pitney bowes

Spectrum™ Spatial Map Uploader
Version 12.0 SP1

Spectrum Spatial Map Uploader Guide (Pre-MapInfo Pro 16)

Welcome to the legacy version of the Spectrum Spatial Map Uploader for use with Spectrum™ Technology Platform MapInfo® Pro versions prior to version 16.

Rather have a PDF? **Download the Map Uploader Guide PDF**

About the Spectrum Spatial Map Uploader

The Map Uploader is a tool that allows you to upload a map workspace, along with the layers and tables it uses from MapInfo Professional into Spectrum Spatial. It can be downloaded from the Spectrum Spatial section of the Welcome Page, under **Map Uploader** on the Utilities tab.

The layers in the map workspace can include style overrides or thematic maps that have been created in MapInfo Professional and these will be included in the upload.

The uploader creates a named map for the workspace and separate named layers and named tables.

The actual data is not uploaded as it remains in its original location and referenced by the named table. The named layers contain the information on how to style or theme the data when viewed on the map.

Prerequisites

To use the Spectrum Spatial Map Uploader you need:

- MapInfo Pro 15 or above

    Note: The Map Uploader will not work for GeoPackage files using MapInfo Pro 16.

- Microsoft .NET Framework 4.5 or above (this is installed as part of MapInfo Pro)
- Spectrum Spatial 11.1
- a Spectrum user with an admin role
- for all tables, either file-based or in a supported database (Oracle, SQL Server, or PostGIS), a named connection that will be used by Spectrum Spatial to access the data source

    Note: Named connections are created in Spatial Manager. For more information on creating named connections, refer to the Spatial Manager documentation in the *Spectrum Spatial Guide*

Adding the Uploader to the MapInfo Professional Tools Menu

The Spectrum Spatial Map Uploader can be downloaded as a MapInfo Professional plugin file from the Spectrum Spatial section of the Welcome Page, under **Map Uploader** on the Utilities tab.

1. Click **Download Zip** to save the map-uploader.zip file. Unzip it to a location that is accessible to MapInfo Professional. It is recommended that you place the plugin where your other

MapInfo Professional plugins are installed, or here: `C:\Program Files(x86)\MapInfo \Spectrum Spatial Map Uploader`

2. Start MapInfo Professional.

3. Go to the **Tools** > **Tool Manager** menu in MapInfo Professional. This will open the **Tool Manager** dialog.

4. Click the **Add Tool** button and browse to the location where the Spectrum Spatial Map Uploader MBX file is located, and select "`SpectrumMapUploader.MBX`" and click **Open**.

5. Add a **Title** (Such as Spectrum Spatial Uploader) and an optional description for the tool.

6. Click **OK**.

7. Ensure that the **Loaded** and **Autoload** check boxes are ticked.

8. Click **OK**.

The Uploader will now always appear on the tools menu when you open MapInfo Professional.

To test the installation and start the Map Uploader:

1. Start MapInfo Professional.

2. Go to the **Tools** > **Tool Manager** menu in MapInfo Professional. This will open the **Tool Manager** dialog.

3. Select **Spectrum Spatial Map Uploader**.

4. Click **OK**.

> **Note:** If the plugin throws an IOException on startup, and references the **upload-settings.config_** file, remove the file from the error message (typically %appdata% \MapInfo\Spectrum Spatial Map Uploader\upload-settings.config_) and restart the application.

Configure Map Uploader to Work with SSL Using HTTPS

In order for the Map Uploader to work with SSL using HTTPS communication with the Spectrum server, you need to make a configure change in the Uploader.config file.

> **Note:** You must make the configuration change prior to loading the Map Uploader in MapInfo Pro.

To change the configuration:

1. Open the Uploader.config file in a text editor. This file is located where you unzipped the map-uploader.zip file.

2. In the SpectrumServiceSoapBinding section of the config file, modify the security section as follows:

```
<security mode="Transport">
```

```
<transport clientCredentialType="Basic" proxyCredentialType="None"
realm="DCGRealm" />
<message clientCredentialType="UserName"
algorithmSuite="Default" />
</security>
```

3. Save the Uploader.config file.

You can now load the plugin into MapInfo Pro and connect to the Spectrum^{TM} Technology Platform server with SSL using HTTPS.

Selecting Language for Spectrum Spatial Map Uploader

Spectrum Spatial Map Uploader allows you to choose a preferred language for the uploader. To achieve this, you need to specify the particular language code in the `lang` field of appSettings in `Uploader.config` file. This file is located where you unzipped the Map Uploader. By default, the `lang` field is empty and the system's default language is set.

These languages are supported:

Language	Code
Welsh	cy
Danish	da
German	de
English - Default	en
English - Australia	en-au
English - Great Britain	en-gb
Spanish	es
Finnish	fi
French	fr
Japanese	ja-JP
Dutch	nl
Portuguese	pt
Swedish	sv

Logging into the Map Uploader

After installing the Map Uploader you will need to launch it from the MapInfo Pro tools menu and login.

> **Note:** Prior to launching the Map Uploader, you must have a map open in MapInfo Pro.

To open the Spectrum Spatial Map Uploader:

1. Select **Tools > Spectrum Spatial Map Uploader** to launch it.
2. Specify the URL of the Spectrum Spatial server in the **Server** text box.
3. Enter the credentials in the User name and Password text box to login to the **Spectrum Spatial Map Uploader**. The user must already have been created and be a member of the admin roles.
4. (Optional) Select **Enable Logging** check box to to create a log file of the service requests made when uploading data. The Spectrum Spatial Map Uploader log file is located in the file: `%AppData%\Pitney Bowes\Spectrum Spatial Map Uploader\log\messages.log`.
5. Click **Login**.

Uploader Settings Tab - File Based Data

Before uploading maps that use file based data, it is necessary to ensure that the correct Named Connections and Relative path to data are added to the settings tab and saved. Spectrum uses the Named Connection and appends the relative path to the connection to access the data.

To enter local and server paths:

1. Select **Spectrum Spatial Map Uploader** > **Settings** tab.
2. Select a **Named Connection** from the drop down. The Server source folder will be displayed. This location is the data directory on the server where the Named Connection points.
3. Optionally, enter the **Relative path to data**. If the data on the server resides in the Server source folder, you can leave this field blank. If the data resides in a sub folder or the Server source folder defined by the Named Connection, you need to add the folder structure. For example, if your named connection is /NamedConnections/TestData but the actual data is located in a sub folder called World inside the TestData folder, the relative path would be /World.
4. Click Apply.

Uploading a Map to Spectrum Spatial

To upload a map to Spectrum Spatial:

1. Select **Spectrum Spatial Map Uploader** > **Uploader** tab.
2. Enter the map name in **Map Name** text box.
3. Select **Upload Map** to create a named map when uploading. If this is not selected, only named tables will be uploaded, and no map will be created.
4. Enter a path in the **Repository Path** text box to upload resources to a specific folder in Spectrum. Click Browse and select the folder location in the repository. Right click on any folder in the repository browser and you can add a new folder. Click OK when done selecting the folder. For more information on how folder names should be specified, refer to **Appendix C**.
5. Select the corresponding JDBC Named Connection for each of the tables. For file based named connections you must first configure the Map Uploader Settings Tab.
6. Click Upload.

Points to Remember:

- For all named connections (file and database) you must create these using Spatial Manager prior to using the Map Uploader. For more information on Spatial Manager, see the *Spectrum Spatial Guide*.
- When uploading tables that point to Oracle, SQL Server, or PostGIS database tables, you must pick a corresponding named connection for the type. If there are no named connections than these will be marked as unsupported.
- When uploading file based tables Native, Shape, Raster, or Grid, you must pick a corresponding file based named connection for the type. If there are no named connections than these will be marked as unsupported.
- Any unsupported tables are marked with a red cross. You can still upload the remaining tables (ticked green) but the unsupported ones will not be uploaded and will not be included in the map in Spectrum Spatial.
- When named tables are created during the upload process, they will include references to the named connections. Named tables are not created using in-line connection definitions.
- Some geometries are unsupported, these are: Collection, Multi-Point and Ellipse.
- Some unsupported tables are listed in **Appendix A**.
- Details of the named resources created are in **Appendix B**.

Appendix A- Supported Layer and Table Types

Not all layer and table types that are used in MapInfo Professional are supported in Spectrum Spatial. The following lists show those that are supported and unsupported.

Supported Table and Layer Types

These layer and table types are supported and will always be shown with a green tick.

- Image (raster) layers and tables
- Thematic Layers (Range, Individual Value and Dot Density are including as part of the source layer and define the style to be used. Graduated Theme, Bar Chart and Pie Chart thematics are uploaded as separate layers in addition to the source layer)
- Grid layers and tables
- DBF table
- Native table
- Shapefile table (The shapefile will be used rather than the TAB file)

> **Note:** A TAB file is not created via the universal translator to create a shapefile from the tools menu.

The following tables are supported only if there is a named connection available. If there are no named connections then they will be shown as unsupported. Note that the Map Uploader will not validate that table is available with the named connection chosen. It is up to the user to ensure that the connection chosen is correct for that table. The table name will be taken from the TAB file and the remaining connection details (including database scheme/owner) from the named connection.

- ODBC live Table
- ODBC linked Table
- Oracle live Table
- Oracle linked Table

Note that although Map Uploader will allow you to upload any ODBC tables, Spectrum Spatial is only tested on and supports the following RDBMS:

- Oracle
- SQL Server
- Post GIS

Unsupported Layer and Table Types

These layer and table types are not supported and will always be shown with a red cross.

- Cosmetic layer
- WMS and WMTS layers and tables
- Tile Server layers and tables
- Result table
- View table
- WFS table

- FME table
- Access Table
- ASCII table
- XLS table
- XY tables

Duplicate Name Issues

MapInfo Professional allows users to open the same table (from the same or different locations) and add it multiple times as layer in a map. The Spectrum™ Technology Platform will not upload the same layer more than once in each map as it will result in creation of duplicate layers which are not supported by Spectrum. In this scenario the Uploader will allow the first occurrence of the layer to be uploaded but will not upload the remaining ones. If you wish to upload the same layer multiple times, for example, with different style override or other settings, you can add it to a new map which can be uploaded.

Appendix B- Named Resources

When uploading a map the following 3 folders will be added to the repository below the path specified in the uploader.

- NamedMaps
- NamedLayers
- NamedTables

For any tables that are uploaded a Named Table resource will be added to the Named Tables folder. This will contain the path to the TAB file or connection details to the data source. No layer information (such as style, zoom layer settings, label settings) is held in the Named Table.

If the map is uploaded it will be created as a Named Map and added to the NamedMaps folder.

In addition a Named layer will be created for each layer. The Named Layer will be added to the NamedLayers folder.

A separate named layer is created for each Named Map and Named Table pair and has a generated name as follows:

```
<Named Map Name>_NM_<Named Table Name>
```

Where labels are enabled a separate label layer will be created and where the layer has an Individual Value, Pie or Bar theme, a separate layer is created for each of the themes. The names of these follow the convention below where ID is the identifier number assigned to the layer in MapInfo Professional.

```
<Named Map Name>_NM_<Named Table Name> Labels
<Named Map Name>_NM_<Named Table Name> Graduated Symbol Theme IDxx
```

```
<Named Map Name>_NM_<Named Table Name> Bar Theme IDxx
<Named Map Name>_NM_<Named Table Name> Pie Theme IDxx
```

Below is an example of a map that contains all of the above layer types and how the Named Resources are created.

Figure 1: Example

Appendix C- Repository Path

When uploading resources you can optionally specify a repository folder to add them to in the **Repository Path** text box. This can be an existing folder or a new folder which does not yet exist. If the folder does not exist the uploader will create it. Also, as described in **Appendix B** the uploader will create three further sub-folders called `NamedMaps`, `NamedLayers` and `NamedTables`.

If you leave the **Repository Path** blank or just use "/" then the resources are added under the root folder. The root is the folder always called "`default`". Hence in this case the uploader will add the resources to the following sub folders.

```
http://<server>:<port>/RepositoryService/repository/default/NamedMaps
http://<server>:<port>/RepositoryService/repository/default/
NamedLayers
http://<server>:<port>/RepositoryService/repository/default/
NamedTables
```

If you specify a single folder name such as "DemoMaps" then the resources are added as follows. If DemoMaps does not exist as a folder it will be created.

```
http://<server>:<port>/RepositoryService/repository/default/DemoMaps/
NamedMaps
http://<server>:<port>/RepositoryService/repository/default/DemoMaps/
NamedLayers
http://<server>:<port>/RepositoryService/repository/default/DemoMaps/
NamedTables
```

You can also specify multiple levels of folder such as "DemoMaps/EnvironmentProject". In this case resources are added as follows:

```
http://<server>:<port>/RepositoryService/repository/default/DemoMaps/
EnvironmentProject/NamedMap
http://<server>:<port>/RepositoryService/repository/default/DemoMaps/
EnvironmentProject/NamedLayer
http://<server>:<port>/RepositoryService/repository/default/DemoMaps/
EnvironmentProject/NamedTable
```

> **Note:** When specifying folders ensure that you use the forward slash "/" and not a backslash "\".