

EDI in OLE Technical Document

Table of Contents

- [Purpose](#)
- [Dependencies](#)
- [Data Model](#)
- [Service Interface Design](#)
 - [EDI for Transmission](#)
 - [EDI for Receiving](#)
 - [Order Record Import](#)
 - [Invoice Import](#)
- [Service Interface Design \(REST/SOAP\)](#)
- [User Interface Design](#)
- [Data Importing](#)
- [Data Exporting](#)
- [Workflow](#)
- [System Parameters](#)
- [Roles and Permissions](#)

Purpose

Electronic Data Interchange (EDI) is an electronic communication method that provides certain well defined standards for exchanging data such as purchase orders, invoice, etc. through any electronic means. In OLE, EDI is primarily used to transfer purchase orders to vendors. It is also used as a way to receive invoices and order record from vendors, electronically.

The document would detail how EDI is being used in OLE and any other information such as configurations involved.

Dependencies

Though the EDI related data (Purchase Orders, Invoices, etc.) resides in Database tables, EDI doesn't have any tables of its own.

Data Model

In OLE, EDI is used as a transmission format for Purchase orders and a format accepted to load Purchase order and Invoices through the Batch Processes. The transmission format data resides in the Vendor related tables where the transmission format for each Vendor is recorded. In the Batch Process, EDI format is an accepted file extension which would be processed to retrieve order or invoice related information and the respective documents created automatically.

ole_pur_po_load_sum_t	ole_vndr_trans_frm_t
PO_LOAD_SUM_ID DECIMAL(10,0)	OLE_VNDR_TRANS_FRMT_ID DECIMAL(8,0)
PRNCPL_ID VARCHAR(45)	OLE_VNDR_TRANS_FRMT VARCHAR(200)
PO_LOAD_SCCSS_CNT DECIMAL(5,0)	VER_NBR DECIMAL(8,0)
PO_LOAD_FAIL_CNT DECIMAL(5,0)	OBJ_ID VARCHAR(36)
PO_LOAD_TOT DECIMAL(5,0)	ROW_ACT_IND VARCHAR(1)
PO_LOAD_PO_TOT DECIMAL(5,0)	
PO_LOAD_DESC VARCHAR(2000)	
FILE_NM VARCHAR(200)	
BAT_PRCS_PRF_ID VARCHAR(40)	
OBJ_ID VARCHAR(36)	
VER_NBR DECIMAL(8,0)	
FDOC_NBR VARCHAR(14)	
EDI_FILE_NM VARCHAR(200)	
PO_LOAD_CREATE_DATE DATETIME	
Indexes	Indexes

The *ole_vndr_trans_frm_t* table contains data on vendor transmission format which is linked with Vendors and the transmission of the PO happens accordingly. EDI files so generated can be sent over email or uploaded through FTP by providing the necessary credentials in OLE. The *ole_pur_po_load_sum_t* table contains the EDI_FILE_NM column where the name of the EDI file name loaded is saved while order records are loaded through the Batch Process.

Service Interface Design

OLE primarily uses EDI as an electronic communication format to transmit purchase orders generated through OLE to Vendors. It also uses EDI as a format to accept both order details and invoice details from Vendors electronically.

EDI for Transmission

Once the purchase order passes through its routing and is approved, the *completePurchaseOrder* method of the *PurchaseOrderServiceImpl* class is called. The method sets all the current and pending indicators for approved purchase order documents. It then calls the *initiateTransmission* method to transmit the purchase order as per the transmission type of the Vendor.

The *initiateTransmission* method creates the EDI file name and the path in which it will exist. The path is configured in the *olefs-config-defaults.xml* under the param name, staging.directory. The *savePurchaseOrderEdi* method is called to create and save the EDI file into the directory. The *saveEdi* method of the *PurchaseOrderEdi* class is called which invokes the *createEdiFile* method to create an EDI document and save it into a file.

The *createEdiFile* method retrieves data from the parameters and builds the EDI file.

EDI for Receiving

Order Record Import

When the Batch Process record is processed, the *start* method of the *AbstractIngestProcessor* is called. The *start* method checks if there are both the EDI and MARC file or just the MARC file and calls the relevant method. If the EDI file is also present the *start* method calls the *marcEdiprocess* method. The *marcEdiprocess* method is for building Order Record values based on ingested MARC and EDI files and calls the *fetchOleOrderRecordForMarcEdi* method of *OleOrderRecordService* class.

The *getMatchingTxRecord* method uses the line item order data from EDI and compares it with the MARC record to match them. After matching, the *build* method of the *OleTxRecordBuilder* class builds and returns a POJO with details from the line item order data. Further details are added and an order record object is built. Finally, the *populateOrderRecordForValidBFN* method of the *AbstractIngestProcessor* class uses the order record object to get the data into OLE.

Invoice Import

When the Batch Process record is processed, depending on the file format, *convertRawDataToXML* method in *BatchProcessInvoiceIngest* class uses *byPassLogicForPreProcess* method and *convertToXml* method in *OLEINVConverter* class for MARC and EDI respectively. This XML data is converted to POJO using the *convertXMLToPojos* method. The POJO so converted is used to retrieve data and load the invoice document in OLE.

Service Interface Design (REST/SOAP)

No Web services are available currently.

User Interface Design

EDI is used in Vendor and Batch Process documents. Both use KRAD's UIF (User Interface Framework). A very good guide on the framework can be found [here](#). The Spring Beans XML used in the case of Batch Process are the *OLEBatchProcessDefinitionDocument.xml* under the *datadictionary* folder and *OLEBatchProcessDefinitionView.xml* under the *uif* folder.

Data Importing

Not Applicable.

Data Exporting

Not Applicable

Workflow

Not applicable.

System Parameters

No System Parameters are currently used.

Roles and Permissions

Not applicable.