Davico Custom Project

for Customer 1-800-Radiator

# Objective

Fulfill customer 1-800-Radiator requirements

1. Receive, convert, and import the **Purchase Orders** through our EDI module
2. Automatically export the required **ASN** data via Task Scheduler every night
3. Automatically export the required **Invoice** data via Task Scheduler every night

# Documentation

Required Specification document from 1-800-Radiator defines inbound PO file layout and outbound ASNs and Invoice file layouts. See “*1800Radiator INTEGRATION FILES FORMATS via SFTP on PREMISE 04152020.pdf”* for details.

# Specifications

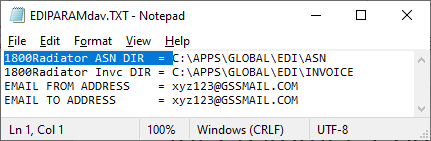
**Program 1: Convert the inbound PO csv file to our EDI file layout**

1. 1-800-Radiator will send a PO file via FTP to Davico in csv format according to “*1800Radiator INTEGRATION FILES FORMATS via SFTP on PREMISE 04152020.pdf”* on pages 5 and 6.
2. The user will run the custom GAB program from a custom menu item.
   1. The custom GAB program will find the PO files by opening a windows explorer screen for the user to point to 1800Radiator’s FTP /yourfolder/outbound directory/
   2. The GAB program will then map all PO files into a single EDI file to reside in the …\global\edi\inbound directory. Please name the resulting file with today’s date and time and a reference to the customer (i.e. – 1800Radiator.YYYYMMDD.HHMMSSSS.txt).
      1. See “*1800Radiator\_Inbound\_EDI\_Specs.xlsx*” for map details.
3. The user will run “Order Entry > File > Order Entry from EDI Transactions” in GSS and choose the file to process through the normal EDI method.

**Program 2: Write outbound ASN csv file via Task Scheduler daily**

1. Program is available for running via Task Scheduler after hours.
2. Capture all shipments created on today’s date (SHIPMENT\_HEADER, SHIPMENT\_LINES) for all 1-800-Radiator customers where SHIPMENT\_HEADER: CODE\_SORT = “EDI1800”.
3. Create required CSV file from shipment data. ***Note for all record types:*** Leave “?” fields blank.

* LineCode = 'R' => We can ignore all fields here, so send “R,?,?,?,?” where ‘?’ are blank
* LineCode = 'H' => [PO#], [today’s date yyyy-mm-dd], 0
* LineCode = 'SH' => [Total Qty Shipped], EA, [Total Weight], LB, [Carrier Name], ?, ?, [Tracking#], ?, ?, ?, ?, [ShippingDate(yyyy-mm-dd)]
* LineCode = 'O' => PO#, ?, ?
* LineCode = 'I' => [QuantityShipped], [Item#=LineUser1], [PartNumber=GSSPart], ?, EA, ?, ?, ?, ?, ?, ?
* LineCode = 'S' => [Total Lines=Record Count in the file for all record types], ?, ?

1. File Name doesn’t matter as long it doesn’t repeat itself over time; however, the extension of this file needs to be “.ASN”. (i.e. - 1800Radiator.YYYYMMDD.HHMMSSSS.ASN)
2. File needs to be saved to the 1800Radiator FTP folder (yourfolder/inbound/ASN).
   * To determine the directory, read the …global\files\EDIPARAMccc.TXT file for the “1800Radiator ASN DIR = ” record contents and use contents after “ = “.
   * 

**Program 3: Write outbound invoice csv file via Task Scheduler daily**

1. Program is available for running via Task Scheduler after hours.
2. Capture all invoices created on today’s date (ORDER\_HIST\_HEAD, ORDER\_HIST\_LINES) for all 1-800-Radiator customers where ORDER\_HIST\_HEAD: CODE\_SORT = “EDI1800”.
3. Create required CSV file from invoice data. ***Note for all record types:*** Leave “?” fields blank.

* LineCode = 'R' => We can ignore all fields here, so send “R,?,?” where ‘?’ are blank
* LineCode = 'H' => “Davico Mfg”, [Invoice for PO#], [today’s date], [PO#]
* LineCode = 'D' => [Item#=LineUser1], [PartNumber=GSSPart], [QuantityInvoiced/Shipped], ?, [AmountInvoiced, price X qty]
* LineCode = 'S' => ?, ?, ?, ? , Freight Cost

1. File Name doesn’t matter as long it doesn’t repeat itself over time; however, the extension of this file needs to be “.INV”. (i.e. - 1800Radiator.TodaysDateYYYYMMDD.TimeHHMMSSSS.INV)
2. File needs to be saved to the 1800Radiator FTP folder (yourfolder/inbound/Invoice).
   * To determine the directory, read the …global\files\EDIPARAMccc.TXT file for the “1800Radiator Invc DIR =” record contents and use contents after “ = “.
   * 