

World Premium Points of Interest Getting Started Guide



Version: 1.0

Table of Contents

INTRODUCTION	3
1. Preface.....	3
2. Data Characteristics.....	3
3. Loading the data into RDMS Databases	3
Oracle.....	3
PostgreSQL.....	4
SQL Server.....	6
4. Loading the data into GIS Systems and spatializing.	8
Loading POI Data into MapInfo Professional.....	8
5. Data Transformation and Associated Tools.....	10
How to Convert POI data using MIPro.....	11
How to convert POI data using GDAL	11
How to convert POI data using FME	11

INTRODUCTION

1. Preface

This guide intends to offer some hints and tips to get up and running with Pitney Bowes World Premium Points of Interest Data.

Due to the size of the POI data, it is suggested to follow the recommended size considerations for the software that is being used.

2. Data Characteristics

The data characteristics for the points of interest data is stated below

- Typical volume sizes: ~ 20KB to 14GB
- Format: Flat text file
- Delimiters: Pipe (|) delimited Separator None
- Header: Yes
- Encoding: UTF 8
- Line Feed: Linux

The Reference schema can either be download from [HERE](#) or can be reviewed through the product documentation available with the product.

NB: The world premium points of interest data is not indexed but PB_ID is the unique identifier for the data and should be used as a primary key.

3. Loading the data into RDMS Databases

Oracle

To load data into Oracle you will need

1. A machine with Oracle client installed.
You can download it from here...
<http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html>
 2. SQL Loader tool (Packaged with Oracle client)
 3. Create Table SQL file.
 4. Oracle CTL template file.
 5. Indexes and Spatial script file
- Create the table structure in the database. A template create table script can be found in the [POI_CREATE_TABLE.SQL file](#). Make sure to change the table name in the template to the country you wish to load.
 - Once the table structure is created you will need to take a copy of the Oracle CTL template [POI_ORACLE_CTL_TEMPLATE.CTL](#) and add it to the same directory

as your POI data file. Open the CTL file in a text editor and add in the name of the POI txt file and Oracle destination table by replacing the <country_code> tag with the ISO3 country code of the file you are loading.

- Open the command prompt (start menu type cmd in the search bar). Enter the [SQL LOADER SCRIPT](#) with user details etc.

i.e. sqlldr userid=<username>/<password>

@<database connection> control=<ctl file name> log=<log file name>

- After running this you should see the data loading into the database

```
SQL*Loader: Release 11.2.0.3.0 - Production on Wed Feb 15 14:58:55 2017
Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.
Save data point reached - logical record count 1000000.
Save data point reached - logical record count 2000000.
Save data point reached - logical record count 3000000.
Save data point reached - logical record count 4000000.
Save data point reached - logical record count 5000000.
Save data point reached - logical record count 6000000.
Save data point reached - logical record count 7000000.
Save data point reached - logical record count 8000000.
Save data point reached - logical record count 9000000.
Save data point reached - logical record count 10000000.
Save data point reached - logical record count 11000000.
Save data point reached - logical record count 12000000.
Save data point reached - logical record count 13000000.
Save data point reached - logical record count 14000000.
Save data point reached - logical record count 15000000.
Save data point reached - logical record count 16000000.
Save data point reached - logical record count 17000000.
Save data point reached - logical record count 18000000.
Save data point reached - logical record count 19000000.
Save data point reached - logical record count 20000000.
Save data point reached - logical record count 21000000.
Save data point reached - logical record count 22000000.
Save data point reached - logical record count 23000000.
Save data point reached - logical record count 24000000.
Load completed - logical record count 24198097.
```

- Once this is finished you may choose to add a Primary Key
ALTER TABLE POI_XXX add constraint POI_XXX_PK primary key("PB_ID");
- And add a geometry column to the table.

Please find the script to add geometry [HERE](#)

PostgreSQL

To load data into PostgreSQL you will need

1. A machine with PostgreSQL installed, preferably version 9.2 onwards.
You can download it from here- <https://www.postgresql.org/download/>
 - I. Download and Install PostgreSQL.
 - II. At the end you'll be prompted to install additional items using StackBuilder (window will pop up)
 - III. It is recommended to select yes and choose PostGIS extension (To create a spatial database)
 - IV. Finish installing

2. Create Table SQL file

Method 1- Create table Script

1. Create the table structure in the database. A template create table script can be found in the [POI CREATE TABLE.POSTGRESQL file](#). Make sure to change the table name in the template to the country you wish to load
2. Once the table structure is created, you can copy the data using PostgreSQL Copy command. E.g.

```
copy poi_<country_code> from <path to txt file> delimiter '|' quote e'\b' csv header;
```

- |table_name| - Name of created table using the 'Create table script'
- |Path to file| - Path to file location
- Delimiter – Define the delimiter, for wppoi it is Vertical bar
- CSV header- Ignores the first line as it has headers.

```
Working example- copy poi_usa from 'C:\wppoi_usa.txt' delimiter '|' quote e'\b' csv header;
```

Method 2- Use the data load PostgreSQL function.

1. Open a new SQL query window (In PGAdmin III press Ctrl-E which will give you a query window)
2. Open up [POI_POSTGRES_LOADER_SCRIPT.sql](#);
3. Select the contents in notepad or any other text editor and copy contents using Ctrl-C
4. Go to the query window that you've opened in above step and paste
5. Press Ctrl-A and then click the green arrow button and this will execute the code and build the function
6. You should get a message in the Output pane stating the Query returned successfully with no result in x ms

Method 3- Import CSV file into table using pgAdmin III.

1. Use pgAdmin GUI tool to import CSV file into a table in PostgreSQL server
2. Right click on the table, choose the import
3. Browse to the file, choose format as TXT and click the Columns tab.
4. Select the columns to import
5. Check the Header checkbox because our file has a header, choose Delimiter (|) and click Import button.
6. Finally, when the import process completes, click the Done button.

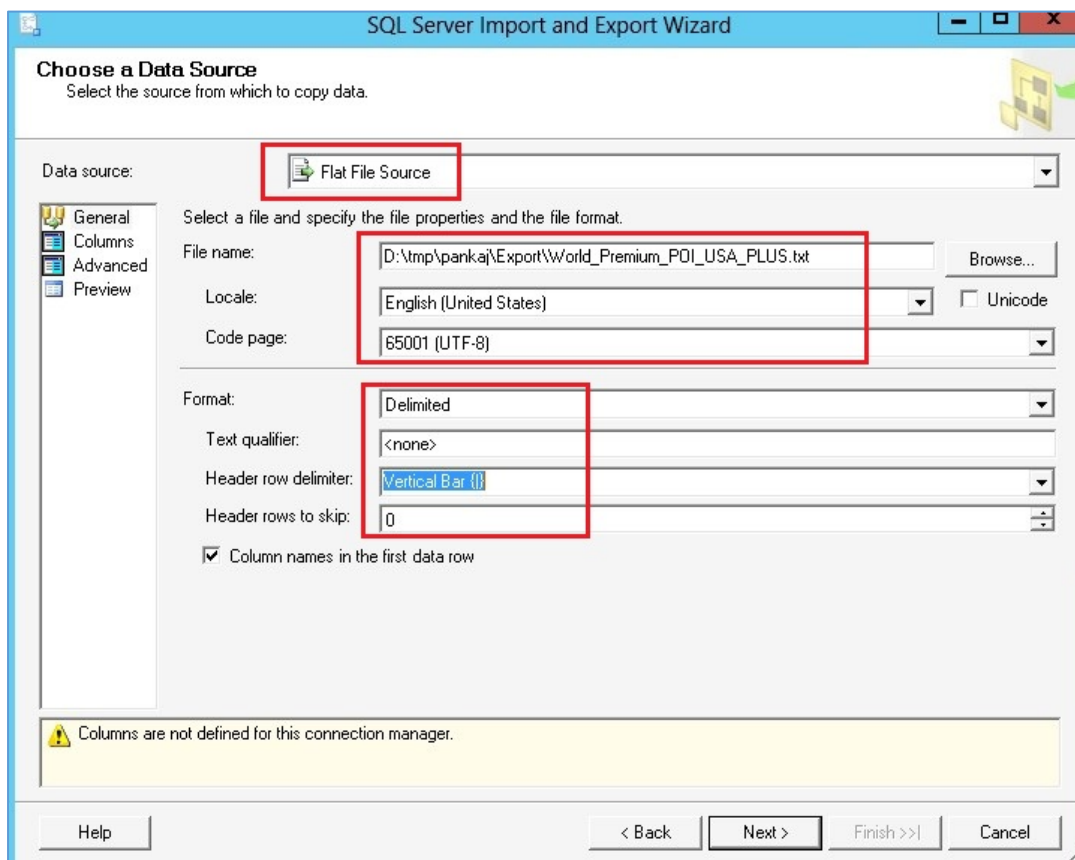
Spatializing the data-

1. Enable PostGIS extension in the database
2. Add Geometry column to the table
3. Please find the script to add geometry [HERE](#)

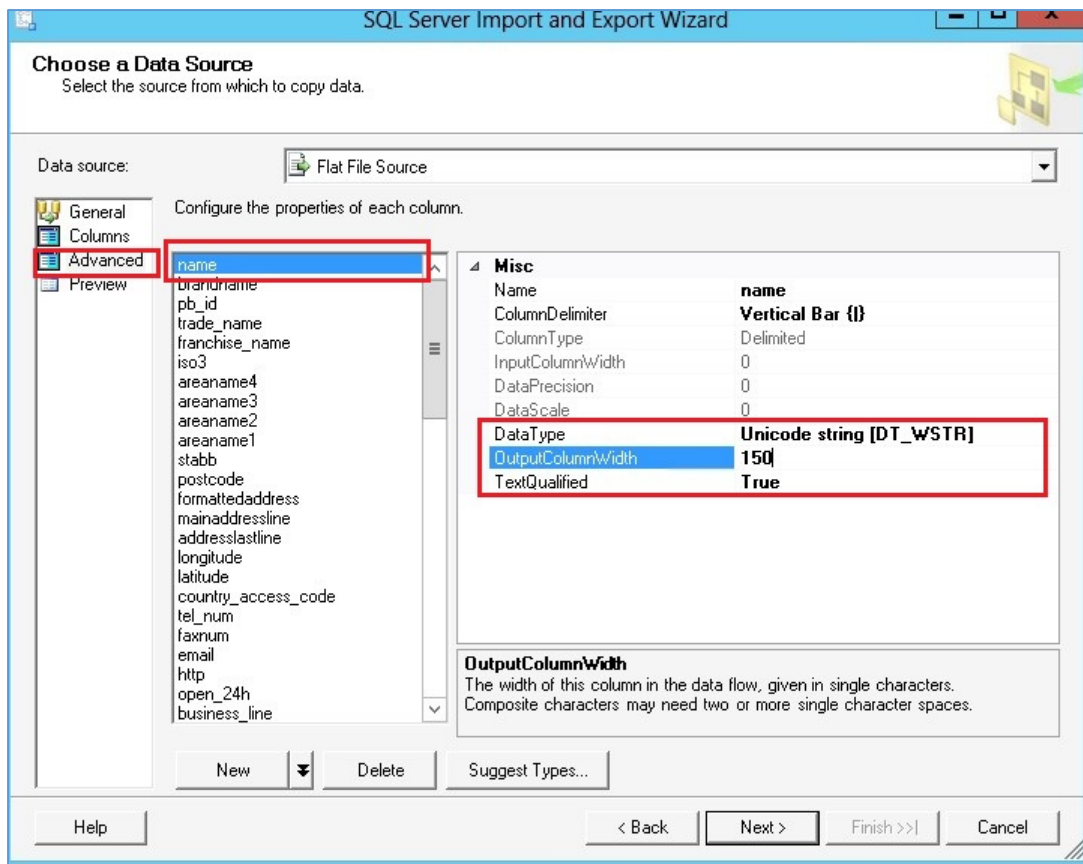
SQL Server

There are two ways to load the data into SQL server using Import and Export wizard or Bulk Insert utility

- A. To import or export data by using the SQL Server Import and Export Wizard
 - a) Start the SQL Server Import and Export Wizard.
 - b) On the corresponding wizard pages, select a data source and a data destination.
 - c) Set the options for the type of destination that you selected.
 - d) Select the source as Flat File, you can specify the following
 - Specify the row delimiter as PIPE in the source file.



- e) Change the mappings between source and destination columns, or change the metadata of destination columns:
 - Map source columns to different destination columns.
 - Change the data type in the destination column.
 - Set the length of columns with character data types.
 - Set the precision and scale of columns with numeric data types.
 - Specify whether the column can contain null values.



f) Save and run a package.

B. To import data by using the SQL Bulk Insert Utility with Format File:-

- g) Create the table structure in the database. A template create table script can be found in the POI_CREATE TABLE_LOAD_SQL_SERVER [FILE](#). Make sure to change the table name in the template to the country you wish to load
- h) 'Format' FMT file is created and available [HERE](#) and which can be invoked by using 'Bulk Insert'

Note: The FMT was created in SQL server 2014 using (Microsoft SQL Server\110). The version of the bcp utility (Bcp.exe) used to read a format file must be the same as, or later than the version used to create the format file. For example, SQL Server 2014bcp can read a version 10.0 format file, which is generated by SQL Server 2008bcp, but SQL Server 2008bcp cannot read a version 11.0 format file, which is generated by SQL Server 2014bcp.

If new FMT file needs to be create please follow below setups
(<https://msdn.microsoft.com/en-us/library/ms191516.aspx>):

- Command to create FMT file , run this command on Command window prompt:
Bcp [<database name>].[db].<table_name> format nul -c -f path*.Fmt -S <server name> -T

-T: - Specifies that the bcp utility connects to SQL Server with a trusted connection using integrated security. If -T is not specified, you must specify -U and -P to successfully log in.

- Use the below command for Bulk insert-

```
BULK INSERT
[ database_name ]. [ schema_name ] . [ table_name ]
FROM 'data_file'
WITH (
  FORMATFILE = '*.Fmt',
  FIRSTROW = 2,
  BATCHSIZE = 1000000,
  ROWS_PER_BATCH = 500000,
  ERRORFILE = 'path\Error_Log3.log',
  MAXERRORS = 500000
)
```

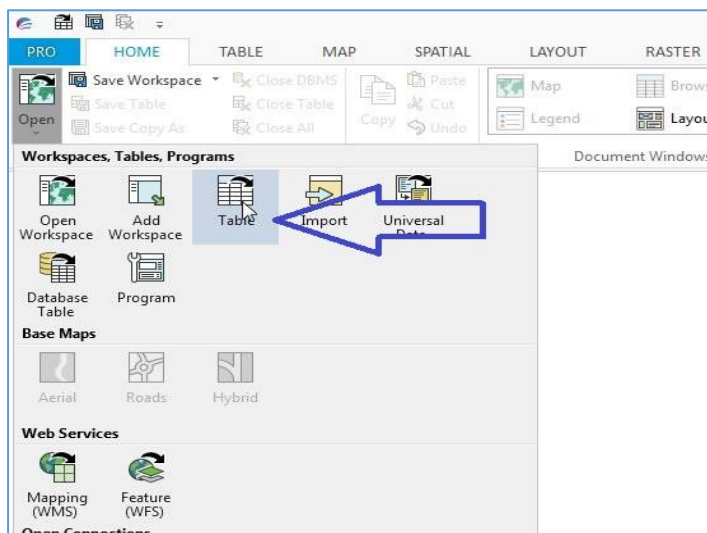
4. Loading the data into GIS Systems and spatializing.

Loading POI Data into MapInfo Professional

To load data into [MapInfo Professional](#) you must *have* version *15.0 or Higher*

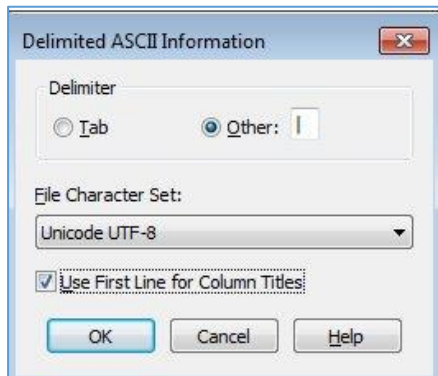
Open WPPOI data into MIPro

- a) Start Mapinfo Professional
- b) Go to 'HOME' tab and click on 'Open Table'

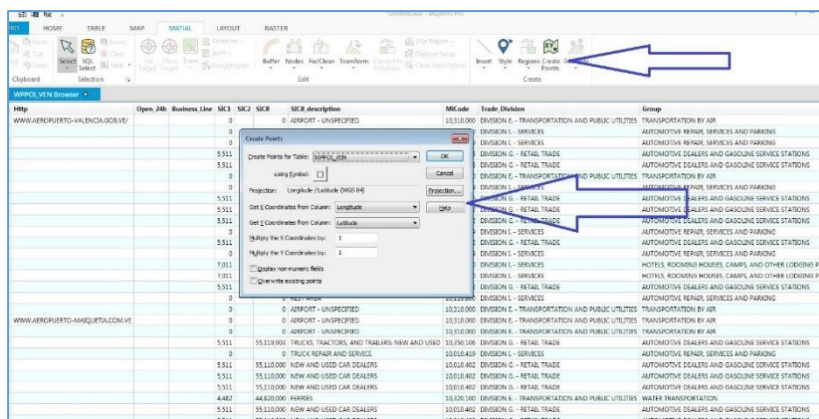


- c) Select File Type "Delimited ASCII (*.txt)" and select WPPOI text file from your local directory.

- d) Select Delimiter “Other” and type ‘|’ in text box. Select File Character Set as “Unicode UTF-8” from drop down and Check on “Use First Line for Column Titles”. Click OK.



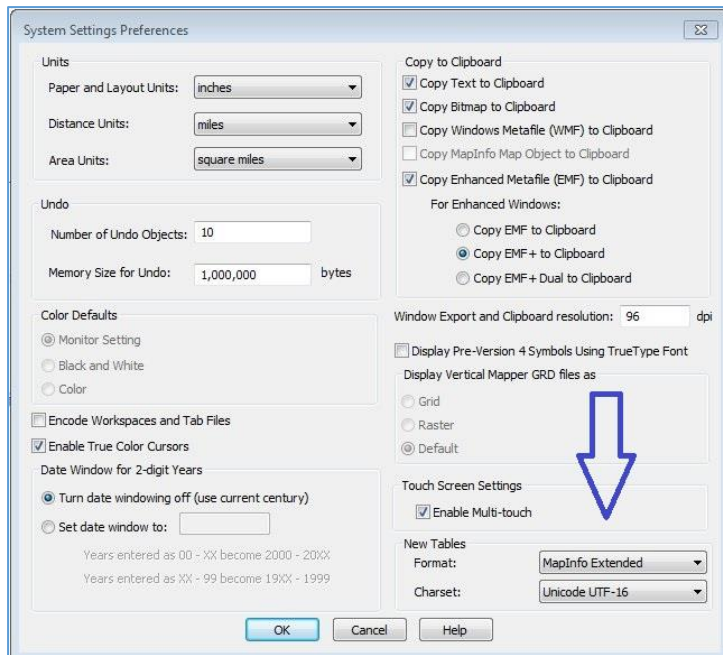
- e) Once this is finished you may view WPPOI data in the browser window.
 f) Go to “Spatial” TAB and select “Create Points”
 g) Select following options in “Create Point” Dialog box
 Create Point for Table → WPPOI Table Name
 Using Symbol → select Icons
 Projection → Longitude / Latitude (WGS 84)
 Get X Coordinates from Column → Longitude
 Get Y Coordinates from Column → Latitude



- h) Go To “Map” tab and select Click on Map

Step to Open More than 2GB WPPOI data into MIPRO

- i) Go To “Pro” tab
 j) Select “System Setting” from “Option Menu”
 k) Select “Mapinfo Extended” from “Format” Drop Down
 l) Follow Step 1 – 8 from above section (Step to Open WPPOI data into MIPRO)



More tutorial and references:

<http://www.pitneybowes.com/us/location-intelligence/geographic-information-systems/mapinfo-pro.html>

<https://www.youtube.com/user/mapinfo>

<https://www.youtube.com/watch?v=tovr-g1FHNE>

<https://www.youtube.com/watch?v=FMDRcdu2Xo>

5. Data Transformation and Associated Tools.

The POI data can be used in different formats available but needs to be converted as apt for the tool or the application that will use it. There are many available open source tools and applications available for conversion.

In this section we will provide some examples on how to convert the POI files into four different formats with four different tools.

Formats:

- GeoJson
- SHP
- Native X
- GeoPackage

Tools:

- MapInfo Professional
- FME
- GDAL

How to Convert POI data using MIPro.

To convert POI to either Extended Tab or Geopackage you can follow the steps in [step 4](#)

You can then use the save as command to save the data to your chosen format

How to convert POI data using GDAL

- To convert txt to above formats we have to first convert it to csv and we need to create *.vrt file for conversion which can be found [HERE](#). We need to change the path in the commands below as per your local paths.

To TAB

- ogr2ogr -f "Mapinfo File" "G:\XXX\out.TAB" "G:\XXX\Sample.vrt"

To SHP

- ogr2ogr -f "ESRI Shapefile" "G:\XXX\out.shp" "G:\XXX\Sample.vrt"

To GEOJSON

- ogr2ogr -f "GeoJSON" "G:\XXX\out.json" "G:\XXX\Sample.vrt"

To GEOPACKAGE

- while creating the .gpkg file it will show an error due to the column name "GROUP" which can be removed by simply renaming the column. For example – "GROUP_1".
- ogr2ogr -f "GPKG" "G:\XXX\out.gpkg" "G:\XXX\Sample.vrt"

How to convert POI data using FME

FME can convert Text files to GeoJson, Esri Shape, GeoPackage and Mapinfo Tab but does not currently support writing to MapInfo Extended Tab format. If converting to tab, the data may need to be split to avoid file size constraints.

While converting to SHP some column name may be truncated due to DBF restrictions. It supports field names up to 10 characters in length. Column names should be renamed accordingly.

An example FME workbench to convert the files into various formats can be found [HERE](#)