN)	Order Key Company (KCOO)
Order Number (DOCO)	
Additional Fields	
Batch Number (EDBT)	Order Suffix (SFXO)
Company (CO)	Transaction Originator (TORG)
Trading Partner ID (PNID)	Backorders Allowed (BACK) –Y/N
Purchase Order Number (VR01)	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Transaction Date (TRDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)

Detail – F47132

Prefix SZ

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Key Company
KEY:EDI Document Number (EDOC)	KEY:EDI Line Number (EDLN)
EDI Order Type (DCTO)	Order Company (Order Number)
Send/Receive Flag = R (EDER)	Transaction Set Purpose (TPUR)
Bill-to Address Number (AN8)	Document (Order No, Invoices, etc.)
Additional Fields	
EDI Transaction Set (EDST)	Order Suffix (SFXO)
Trading Partner ID (PNID)	Change Code (CHGC)
Business Unit (MCU)	Unit of Measure (UOM)
Line Type (LNTY)	Unit of Measure – Primary (UOM1)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Transaction Originator (TORG)
Status Code – Next (NXTR)	Committed (H/S) (COMM)
Company (CO)	Backorders Allowed (BACK) – Y/N
Date – Actual Ship Date (ADDJ)	Line Number (LNID)
Short Item Number (ITM)	2 nd Item Number (LITM)
Units-Order/Transaction Quantity	Quantity Shipped (SOQS)

Notes:

SQL Access Action: JdeF4201SelectAn8KcooDctoMcuCo from Table F4201

*SQL statement goes out to the F4201 table using the input parameters listed below. The query returns the SHAN8 , SHKCOO, SHDCTO, SHMCU, and SHCO, values from table F4201 and the output values are mapped to the SYAN8 (Address Number), SYKCOO (Order Number), SYDCTO (Order Type), SYMCU (Business Unit) and SYCO (Company) fields in Table F47131.

Parameters:

Address Number (SHAN8)–Output (SYAN8)

Order Number (SHKCOO)–Output (SYKCOO)

Order Type (SHDCTO)–Output (SYDCTO)

Business Unit (SHMCU)–Output (SYM CU)

Company (SHCO)–Output (SYCO)

Document (SHDOCO)-Input (For example, value in WO602: Depositor Order Number)

Inbound 867 Product Transfer & Resale Report Mapping

Header – F47011

Prefix SY

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	Business Unit (MCU)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Transaction Set Purpose (TPUR)	Bill to Address Number (AN8)
Ship to Address (SHAN)	Purchase Order Number (VR01)
Vendor Number (VR02)	
Additional Fields	
Batch Number (EDBT)	Date Requested (DRQJ)
EDI-Transmission Date (EDDT)	Price and Adjustment Schedule (ASN)
Trading Partner ID (PNID)	User Id (USER)
Date Updated (UPMJ)	Program Id (PID)
Transaction Date (TRDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	

Detail – F47012

Prefix SZ

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Key Company
KEY:EDI Document Number (EDOC)	KEY:EDI Line Number (EDLN)
EDI Order Type (DCTO)	EDI Transaction Set (EDST)
Send/Receive Flag = R (EDER)	Transaction Set Purpose (TPUR)
Bill-to Address Number (AN8)	Ship to Address (SHAN)
Short Item Number (ITM)	Order/Transaction Quantity (UORG)
Purchase Order Number (VR01)	Vendor Number (VR02)
Additional Fields	
Batch Number (EDBT)	EDI-Transmission Date (EDDT)
Trading Partner ID (PNID)	Date – Order/Transaction (TRDJ)
Business Unit – Header (MCU)	Transaction Originator (TORG)
Amount – Price per Unit (UPRC)	Date – Invoice (IVD)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Pull Signal (PSIG)
Item Number (AITM)	

Header Notes – F4714

Prefix ZT

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Line Number (EDLN)
KEY:EDI Document Number	KEY:File Name (FILE)
KEY:EDI Document Key Company	KEY:Line Number Work Order
EDI Order Type (DCTO)	
Additional Fields	
Batch Number (EDBT)	Work Station Id (JOBN)
User ID (USER)	Date Updated (UPMJ)
Program Id (PID)	Time of Day (TDAY)
Transaction Originator (TORG)	Order Key Company (KCOO)
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	

Notes:

F4714.TXLN – Field is only 60 characters in length

Detail Notes – F4715

Prefix ZD

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
KEY:File Name (FILE)	KEY: Line Number Work Order (LINS)
EDI Order Type (DCTO)	File Name (EDST) – ‘F47011’
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	
Additional Fields	
Batch Number (EDBT)	User Id (USER)
Program Id (PID)	Work Station Id (JOBN)
Date Updated (UPMJ)	Time of Day (TDAY)
Line Number (LNID)	Transaction Originator (TORG)

Notes:

F4715.TXLN – Field is only 60 characters in length

Address – F4706

Prefix ZA

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
KEY:File Name (FILE) – ‘F47011’	EDI Order Type (DCTO)
KEY:Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	
Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)

Program Id (PID)	Address Line 2 (ADD2)
Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
City (CTY1)	State (ADDS)
Transaction Originator (TORG)	

Notes:

SQL Access Action: JdeF0111SelectWwan8 from Table F0111

*SQL statement goes out to the F0111 table using the input parameters listed below. The query returns the WWAN8 value from table F0111 and this output value is mapped to the SYSHAN field (Ship To Address Number) in Table F47011.

Parameters:

Address Number (WWAN8)–Output (SYSHAN)

Name (WWMLNM)-Input (For example, value from element N104: Identification Code) Type

Code (WWTYC)-Input (For example, hard code “Z”)

WWREM1 (Remark 1)-Input (For example, (PNID) Trading Partner ID)

SQL Access Action: JdeF0005SelectKyUsingSy/rt from Table F0005

*SQL statement goes out to the F0005 table using the input parameters listed below. The query returns the DRKY value from table F0005 and this output value is mapped to the SYMCU field (Business Unit) in Table F47011.

Parameters:

User Defined Code (DRKY)–Output (SYMCU)

Product Code (DRSY)-Input (For example, hard code “5547”) User

Defined Code (DRRT)-Input (For example, hard code “BU”)

Inbound 823 Lockbox Mapping

Payment Header – F47111

Prefix Z1

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Alpha Name (ALPH)	
Additional Fields	
EDI Transmission Date (EDDT)	Processed – Y/N (EDSP)
Trading Partner ID (PNID)	Time of Day (TDAY)
Date Updated (UPMJ)	

EDI Lockbox Header – F47112

Prefix Z2

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
EDI Transaction Set (EDST)	Send/Receive Flag = R (EDER)
Deposit Number (DEPN)	Total Transaction Amount (TTOT)
Additional Fields	
EDI Transmission Date (EDDT)	Processed – Y/N (EDSP)
Trading Partner ID (PNID)	Time of Day (TDAY)
Batch Number (EDBT)	Record Sequence (EDSQ)
Date Updated (UPMJ)	

Payment Detail – 47113

Prefix Z3

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
KEY:Deposit Number (DEPN)	KEY:Line Number (LNID)
Check Number (DOCM)	Bank Transit Number (TNST)
Customer Bank Account Number	Currency Code (CRCD)
Total Transaction Amount (TTOT)	Address Number (AN8)
Additional Fields	
Check/Item Date (DMTJ)	Processed – Y/N (EDSP)
Batch Number (EDBT)	Time of Day (TDAY)
Record Sequence (EDSQ)	Alpha Name (ALPH)
EDI Transaction Set Number (EDST)	EDI Send/Receive Indicator (EDER)
Date Updated (UPMJ)	

Payment Detail – F47114

Prefix Z4

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
KEY:Deposit Number (DEPN)	KEY:Line Number (LNID)
KEY:Sequence Number (SEQN)	Check Number (DOCM)
Document Number (DOC)	Document Type (DCT)
Document Pay Item (SFX)	Gross Amount (AG)
Currency Amount field (ACR)	Receipt Matching Reference (RMR1)
Foreign Discount Taken (CDSA)	Control/Statement Field (CTL)
Discount Taken (ADSA)	
Additional Fields	
Document Company (KCO)	Processed – Y/N (EDSP)
Line Number (LNID)	Time of Day (TDAY)
Batch Number (EDBT)	Record Sequence (EDSQ)
Purchase Number (PO)	Remark (RMK)
Date Updated (UPMJ)	EDI Send/Receive Indicator (EDER)
EDI Transaction Set Number (EDST)	

Inbound 856 Advance Ship Notice Mapping

Header – F47031

Prefix SY

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	EDI Order Type (DCTO)
EDI Transaction Set = 856(EDST)	Send/Receive Flag = R (EDER)
Transaction Set Purpose (TPUR)	Bill-to Address Number (AN8)
Ship to Address (SHAN)	Order Key Company (KCOO)
Order Number (DOCO)	
Additional Fields	
Batch Number (EDBT)	Order Suffix (SFXO)
Shipment ID (SPID)	Transaction Originator (TORG)
Trading Partner ID (PNID)	Remark (RMK)
Purchase Order Number (VR01)	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Actual Ship Date (ADDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Order Date (TRDJ)	Ship Date (PPDJ)

Detail – F47032

Prefix SZ

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Key Company
KEY:EDI Document Number (EDOC)	KEY:EDI Line Number (EDLN)
EDI Order Type (DCTO)	Order Company (Order Number)
Send/Receive Flag = R (EDER)	Transaction Set Purpose (TPUR)
Bill-to Address Number (AN8)	Document (DOCO)
Address Number – Ship To (SHAN)	Hierarchical Level (HLVL)
Hierarchical Level Code (HL03)	
Additional Fields	
EDI Transaction Set (EDST)	Order Suffix (SFXO)
Trading Partner ID (PNID)	Type Record (EDTY)
Business Unit (MCU)	Unit of Measure (UOM)
Ship Date (PPDJ)	Shipment ID (SPID)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Transaction Originator (TORG)
Extended Price (AEXP)	Carrier Number (ANCR)
Item Number – Customer/Supplier	EDI Transmission Date (EDDT)
Date – Actual Ship Date (ADDJ)	Line Number (LNID)
Short Item Number (ITM)	2 nd Item Number (LITM)
Units-Order/Transaction Quantity	Gross Weight (GRWT)
Gross Weight Unit of Measure (GWUM)	Quantity Received (UREC)
Line Type (LNTY)	

Address – F4706

Prefix ZA

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: File Name (FILE) – ‘F47011’	EDI Order Type (DCTO)
KEY: Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	
Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)
Program Id (PID)	Address Line 2 (ADD2)
Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
City (CTY1)	State (ADDS)
Transaction Originator (TORG)	

Notes – F4714

Prefix ZT

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Line Number (EDLN)
KEY: EDI Document Number	KEY: File Name (FILE)
KEY: EDI Document Key Company	KEY: Line Number Work Order
EDI Order Type (DCTO)	
Additional Fields	
Batch Number (EDBT)	Work Station Id (JOBN)
User ID (USER)	Date Updated (UPMJ)
Program Id (PID)	Time of Day (TDAY)
Transaction Originator (TORG)	Order Key Company (KCOO)
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	

Notes:

F4714.TXLN – Field is only 60 characters in length

SQL Access Action: JdeF0111SelectWwan8 from Table F0111

*SQL statement goes out to the F0111 table using the input parameters listed below. The query returns the WWAN8 value from table F0111 and this output value is mapped to the SYSHAN field (Ship To Address Number) in Table F47046.

Parameters:

Address Number (WWAN8)–Output (SYSHAN)
Name (WWMLNM)-Input (For example, value from element N104: Identification Code) Type
Code (WWTYC)-Input (For example, hard code “Z”)
WWREM1 (Remark 1)-Input (For example, (PNID) Trading Partner ID)

SQL Access Action: JdePdF4104SelectIvitm from Table F4104

*SQL statement goes out to the F4104 table using the input parameters listed below. The query returns the IVITM value from table F4104 and this output value is mapped to the SZITM field (Second Item Number) in Table F47042.

Parameters:

Short Item Number (IVITM)–Output (SZITM)
Address Number (IVAN8)-Input (For example, value ABAN81 from Table F0101 which was returned in previous SQL query)
Item Number (IVCITM)-Input (For example, value from element IT107 Product/Service ID)

SQL Access Action: JdePdF4104SelectIvlitm from Table F4104

*SQL statement goes out to the F4104 table using the input parameters listed below. The query returns the IVLITM value from table F4104 and this output value is mapped to the SZLITM field (Second Item Number) in Table F47042.

Parameters:

Second Item Number (IVLITM)–Output (SZLITM)

Address Number (IVAN8)-Input (For example, value ABAN81 from Table F0101 which was returned in previous SQL query)

Item Number (IVCITM)-Input (For example, value from element IT107 Product/Service ID)

Outbound Mapping

This topic identifies the fields in the JD Edwards interface tables that are mapped to the outbound EDI integrations included with the ERP Connector. Each integration typically maps more than one interface table to create each transaction. The Prefix that appears above each table identifies fields that belong to that table. For example: fields in SYEDOC (Header table) use prefix “SY”, and fields in SZEDOC (Detail table) use prefix “SZ”.

The "Mandatory Fields" listed in each table are the fields that must contain data for the outbound transaction to be valid. The “Additional Fields” are commonly mapped from the JD Edwards interface, but are not required in all cases. The Rulesets included in the ERP Connector product reflect common usage, but may not reflect the exact mapping requirements for your trading partners. Use the Clarify Studio’s Ruleset editor to add or modify mappings based on the EDI specifications provided by your trading partners.

Outbound 810 Invoice Mapping

Header – F47046

Prefix SY

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	Bill to Address Number (AN8)
Date – Order/Transaction (TRDJ)	Purchase Order Number (VR01)
Ship to Address (SHAN)	
Additional Fields	
Batch Number (EDBT)	Date Requested (DROJ)
EDI-Transmission Date (EDDT)	Price and Adjustment Schedule (ASN)
Trading Partner ID (PNID)	User Id (USER)
Date Updated (UPMJ)	Program Id (PID)
Date – Actual Ship Date (ADDJ)	Work Station Id (JOBN)
Order Company (Order Number)	Document (Order No Invoice etc.)
Transaction Type Code (DCTO)	Method of Payment (FRTH)
Time of Day (TDAY)	

Notes:

Database Adapter (Data Binding): Name:

JDE.810.F47046/F47047

Version: PRODDTA (Production) or TESTDTA (Test)

Selection Criteria:

(Prod) SELECT * FROM PRODDTA.F47046 WHERE SYEDSP = “ “;

(Test) SELECT * FROM TESTDTA.F47046 WHERE SYEDSP = “ “;

Invoice Detail – F47047

Prefix SZ

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
Date – Order/Transaction (TRDJ)	Purchase Order Number (VR01)
Ship to Address (SHAN)	Bill to Address Number (AN8)
Line Number (LNID)	
Additional Fields	
Batch Number (EDBT)	Date Requested (DROJ)
EDI-Transmission Date (EDDT)	Price and Adjustment Schedule (ASN)
Trading Partner ID (PNID)	User Id (USER)
Date Updated (UPMJ)	Program Id (PID)
Date – Actual Ship Date (ADDJ)	Work Station Id (JOBN)
Order Company (Order Number)	Document (Order No Invoice etc.)
Transaction Type Code (DCTO)	Method of Payment (FRTH)
Quantity Shipped (SOQS)	Unit of Measure (UOM)
Amount – Price per Unit (UPRC)	Item Number (LITM)
Customer/Supplier – Item Number	Description (DSC1)
Item Number – Short (ITM)	3 rd Item (AITM)
Line Type (LNTY)	Amount – Extended Price (AEXP)
End Use (EUSE)	Carrier Number (CARS)
Time of Day (TDAY)	

Notes:

SQL Access Action: JdeF47047SelectSoqsSum from table F47047 (Version 1)

*SQL statement selects SOQS by using the input parameters in the fields below. The output value is used to populate the element IT102

Parameters:

- EDI- Document Number (EDOC)-Input (SYEDOC from table F47046) Item Number- Short (ITM)-Input (SZITM from table F47047)
- EDI- Document Type (EDCT)-Input (SYEDCT from table F47046)
- Company – Key (EKCO)-Input (SYEKCO from table F47046)
- Quantity Shipped (SOQS)-Output (IT102)

Additional Header – F470461

Prefix S3

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	Bill to Address Number (AN8)
Document – Voucher, Invoice, etc.	
Additional Fields	
Batch Number (EDBT)	Date – Due (DDU)
Discount Days (DCD)	Date – Net Due (DDJ)
Net Days to Pay (NDTP)	User Id (USER)
Date Updated (UPMJ)	Program Id (PID)
Description – Payment Terms (PTD)	Work Station Id (JOBN)
Amount – Gross (AG)	Document (Order No Invoice etc.)
Transaction Type Code (DCTO)	Time of Day (TDAY)

Order Address Information – F4706

Prefix ZA

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: File Name (FILE) – ‘F47011’	EDI Order Type (DCTO)
KEY: Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	
Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)
Program Id (PID)	Address Line 2 (ADD2)
Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
City (CTY1)	State (ADDS)
Address Number (AN8)	Country (CTR)

Outbound 810 Invoice – Header Notes – F4714

Prefix ZT

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Line Number (EDLN)
KEY: EDI Document Number	KEY: File Name (FILE)
KEY: EDI Document Key Company	KEY: Line Number Work Order
EDI Order Type (DCTO)	
Additional Fields	
Batch Number (EDBT)	Work Station Id (JOBN)

User ID (USER)	Date Updated (UPMJ)
Program Id (PID)	Time of Day (TDAY)
Transaction Originator (TORG)	Order Key Company (KCOO)
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	

Notes:

SQL Access Action: JdeF4074SelectAprp1Uprc from Table F4074 (Prefix AL) – (Version 1)

*SQL statement selects APRP1 and UPRC by using the input parameters in the fields below. The output values are used to populate the elements SAC02, SAC05, SAC08

Parameters:

KEY: Document (Order No Invoice etc.) (DOCO)–Input (SYDOCO value from Table F47046)

KEY: Line Number (LNID)–Input (SZLNID value from Table F47047)

KEY: Order Company (Order Number) (KCOO)–Input (SYKCOO value from Table F47046)

KEY: Sequence Number (OSEQ)–Input (Hard Coded, example “10”)

Commodity Class (APRP1)–Output (This value is used to find service, promotion, allowance, or charge description in the F005 table which then populates SAC02)

Amount – Price per Unit (UPRC)-Output (This value is used to get the SAC05 and SAC08 values)

*This query may be used several times, changing the ALOSEQ value. The ALOSEQ value is an arbitrary number that can be increased or decreased. In this example the ALOSEQ value links to the service, promotion, allowance, or charge description in the F0005 table.

SQL Access Action: JdeF0005SelectDI01/dI02Sy/rt/ky from Table F0005 (Version1)

*SQL statement (DL01 lookup) selects the Description from User Defined Codes Table by using the input parameters defined in the fields below. The output value is used to populate element SAC02.

Parameters:

Description (DL01)–Output (SAC02)

KEY: Product Code (SY)–Input (example: Hard Coded “45”)

KEY: User Defined Codes (RT)-Input (example: Hard Coded “P1”)

KEY: User Defined Codes (KY)-Input (ALAPRP1 value from Table F4074)

SQL Access Action: JdeF0111SelectWwrem1WwnickWwmlnmWwalph from Table F0111 (Prefix WW)- Versions 1 and 2

*SQL statement selects the Customer Mailing Name or DUNS number from the F0111 (Who’s Who) Table by using the input parameters defined in the fields below. The output value is used to populate element N104.

Parameters:

Mailing Name (MLNM)–Output (N104)

KEY: Address Number (AN8)-Input (SYSHAN value from Table F47046 for Ship To or SYAN8 value from Table F47046 for Bill To)

Remark 1 (REM1)-Input (Hard Coded "Z" in our example)

SQL Access Action: JdeF0101SelectAban81Abalph from Table F0101 (Version 1)

*SQL statement selects the Ship To Name from the Address Book Master Table by using the input parameters defined in the fields below. The output value is used to populate element N102.

Parameters:

Name – Alpha (ALPH)-Output (N102)

KEY: Address Number (SHAN)-Input (This number is the SHAN value from the Header table)

SQL Access Action: JdeF0101SelectAban81Abalph from Table F0101 (Version 2)

*SQL statement selects the Carrier Information from the Address Book Master Table by using the input parameters defined in the fields below. The output value is used to populate element CAD05.

Parameters:

Name – Alpha (ALPH)-Output (CAD05)

KEY: Address Number (CARS)-Input (This number is the CARS value from the Detail table)

SQL Access Action: JdeF0101SelectAban81Abalph from Table F0101 (Version 1)

*SQL statement selects the Bill To Name from the Address Book Master Table by using the input parameters defined in the fields below. The output value is used to populate element N102.

Parameters:

Name – Alpha (ALPH)-Output (N102)

KEY: Address Number (AN8)-Input (This number is the AN8 value from the Header table)

SQL Access Action: JdeF4906SelectCmscac from Table F4906 –Version 2

*SQL statement selects the SCAC Code (Carrier Number) from Table F4906 by using the input parameters defined in the fields below. The output value is used to populate element CAD04.

Parameters:

Standard Carrier Alpha Code (CMSCAC)-Output (CAD04)

Carrier Number (CMCARS)-Input (SZCARS value from Table F47047)

Outbound 856 Advance Ship Notice Mapping

Header – F47036

Prefix SY

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	Transaction Set Purpose Code (TPUR)
Trading Partner ID (PNID)	Send/Receive Flag = R (EDER)
Additional Fields	
Batch Number (EDBT)	Time of Day (TDAY)
Transaction Originator (TORG)	User Id (USER)
Program Id (PID)	Work Station Id (JOBN)

Detail – F470371

Prefix SZ

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Key Company
KEY:EDI Document Number (EDOC)	KEY:EDI Line Number (EDLN)
KEY:Type Code (STPC)	KEY:Work Order–Line Number (LINS)
EDI Order Type (DCTO)	Order Company (Order Number)
Pick Slip Number (PSN)	Send/Receive Flag = R (EDER)
Bill-to Address Number (AN8)	Document (DOCO)
Hierarchical Level Code (HL03)	Additional Container I.D. (CNID)
Additional Fields	
EDI Transaction Set = 856 (EDST)	Order Suffix (SFXO)
Trading Partner ID (PNID)	Date Requested (DRQJ)
Business Unit (MCU)	Unit of Measure (UOM)
Line Type (LNTY)	Unit of Measure – Primary (UOM1)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Transaction Originator (TORG)
Company (CO)	Purchase Order Number (VR01)
Date – Actual Ship Date (ADDJ)	Line Number (LNID)
Short Item Number (ITM)	2 nd Item Number (LITM)
Units-Order/Transaction Quantity	Quantity Shipped (SOQS)

Notes:

Custom Detail levels can be created for different ASN hierarchical levels (HLs), using F470371S (shipment), F470371O (order), F470371I (item), F470371P (pack / carton), F470371T (tare / pallet)

Data Binding:

Name: JDE.856.F470376/F470371
Version: PRODDTA (Production) or TESTDTA (Test)

March 2015

38

EXTOL International, Inc.

This confidential information may not be provided to 3rd parties.

Address – F4706

Prefix ZA

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: File Name (FILE) – ‘F47011’	EDI Order Type (DCTO)
KEY: Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	
Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)
Program Id (PID)	Address Line 2 (ADD2)
Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
City (CTY1)	State (ADDS)

Notes:

SQL Access Action: JdeF4706SelectAllFields from Table F4706)-Versions SOI and SOTPI

*SQL statement selects ADD1, ADD2, CTY1, ADDS, ADDZ and CTR by using the input parameters in the field below. The output values are used to populate the N301, N302, N401, N402, N403

Parameters:

- Document (Order No Invoice etc.) (EDOC)–Input (SYEDOC value from F47036)
- Type – Address Number (ANTY)–Input (Hard Coded value, for example “1” for Bill To or “2” for Ship To)
- Company – Key (EKCO)–Input (SYEKCO value from F47036)
- Name – Mailing (MLNM)–Output (N302)
- Address Line 1 (ADD1)–Output (N301) Address Line 2 (ADD2)–Output
- City (CTY1)–Output (N401) State (ADDS)–Output (N402)
- Postal Code (ADDZ)-Output (N403)
- Country (CTR)-Output (N404)

SQL Access Action: Jdef0111SelectWwmlnmWwalph from Table F0111 (Prefix WW)-Versions SOI and SOTPI

*SQL statement selects the Customer Mailing Name or DUNS number from the F0111 (Who’s Who) Table by using the input parameters defined in the fields below. The output value is used to populate element N104.

Parameters:

- Mailing Name (MLNM)–Output (N104)
- KEY: Address Number (AN8)-Input (SYSHAN value from Table F47036)
- Remark 1 (REM1)-Input (Hard Coded “Z” in our example)

SQL Access Action: JdeF4211SelectSdsoqs from Table F4211 (Prefix SD)-Version SOI

*SQL statement selects the Quantity Shipped Value (SOQS) from Table F4211 by using the input parameters defined in the fields below. The output value is used to populate element SN102.

Parameters:

Mailing Name (MLNM)–Output (N104)

Document (DOCO)-Input (SYDOCO value from Table F47036)

Order Number (KCOO)-Input (SYKCOO value from Table F47036)

Line Number (LNID)-Input (SZLNID value from Table F470371)

SQL Access Action: JdeF4104SelectIvcitm from Table F4104 (Prefix SD)-Version SOTPI

*SQL statement selects the Item Number Value (CITM) from Table F4104 by using the input parameters defined in the fields below. The output value is used to populate element LIN03.

Parameters:

Item Number (CITM)–Output (LIN03)

2nd Item Number (LITM))-Input (SZLITM from Table F470371)

Outbound 855 PO Acknowledgement Mapping

Header – F47026

Prefix SY

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	EDI Order Type (DCTO)
Transaction Set Purpose (TPUR)	Purchase Order Number (VR01)
Additional Fields	
Batch Number (EDBT)	Promised Delivery Date (PDDJ)
EDI – Transmission Date (EDDT)	Date – Requested (DROJ)
Trading Partner ID (PNID)	Carrier Code (CARS)
Address Number – Ship To (SHAN)	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Transaction Date (TRDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Address Number – Bill To (AN8)	Order Company (Order Number)
Document (Order No Invoice etc.) (DOCO) - *Qualifier "VN"	

Notes:

Database Adapter (Data Binding): Name:

JDE.855.F47026/F47027

Version: PRODDTA (Production) or TESTDTA (Test)

Selection Criteria:

(Prod) SELECT * FROM PRODDTA.F47026 WHERE SYEDSP = " " AND SYEDST = '855'

(Test) SELECT * FROM TESTDTA.F47026 WHERE SYEDSP = " " AND SYEDST = '855';

Detail – F47027

Prefix SZ

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Key Company
KEY: EDI Document Number (EDOC)	KEY: EDI Line Number (EDLN)
End Use (EUSE) – Line Number	Unit of Measure (UOM)
Item Number (LITM) – Qualified by VN	Item Number (AITM)–UPC Qualified by
Additional Fields	
Batch Number (EDBT)	EDI Transmission Date (EDDT)
Trading Partner ID (PNID)	Item Description (DSC1)
Purchase Order Number (VR01)	Unit of Measure (UOM)
Line Type (LNTY)	Carrier Code (CARS)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)

Time of Day (TDAY)	Amount–Price Per Unit (UPRC)
Quantity Shipped (SOQS)	

Header Notes – F4714

Prefix ZT

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Line Number (EDLN)
KEY:EDI Document Number (EDOC)	KEY:File Name (FILE)
KEY:EDI Document Key Company	KEY:Line Number Work Order (LINS)
EDI Order Type (DCTO)	
Additional Fields	
Batch Number (EDBT)	Work Station Id (JOBN)
User ID (USER)	Date Updated (UPMJ)
Program Id (PID)	Time of Day (TDAY)
Transaction Originator (TORG)	Order Key Company (KCOO)
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	

Notes:

F4714.TXLN – Field is only 60 characters in length

SQL Access Action: JdeF0116SelectAllFields from Table F0116 (Versions 1 and 2)

*SQL statement selects the Ship To address information (Address Line 1, Postal Code, City, State and Country) from the Address by Date Table by using the input parameters defined in the fields below. The output values are used to populate elements N301, N401, N402, N403, and N404.

Parameters:

Address Line 1 (ADD1)–Output (N301)

Postal Code (ADDZ)–Output (N403) City
(CTY1)–Output (N401)

State (ADDS)–Output (N402)

Country (CTR)–Output (N404)

KEY: Address Number (SHAN)–Input (This number is the SHAN value from the Header table) KEY:

Date – Beginning Effective (EFTB)

SQL Access Action: JdeF0116SelectAllFields from Table F0116 (Versions 1 and 2)

*SQL statement selects the Bill To address information (Address Line 1, Postal Code, City, State and Country) from the Address by Date Table by using the input parameters defined in the fields below. The output values are used to populate elements N301, N401, N402, N403, and N404.

Parameters:

Address Line 1 (ADD1)–Output (N301)

Postal Code (ADDZ)–Output (N403) City
(CTY1)–Output (N401)

State (ADDS)–Output (N402)

Country (CTR)–Output (N404)

KEY: Address Number (AN8)–Input (This number is the AN8 value from the Header table)

KEY: Date – Beginning Effective (EFTB)

SQL Access Action: JdeF4074SelectAprp1Uprc from Table F4074 (Prefix AL) (Version1)

*SQL statement selects APRP1 and UPRC by using the input parameters in the fields below. The output values are used to populate the elements SAC02, SAC05, SAC08

Parameters:

KEY: Document (Order No Invoice etc.) (DOCO)–Input (SYDOCO value from Table F47026)

KEY: Line Number (LNID)–Input (SZLNID value from Table F47027)

KEY: Order Company (Order Number) (KCOO)–Input (SYKCOO value from Table F47026) KEY:

Sequence Number (OSEQ)–Input (Hard Coded, example “10”)

Commodity Class (APRP1)–Output (This value is used to find service, promotion, allowance, or charge description in the F005 table which then populates SAC02)

Amount – Price per Unit (UPRC)-Output (This value is used to get the SAC05 and SAC08 values)

*This query may be used several times, changing the ALOSEQ value. The ALOSEQ value is an arbitrary number that can be increased or decreased. In this example the ALOSEQ value links to the service, promotion, allowance, or charge description in the F0005 table.

SQL Access Action: JdeF0005SelectDL01/dl02Sy/rt/ky from Table F0005 (Version1)

*SQL statement (DL01 lookup) selects the Description from User Defined Codes Table by using the input parameters defined in the fields below. The output value is used to populate element SAC02.

Parameters:

Description (DL01)–Output (SAC02)

KEY: Product Code (SY)–Input (example: Hard Coded “45”)

KEY: User Defined Codes (RT)-Input (example: Hard Coded “P1”)

KEY: User Defined Codes (KY)-Input (ALAPRP1 value from Table F4074)

SQL Access Action: JdeF0111SelectWwrem1WwnickWwmlnmWwalph from Table F0111 (Prefix WW)- (Versions 1 and 2)

*SQL statement selects the Customer Mailing Name or DUNS number from the F0111 (Who’s Who) Table by using the input parameters defined in the fields below. The output value is used to populate element N104.

Parameters:

Mailing Name (MLNM)–Output (N104)

KEY: Address Number (AN8)-Input (SYSHAN value from Table F47026 for Ship To or SYAN8 value from Table F47026 for Bill To)

Remark 1 (REM1)-Input (Hard Coded “Z” in our example)

SQL Access Action: JdeF0101SelectAban81Abalph from Table F0101 (Version 1 and 2)

*SQL statement selects the Ship To Name from the Address Book Master Table by using the input parameters defined in the fields below. The output value is used to populate element N102.

Parameters:

Name – Alpha (ALPH)-Output (N102)

KEY: Address Number (SHAN)-Input (This number is the SHAN value from the Header table)

SQL Access Action: JdeF0101SelectAban81Abalph from Table F0101 (Version 1 and 2)

*SQL statement selects the Bill To Name from the Address Book Master Table by using the input parameters defined in the fields below. The output value is used to populate element N102.

Parameters:

Name – Alpha (ALPH)-Output (N102)

KEY: Address Number (AN8)-Input (This number is the AN8 value from the Header table)

Outbound 940 Warehouse Shipping Order Mapping

Header – F47026

Prefix SY

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	EDI Order Type (DCTO)
Transaction Set Purpose (TPUR)	Purchase Order Number (VR01)
Address Number – Ship To (SHAN)	Address Number – Bill To (AN8)
Additional Fields	
Batch Number (EDBT)	Promised Delivery Date (PDDJ)
EDI – Transmission Date (EDDT)	Date – Requested (DRQJ)
Trading Partner ID (PNID)	Carrier Code (CARS)
Zone Number (ZON)	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Transaction Date (TRDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Freight Handling Code (FRTH)	Order Company (Order Number)
Document (Order No Invoice etc.) (DOCO) - *Qualifier "VN"	
Price and Adjustment Schedule (ASN)	

Notes:

Database Adapter (Data Binding):

Name: JDE.940.F47026/F47027

Version: PRODDTA (Production) or TESTDTA (Test)

Selection Criteria:

(Prod) SELECT * FROM PRODDTA.F47026 WHERE SYEDSP = " " AND SYEDST = '940';
 (Test) SELECT * FROM TESTDTA.F47026 WHERE SYEDSP = " " AND SYEDST = '940';

Detail – F47027

Prefix SZ

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Key Company
KEY:EDI Document Number (EDOC)	KEY: EDI Line Number (EDLN)
End Use (EUSE) – Line Number	Unit of Measure (UOM)
Item Number (LITM)	Item Number (AITM)
Additional Fields	
Batch Number (EDBT)	EDI Transmission Date (EDDT)
Trading Partner ID (PNID)	Item Description (DSC1)
Purchase Order Number (VR01)	Unit of Measure (UOM)
Line Type (LNTY)	Carrier Code (CARS)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Amount–Price Per Unit (UPRC)
Quantity Shipped (SOQS)	Date – Promised Delivery (RSDJ)
Item Number – Short (ZITM)	Unit Weight (ITWT)
Unit Volume (ITVL)	

Notes:

F47027.DSC1 – Field is only 30 characters in length

SQL Access Action: JdeF0005SelectDI01/di02Sy/rt/ky from Table F0005

*SQL statement (DL02 lookup) selects the Description from User Defined Codes Table by using the input parameters defined in the fields below. The output value is used to populate element W6602.

Parameters:

Description (DL02)–Output (W6602)

KEY: Product Code (SY)–Input (example: Hard Coded “40”)

KEY: User Defined Codes (RT)–Input (example: Hard Coded “AS”) KEY:

User Defined Codes (KY)–Input (SYASN value from Table 47026)

SQL Access Action: JdeF0116SelectAllFields from Table F0116

*SQL statement selects the *Ship To* address information (Address Line 1, Postal Code, City, State and Country) from the Address by Date Table by using the input parameters defined in the fields below. The output values are used to populate elements N301, N401, N402, N403, and N404.

Parameters:

Address Line 1 (ADD1)–Output (N301)
Postal Code (ADDZ)–Output (N403) City
(CTY1)–Output (N401)
State (ADDS)–Output (N402)
Country (CTR)–Output (N404)

KEY: Address Number (SHAN)–Input (This number is the SHAN value from the Header table) KEY:
Date – Beginning Effective (EFTB)

SQL Access Action: JdeF0116SelectAllFields from Table F0116

*SQL statement selects the *Bill To* address information (Address Line 1, Postal Code, City, State and Country) from the Address by Date Table by using the input parameters defined in the fields below. The output values are used to populate elements N301, N401, N402, N403, and N404.

Parameters:

Address Line 1 (ADD1)–Output (N301)
Postal Code (ADDZ)–Output (N403) City
(CTY1)–Output (N401)
State (ADDS)–Output (N402)
Country (CTR)–Output (N404)

KEY: Address Number (AN8)–Input (This number is the AN8 value from the Header table)
KEY: Date – Beginning Effective (EFTB)

SQL Access Action: JdeF0101SelectAban81Abalph from Table F0101

*SQL statement selects the *Ship To* Name from the Address Book Master Table by using the input parameters defined in the fields below. The output value is used to populate element N102.

Parameters:

Name – Alpha (ALPH)-Output (N102)

KEY: Address Number (SHAN)–Input (This number is the SHAN value from the Header table)

SQL Access Action: JdeF0101SelectAban81Abalph from Table F0101

*SQL statement selects the *Bill To* Name from the Address Book Master Table by using the input parameters defined in the fields below. The output value is used to populate element N102.

Parameters:

Name – Alpha (ALPH)-Output (N102)

KEY: Address Number (AN8)–Input (This number is the AN8 value from the Header table)

SQL Access Action: JdeF41002SelectConvCnv1 from Table F41002

Unit of Measure information is stored in the Unit of Measure Conversion Table.

*SQL statement selects the Conversion Factor from the Items Units of Measure Conversion Factors Table by using the input parameters defined in the fields below. The output value is used in populating the N902 element.

Parameters:

Conversion Factor (CONV)–Output (N902) KEY:

Business Unit (MCU)-Input

KEY: Related Unit of Measure (RUM)–Input (SZUOM value from Detail table F4027)

KEY: Unit of Measure (UM)–Input (Hard Coded “PL”)

KEY: Item Number – Short (ITM)-Input (SZITM from Table F47027)

Outbound 850 Purchase Order Mapping

Header – F47016

Prefix SY

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	EDI Order Type (DCTO)
Transaction Set Purpose (TPUR)	Purchase Order Number (VR01)
Additional Fields	
Batch Number (EDBT)	Promised Delivery Date (PDDJ)
EDI – Transmission Date (EDDT)	Date – Requested (DRQJ)
Trading Partner ID (PNID)	Currency Code – From (CRCD)
Address Number – Ship To (SHAN)	User Id (USER)
Business Unit (MCU)	Program Id (PID)
Transaction Date (TRDJ)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Address Number – Bill To (AN8)	Order Company (Order Number)
Document (Order No Invoice etc.)	Currency Conversion Rate – Spot Rate
Reference (VR01)	Reference 2 (VR02)
Delivery Instructions Line 1 (DEL1)	Delivery Instructions Line 2 (DEL2)
Amount – Order Gross (OTOT)	Carrier Value (SYGCARS)

Notes:

Database Adapter (Data Binding):

Name: JDE.850. F47016/F47017

Version: PRODDTA (Production) or TESTDTA (Test) Selection

Criteria:

(Prod) SELECT * FROM PRODDTA.F47016 WHERE SYEDSP = “ ” AND SYEDST = ‘850’;

(Test) SELECT * FROM TESTDTA.F47016 WHERE SYEDSP = “ ” AND SYEDST = ‘850’;

Detail – F47017

Prefix SZ

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Key Company
KEY: EDI Document Number (EDOC)	KEY: EDI Line Number (EDLN)
Item Number (LITM)	Item Number (AITM)
Additional Fields	
Batch Number (EDBT)	EDI Transmission Date (EDDT)
Trading Partner ID (PNID)	Item Description (DSC1)
Reference 1 (VR01)	Unit of Measure (UOM)
Line Type (LNTY)	Reference 2 (VR02)
User Id (USER)	Program Id (PID)
Work Station Id (JOBN)	Date Updated (UPMJ)
Time of Day (TDAY)	Units – Order/Transaction Quantity
Units – Secondary Quantity Ordered	Amount – Unit Cost (PRRC)

EDI Order Address Information – F4706

Prefix ZA

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
KEY: File Name (FILE) – ‘F47011’	EDI Order Type (DCTO)
KEY: Type Address Number (ANTY) – 1: Sold-To/2: Ship-To	
Additional Fields	
Batch Number (EDBT)	Mailing Name (MLNM)
User Id (USER)	Address Line 1 (ADD1)
Program Id (PID)	Address Line 2 (ADD2)
Work Station Id (JOBN)	Address Line 3 (ADD3)
Date Updated (UPMJ)	Address Line 4 (ADD4)
Time of Day (TDAY)	Postal Code (ADDZ)
City (CTY1)	State (ADDS)

EDI Order Text Lines Header – F4714

Prefix ZT

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Line Number (EDLN)
KEY: EDI Document Number	KEY: File Name (FILE)
KEY: EDI Document Key Company	KEY: Line Number Work Order
EDI Order Type (DCTO)	
Additional Fields	

Batch Number (EDBT)	Work Station Id (JOBN)
User ID (USER)	Date Updated (UPMJ)
Program Id (PID)	Time of Day (TDAY)
Transaction Originator (TORG)	Order Key Company (KCOO)
Message Text (TXLN) – <i>Field is only 60 characters in length</i>	

EDI Order Text Lines Detail – F4715

Prefix ZD

Mandatory Fields	
KEY:EDI Document Type (EDCT)	KEY:EDI Document Number (EDOC)
KEY:EDI Document Key Company	KEY:EDI Line Number (EDLN)
KEY:File Name (FILE)	KEY: Line Number Work Order (LINS)
EDI Order Type (DCTO)	Message Text (TXLN)
Additional Fields	
Batch Number (EDBT)	User Id (USER)
Program Id (PID)	Work Station Id (JOBN)
Date Updated (UPMJ)	Time of Day (TDAY)

Notes:

F4715.TXLN – Field is only 60 characters in length

SQL Access Action: JdeF0111SelectWwrem1WwnickWwmlnmWwalph from Table F0111

*SQL statement selects the Customer Mailing Name or DUNS number from the F0111 (Who's Who) Table by using the input parameters defined in the fields below. The output value is used to populate element N104.

Parameters:

Mailing Name (WWMLNM)–Output (N104)

KEY: Address Number (AN8)-Input (SYSHAN value from Table F47016 for *Ship To* or SYAN8 value from Table F47016 for *Bill To*)

Remark 1 (REM1)-Input (Hard Coded "Z" in our example)

SQL Access Action: JdeF0101SelectAbtx2Alph from Table F0101

*SQL statement selects the Tax ID and Alpha Name from the F0101 Table by using the input parameters defined in the fields below. The output values are used to populate elements TD503 and TD505.

Parameters:

Tax ID (ABTX2)–Output (TD503)

Alpha Name (ABALPH)-Output (TD505)

Address Number (ABAN8)-Input (SYCARS value from Table F47016)

Outbound 846 Inventory Advice Mapping

Header – F47126

Prefix M1

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	EDI Order Type (DCTO)
Address Number (AN8)	EDI Line Number (EDLN)
Additional Fields	
Batch Number (EDBT)	User Id (USER)
EDI – Transmission Date (EDDT)	Program Id (PID)
Trading Partner ID (PNID)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Reference (VR01)	

Notes:

Database Adapter (Data Binding):

Name: JDE.846. F47126/F47127

Version: PRODDTA (Production) or TESTDTA (Test)

Selection Criteria:

(Prod) SELECT * FROM PRODDTA.F47126 WHERE SYEDSP = " " AND SYEDST = '846';

(Test) SELECT * FROM TESTDTA.F47126 WHERE SYEDSP = " " AND SYEDST = '846';

Detail – F47127

Prefix MJ

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
Address Number (AN8)	EDI Order Type (DCTO)
2 nd Item Number (LITM)	3 rd Item Number (AITM)
Quantity Available (TRQT)	
Additional Fields	
Batch Number (EDBT)	User Id (USER)
EDI – Transmission Date (EDDT)	Program Id (PID)
Trading Partner ID (PNID)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Reference (VR01)	

Outbound 852 Product Activity Mapping

Header – F47126

Prefix M1

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	EDI Order Type (DCTO)
Address Number (AN8)	EDI Line Number (EDLN)
Additional Fields	
Batch Number (EDBT)	User Id (USER)
EDI – Transmission Date (EDDT)	Program Id (PID)
Trading Partner ID (PNID)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Reference (VR01)	

Notes:

Database Adapter (Data Binding):

Name: JDE.846. F47126/F47127

Version: PRODDTA (Production) or TESTDTA

(Test) Selection Criteria:

(Prod) SELECT * FROM PRODDTA.F47126 WHERE SYEDSP = "" AND SYEDST = '852'; (Test) SELECT * FROM TESTDTA.F47126 WHERE SYEDSP = "" AND SYEDST = '852';

Detail – F47127

Prefix MJ

Mandatory Fields	
KEY: EDI Document Type (EDCT)	KEY: EDI Document Number (EDOC)
KEY: EDI Document Key Company	KEY: EDI Line Number (EDLN)
Address Number (AN8)	EDI Order Type (DCTO)
2 nd Item Number (LITM)	3 rd Item Number (AITM)
Quantity Available (TRQT)	
Additional Fields	
Batch Number (EDBT)	User Id (USER)
EDI – Transmission Date (EDDT)	Program Id (PID)
Trading Partner ID (PNID)	Work Station Id (JOBN)
Time of Day (TDAY)	Date Updated (UPMJ)
Reference (VR01)	

General Recommendations & Best Practices

This topic provides some general recommendations and best practices when working with the *ERP Connector* product.

Re-purpose sample Projects for your own business needs

The integrations provided in the *ERP Connector* provide a starting point for creating customized implementations that utilize non-standard JDE documents, partner-specific EDI documents, customized data transformations and process automations, and partner-specific communication adapters.

They serve as examples of EDI and JDE integrations only. They can, however, be customized for your actual business needs, and to rapidly onboard new Trading Partners. The Clarify Studio provides all of the tools necessary to customize, test, and deploy integration applications based on the default integrations included in the package.

How you re-purpose these examples depends on many factors, but the general path to customization will always reflect the creation of new objects, configuration with your environments, and the utilization of the Clarify Project Generator. This feature helps you quickly generate Projects by using an existing Project as a template. Clarify replicates the template Project and its resources, updates new resource names, re-configures enveloping and trading partner info (for EDI Projects), maintains resource dependencies, and updates other attributes and values to automatically reflect the newly generated Project.

For more information, please refer to the *Wizards and other Tools* topic in the *Clarify User Guide*.

Follow Naming Conventions

A good practice is to use a standardized method for naming Projects, Packages, and Clarify objects, but it's especially critical when employing a modular approach to designing Clarify solutions. Several rules are recommended based on common standards and experience, and provide several benefits:

- Enhance clarity in cases of potential ambiguity
- Avoid naming collisions or duplicate names
- Provide meaningful information
- At-a-glance understanding of Projects/Packages/objects, especially if you are away from developing integrations for an extended period of time
- Predictability

For more information, please refer to the *Guidelines and Recommendations* topic in the *Clarify User Guide*.

Practice Version Control

We recommend using version control to take advantage of change management, even if there is only one Clarify user, and no shared Projects. With an SVN repository for example, you can track all your changes, and your ability to revert to a previously saved state is much more robust than that of a resource's local history.

Cleo hosts an SVN repository to share Projects and resources. Please contact Cleo Support for initial registration and setup.

For more information, refer to the *Versioning & Synchronizing Projects* topic in the *Clarify v3 User Guide*.

Use the Cleo Dashboard

The Cleo Dashboard reports can also be used to view status data related to EDI transactions as demonstrated in the sample integrations as part of *ERP Connector*.

Please contact your Account Manager for more information.

Additional Information

Please refer to these topics additional information related to the ERP Connector for JDE.

Identify Numeric Fields with Implied Decimal Positions

If you refresh the Clarify database schemas for E1 (as described in the topic: [Configure ERP Connector for JDE Integrations with EnterpriseOne](#)), you can use the SQL query below to identify the numeric fields whose decimal point settings must be updated in the refreshed Clarify schemas. The query returns the numeric field names for a given table (e.g., F47011, F47036), by querying several JDE database tables and ultimately the data dictionary (F9210).

```
SELECT TDOBNM,
```

```
(Select SIMD From "OL910"."F9860"
```

```
Where SIOBNM = TDOBNM
```

```
AND SIFUNO = 'TBLE')
```

```
Table_Description,
```

```
TDSQLC,
```

```
(Select FRDSCR From "DD910"."F9202"
```

```
Where FRDTAI = TDOBND
```

```
AND FRSYR = '')
```

```
Column_Description,
```

```
(Select DRDL01 From "DD910"."F9210", "CRPCTL"."F0005"
```

```
Where FRDTAI = TDOBND
```

```
AND LTRIM(RTRIM(FROWTP)) = LTRIM(RTRIM(DRKY))
```

```
AND DRSY = 'H98' AND DRRT = 'DT')
```

```
Column_Type,
```

```
(Select FRDTAS From "DD910"."F9210"
```

```
Where FRDTAI = TDOBND)
```

```
Column_Length,
```

```
(Select FROWDI From "DD910".F9210"
```


Where FRDTAI = TDOBND)

DD_Item,

(Select FROWER From "DD910"."F9210"

Where FRDTAI = TDOBND)

Edit_Rule,

(Select FROER1 From "DD910"."F9210"

Where FRDTAI = TDOBND)

Edit_Rule_Parm1,

(Select FROER2 From "DD910"."F9210"

Where FRDTAI = TDOBND)

Edit_Rule_Parm2,

(Select Frcdec From "DD910"."F9210"

Where FRDTAI = TDOBND)

Decimal_Pos,

TDPSEQ

FROM "PY910"."F98711"

WHERE TDOBNM = 'F47011'

ORDER BY TDOBNM, TDPSEQ

8. The database catalog / schema names need to be updated to reflect the values for your system. For example "DD910", "CRPCTL", "PY901", "OL910".
9. Replace "F47011" in the last line with each JDE table identifier you need to query.

JD Edwards Table Naming Conventions

To ensure consistency and prevent duplication, JD Edwards table names follow standard naming conventions. The ERP Connector for JDE integrations observe these conventions when referencing JDE tables.

The name of a JDE table can be a maximum of eight characters and is formatted as **ABBCC** where:

A = Type of

Object **BB** =

System Code **CC**

= Type of File

A (first Digit) = defines the object
F – Table (where the data is stored)

BB (second and third digits) = the system code such as:

00 – Foundation
01 – Address Book
03 – Accounts Receivable 41 – Inventory
42 – Sales Order Management
43 – Purchasing/Procurement
47 – Electronic Commerce
55 – Reserved for Customer

CC (fourth and fifth digits) = the type of file (JD Edwards transaction ID Number that is unique for the various EDI standard documents)

1– Master
2– Balances
04 – Invoices
06 – Address Information 11 – Detail
14 – Text Lines (Header) 15 – Text Lines (Detail)

An optional sixth and/or seventh digit can provide more information about the file:

1 – Header (Inbound)
2– Detail (Inbound)
3– SDQ Detail (Inbound)
6 – Header (Outbound)
61– Additional Header (Outbound)
7 – Detail (Outbound)
71– Additional Detail Outbound
8 – SDQ (Outbound)

Examples:

F47011 – PO Header Inbound		F47016 – PO Header Outbound	
F	Table	F	Table
47	Electronic Commerce	47	Electronic Commerce
01	Master	01	Master
1	Header – Inbound	6	Header – Outbound

Summary of Inbound Transaction Tables:

Table	Naming Convention
Header/Primary	F47__1
Detail	F47__2
Shipment, Destination, Quantity (SDQ)	F47__3
Address Information	F4706
Header Text	F4714
Detail Text	F4715
Other (as required)	F47__4

Summary of Outbound Transaction Tables:

Table	Naming Convention
Header/Primary	F47__6
Detail	F47__7
SDQ	F47__8
Other (as required)	F47__9
Header Text	F4714
Detail Text	F4715

Field Types

The maps (Clarify rulesets) included with the ERP Connector observe EnterpriseOne application usage conventions when updating fields in the database tables that constitute the JDE Data Interface for Electronic Data Interchange. You will also need to observe these conventions as you customize or create maps that use the JD Edwards interface tables:

Key Fields: A key field holds a data value that identifies a record in a database table. Account number, product code and customer name are typical key fields. If designated as a primary key, the key value

must be unique within the table column that contains it. When populating data for new records, your maps must supply unique values for these fields.

Mandatory Fields: The JDE E1 application designates some fields as mandatory. When populating data for new records, your maps must supply non-null values for mandatory fields.

Additional Fields: These are fields that are not defined as mandatory by JDE E1 but should be populated in order to support EDI processing. For example, maps used in inbound order processes must supply VR01 (PO Number) and PNID (EDI Trading Partner).

General Recommendations & Best Practices

This topic provides some general recommendations and best practices when working with the *ERP Connector* product.

Re-purpose sample Projects for your own business needs

The integrations provided in the *ERP Connector* provide a starting point for creating customized implementations that utilize non-standard JDE documents, partner-specific EDI documents, customized data transformations and process automations, and partner-specific communication adapters.

They serve as examples of EDI and JDE integrations only. They can, however, be customized for your actual business needs, and to rapidly onboard new Trading Partners. The Clarify Studio provides all of the tools necessary to customize, test, and deploy integration applications based on the default integrations included in the package.

How you re-purpose these examples depends on many factors, but the general path to customization will always reflect the creation of new objects, configuration with your environments, and the utilization of the Clarify Project Generator. This feature helps you quickly generate Projects by using an existing Project as a template. Clarify replicates the template Project and its resources, updates new resource names, re-configures enveloping and trading partner info (for EDI Projects), maintains resource dependencies, and updates other attributes and values to automatically reflect the newly generated Project.

For more information, please refer to the *Wizards and other Tools* topic in the *Clarify User Guide*.

Follow Naming Conventions

A good practice is to use a standardized method for naming Projects, Packages, and Clarify objects, but it's especially critical when employing a modular approach to designing Clarify solutions. Several rules are recommended based on common standards and experience, and provide several benefits:

- Enhance clarity in cases of potential ambiguity
- Avoid naming collisions or duplicate names
- Provide meaningful information
- At-a-glance understanding of Projects/Packages/objects, especially if you are away from developing integrations for an extended period of time
- Predictability

For more information, please refer to the *Guidelines and Recommendations* topic in the *Clarify User Guide*.

Practice Version Control

We recommend using version control to take advantage of change management, even if there is only one Clarify user, and no shared Projects. With an SVN repository for example, you can track all your changes, and your ability to revert to a previously saved state is much more robust than that of a resource's local history.

Cleo hosts an SVN repository to share Projects and resources. Please contact Cleo Support for initial registration and setup.

For more information, refer to the *Versioning & Synchronizing Projects* topic in the *Clarify v3 User Guide*.

Use the Cleo Dashboard

The Cleo Dashboard reports can also be used to view status data related to EDI transactions as demonstrated in the sample integrations as part of *ERP Connector*.

Please contact your Account Manager for more information.

Additional Information

Please refer to these topics additional information related to the ERP Connector for JDE.

Identify Numeric Fields with Implied Decimal Positions

If you refresh the Clarify database schemas for E1 (as described in the topic: [Configure ERP Connector for JDE Integrations with EnterpriseOne](#)), you can use the SQL query below to identify the numeric fields whose decimal point settings must be updated in the refreshed Clarify schemas. The query returns the numeric field names for a given table (e.g., F47011, F47036), by querying several JDE database tables and ultimately the data dictionary (F9210).

```
SELECT TDOBNM,
```

```
(Select SIMD From "OL910"."F9860"
```

```
Where SIOBNM = TDOBNM
```

```
AND SIFUNO = 'TBLE')
```

```
Table_Description,
```

```
TDSQLC,
```

```
(Select FRDSCR From "DD910"."F9202"
```

```
Where FRDTAI = TDOBNM
```

```
AND FRSYR = ' ')
```

```
Column_Description,
```

```
(Select DRDL01 From "DD910"."F9210", "CRPCTL"."F0005"
```

```
Where FRDTAI = TDOBNM
```

```
AND LTRIM(RTRIM(FROWTP)) = LTRIM(RTRIM(DRKY))
```

```
AND DRSY = 'H98' AND DRRT = 'DT')
```

```
Column_Type,
```

```
(Select FRDTAS From "DD910"."F9210"
```

```
Where FRDTAI = TDOBNM)
```

```
Column_Length,
```

```
(Select FROWDI From "DD910"."F9210"
```

Where FRDTAI = TDOBND)

DD_Item,

(Select FROWER From "DD910"."F9210"

Where FRDTAI = TDOBND)

Edit_Rule,

(Select FROER1 From "DD910"."F9210"

Where FRDTAI = TDOBND)

Edit_Rule_Parm1,

(Select FROER2 From "DD910"."F9210"

Where FRDTAI = TDOBND)

Edit_Rule_Parm2,

(Select Frcdec From "DD910"."F9210"

Where FRDTAI = TDOBND)

Decimal_Pos,

TDPSEQ

FROM "PY910"."F98711"

WHERE TDOBNM = 'F47011'

ORDER BY TDOBNM, TDPSEQ

Note:

- *The database catalog / schema names need to be updated to reflect the values for your system. For example "DD910", "CRPCTL", "PY901", "OL910".*
- *Replace "F47011" in the last line with each JDE table identifier you need to query.*

JD Edwards Table Naming Conventions

To ensure consistency and prevent duplication, JD Edwards table names follow standard naming conventions. The ERP Connector for JDE integrations observe these conventions when referencing JDE tables.

The name of a JDE table can be a maximum of eight characters and is formatted as **ABBCC** where:

A = Type of Object

BB = System Code

CC = Type of File

A (first Digit) = defines the object

- **F** – Table (where the data is stored)

BB (second and third digits) = the system code such as:

00 – Foundation

01 – Address Book

03 – Accounts Receivable

41 – Inventory

42 – Sales Order Management

43 – Purchasing/Procurement

47 – Electronic Commerce

55 – Reserved for Customer

CC (fourth and fifth digits) = the type of file (JD Edwards transaction ID Number that is unique for the various EDI standard documents)

01 – Master

02 – Balances

04 – Invoices

06 – Address Information

11 – Detail

14 – Text Lines (Header)

15 – Text Lines (Detail)

An optional sixth and/or seventh digit can provide more information about the file:

1 – Header (Inbound)

2 – Detail (Inbound)

3 – SDQ Detail (Inbound)

6 – Header (Outbound)

61– Additional Header (Outbound)

7 – Detail (Outbound)

71– Additional Detail Outbound

8 – SDQ (Outbound)

F47011 – PO Header Inbound		F47016 – PO Header Outbound	
F	Table	F	Table
47	Electronic Commerce	47	Electronic Commerce
01	Master	01	Master
1	Header – Inbound	6	Header – Outbound

Summary of Inbound Transaction Tables:

Table	Naming Convention
Header/Primary	F47__1
Detail	F47__2
Shipment, Destination, Quantity (SDQ)	F47__3
Address Information	F4706
Header Text	F4714
Detail Text	F4715
Other (as required)	F47__4

Summary of Outbound Transaction Tables

Table	Naming Convention
Header/Primary	F47__6
Detail	F47__7

SDQ	F47__8
Other (as required)	F47__9
Header Text	F4714
Detail Text	F4715

Field Types

The maps (Clarify rulesets) included with the ERP Connector observe EnterpriseOne application usage conventions when updating fields in the database tables that constitute the JDE Data Interface for Electronic Data Interchange. You will also need to observe these conventions as you customize or create maps that use the JD Edwards interface tables:

Key Fields: A key field holds a data value that identifies a record in a database table. Account number, product code and customer name are typical key fields. If designated as a primary key, the key value must be unique within the table column that contains it. When populating data for new records, your maps must supply unique values for these fields.

Mandatory Fields: The JDE E1 application designates some fields as mandatory. When populating data for new records, your maps must supply non-null values for mandatory fields.

Additional Fields: These are fields that are not defined as mandatory by JDE E1 but should be populated in order to support EDI processing. For example, maps used in inbound order processes must supply VR01 (PO Number) and PNID (EDI Trading Partner).

