

Telemapix

For Microsoft Windows[®] Environments (as appropriate)

User Documentation

Version 1.0

Graticule
2 Blenheim Court
Leeds LS2 9AE UK

Telephone:	0113 234 4000
Fax:	0113 246 5071
International Telephone:	+44 113 234 4000
International Fax:	+44 113 246 5071
Email	sales@graticule.com
World Wide Web	http://www.graticule.com

Contents

1	Telemappix	3
1.1	Introduction	3
1.2	Importing Data	3
1.3	SQL Statements	5
1.3.1	Database Structure	5
1.3.2	SELECT FROM	5
1.3.3	WHERE	6
1.3.4	More Information	6
1.4	Common Connection strings	7
1.4.1	SQL Server	7
1.4.2	SQL Server 2005	7
1.4.3	Access	7
1.4.4	Oracle	8
1.4.5	mySQL	8
1.4.6	Interbase	8
1.4.7	IBM DB2	9
1.4.8	Sybase	9
1.4.9	Informix	9
1.4.10	Ingres	10
1.4.11	Mimer SQL	10
1.4.12	PostgreSQL	10
1.4.13	Paradox	10
1.4.14	DSN	11
1.4.15	Excel	11
1.4.16	Text	11
1.4.17	DBF/Foxpro	11
1.4.18	AS/400	11
1.4.19	Visual Foxpro	11
1.4.20	More	12

Chapter 1

Telemappix

1.1 Introduction

The Telemappix plug in allows you to use the power of SQL and Object DataBase Connectivity (ODBC) to import and 'synchronise data within PostcodeAnalyst 2.0 in an automated way. Ensuring data consistency and easy updating of changing information.

It creates what we call a 'live layer', this is a small set of XML files (which can be encrypted if a password is required to connect to the database) which can be copied from one install of PostcodeAnalyst 2 to another (so one user can create a live layer set and distribute that to other users)

Then when the data source changes (for example customers are added to a 'customers' live layer) this change is synchronised within Telemappix by simply clicking an update button.

1.2 Importing Data

The select wizard provides control over the live layers being imported into PostcodeAnalyst. It is accessed through file→Import Live Data, or CTRL+SHIFT+I.

1. The first text box, entitled "SQL Select Command" takes a standard SQL command which is used to extract the data from the database containing the information.
2. Next, enter the connection string to connect to the database (see common connection strings)
3. Now click connect. This will connect to the database and run the SQL query. When complete a message will show you how many rows you have imported, and the View button will bring up a window showing the imported data.
4. In the georeference pane, select the columns which provide location data (such as postcode latitude etc).
5. Click "set georeference."

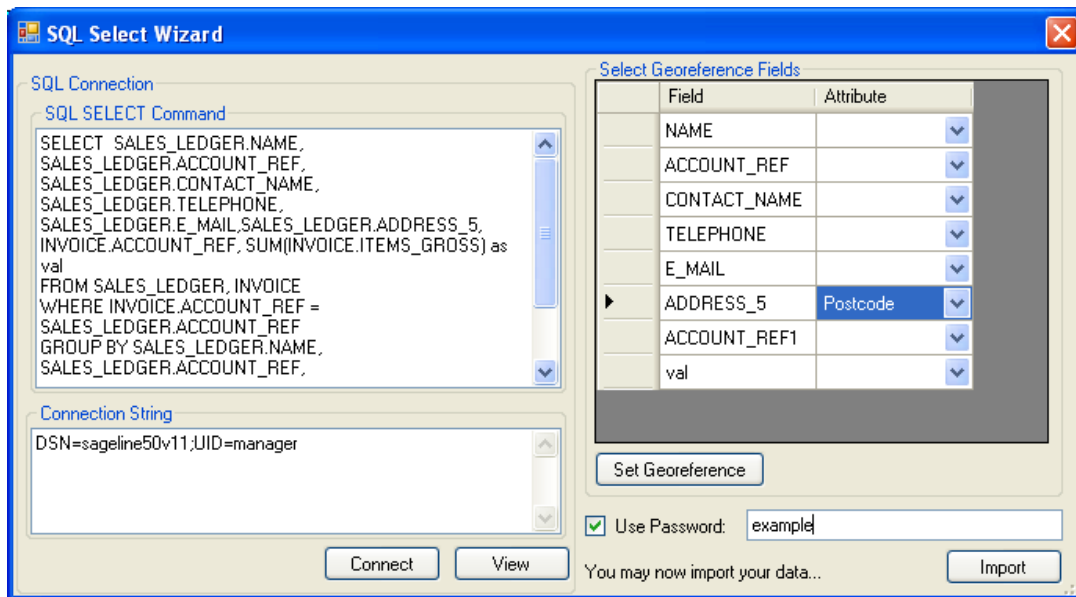


Figure 1.1: Telemappix Live Import Screen

6. If you wish to prevent tampering/viewing of the data used to connect to the database (such as passwords) tick the use password box, and enter a password, this can be whatever you please, and is used to encrypt the connection and select strings. This password will need to be entered the first time the data is refreshed for each user session.
7. Click Import. This will take you to the standard PostcodeAnalyst Import wizard, to define icon shape and colour styling.
8. Once you have created a live layer, the data is imported simply by clicking on the Refresh Data button on the main PostcodeAnalyst Window.



Figure 1.2: Telemappix Live Update Button

If you wish to allow other users to use the same connection without them having to set it up, you should copy all the '.xml' files from your current project directory, into their current project directory. This is by default:

C:\Documents and Settings\{USERNAME}\Application Data\
Gaticule\PostcodeAnalyst 2\MyProject

1.3 SQL Statements

To retrieve data from the database connection you have configured you need to use an SQL (Structured Query Language) statement which you can freely configure to give you complete control over the data set to be processed.

1.3.1 Database Structure

Ordinarily a database will contain one or more tables. Tables are identified by a name, e.g. “Clients”. Within each table there will be a number of columns representing different information, and a number of records represented by each row. For example the table below which we will call “Customers”:

<i>CustomerID</i>	<i>FirstName</i>	<i>Surname</i>	<i>Town</i>	<i>Postcode</i>
1	Robert	Smith	London	SE12 8
2	David	Johnson	Manchester	M31 4
3	Craig	Lambert	Chesterfield	S45 0
4	Helen	James	London	SE12 7

1.3.2 SELECT FROM

The SQL statements you will use the most frequently are the SELECT and FROM statements. These are very simple to use and basically say “I want to SELECT these fields FROM this table”. For example:

```
SELECT FirstName, Surname, Postcode FROM Customers
```

This will select just the Firstname, surname and postcode fields from the Customers table. If you want to select all of the fields then you can use a wildcard search like this:

```
SELECT * FROM Customers
```

If you have fields with a space in their name you should use [Field Name] to select it. It is also possible to select data from multiple tables, for example suppose we have an additional table within our database called “Orders”.

<i>CustomerID</i>	<i>OrderNo</i>	<i>Item</i>
1	1001	Book
2	1002	CD
2	1003	DVD
3	1004	PC Software
3	1005	TV

```
SELECT Customers.FirstName, Customers.Surname, Customers.Postcode,  
Orders.OrderNo, Orders.Item FROM Customers, Orders
```

This would give us a result perfect for a delivery driver:

<i>FirstName</i>	<i>Surname</i>	<i>Postcode</i>	<i>OrderNo</i>	<i>Item</i>
Robert	Smith	SE12 8	1001	Book
David	Johnson	M31 4	1002	CD
David	Johnson	M31 4	1003	DVD
Craig	Lambert	S45 0	1004	PC Software
Craig	Lambert	S45 0	1005	TV

1.3.3 WHERE

The WHERE clause adds a condition to your SELECT statements, allowing you to only select those records where something is true. For example:

```
SELECT * FROM Customers WHERE City='London'
```

This will only select the details for Robert Smith and Helen James from our table. To select all the orders above 1003:

```
SELECT Customers.FirstName, Customers.Surname, Customers.Postcode,  
Orders.OrderNo, Orders.Item FROM Customers, Orders WHERE Orders.OrderNo > 1003
```

There are a number of conditions that can be used including the following:

<i>Operation</i>	<i>Description</i>
=	Equals
<>	Not equal
>	Greater than
<	Less than
<=	Less than or equal
>=	Greater than or equal
BETWEEN	Between an inclusive range
LIKE	Search for a pattern

You may also use 'AND' 'OR' clauses to add additional conditions to your statements.

1.3.4 More Information

For more information try searching for "SQL Tutorials" or take a look at <http://www.w3schools.com/sql/default.asp>

1.4 Common Connection strings

1.4.1 SQL Server

Standard Security:

```
"Driver={SQL Server};Server=Aron1;Database=pubs;Uid=sa;Pwd=asdasd;"
```

Trusted connection:

```
"Driver={SQL Server};Server=Aron1;Database=pubs;Trusted_Connection=yes;"
```

1.4.2 SQL Server 2005

Standard security:

```
"Driver={SQL Native Client};Server=Aron1;Database=pubs;UID=sa;PWD=asdasd;"
```

Trusted connection:

```
"Driver={SQL Native Client};Server=Aron1;Database=pubs;Trusted_Connection=yes;"
```

Equivalents

```
Integrated Security=SSPI equals Trusted_Connection=yes
```

Encrypt data sent over network:

```
"Driver={SQL Native Client};Server=Aron1;Database=pubs;
```

```
Trusted_Connection=yes;Encrypt=yes"
```

Attach a database file on connect to a local SQL Server Express instance:

```
"Driver={SQL Native Client};Server=.\SQLEXPRESS;
```

```
AttachDbFilename=c:\asd\qwe\mydbfile.mdf;
```

```
Database=dbname;Trusted_Connection=Yes;"
```

1.4.3 Access

Standard Security:

```
"Driver={Microsoft Access Driver (*.mdb)}; Dbq=C:\mydatabase.mdb;Uid=Admin;Pwd=;"
```

Workgroup:

```
"Driver={Microsoft Access Driver (*.mdb)};
```

```
Dbq=C:\mydatabase.mdb;SystemDB=C:\mydatabase.mdw;"
```

Exclusive:

```
"Driver={Microsoft Access Driver (*.mdb)};
```

```
Dbq=C:\mydatabase.mdb;Exclusive=1;Uid=admin;Pwd="
```

1.4.4 Oracle

New version:

```
"Driver={Microsoft ODBC for Oracle};  
Server=OracleServer.world;Uid=Username;Pwd=asdasd;"
```

Old version:

```
"Driver={Microsoft ODBC Driver for Oracle};  
ConnectionString=OracleServer.world;Uid=myUsername;Pwd=myPassword;"
```

1.4.5 mySQL

MyODBC 2.50 Local database:

```
"Driver={mySQL};Server=localhost;Option=16834;Database=mydatabase;"
```

MyODBC 2.50 Remote database:

```
"Driver={mySQL};Server=data.domain.com;Port=3306;Option=131072;  
Stmt=;Database=my-database;Uid=username;Pwd=password;"
```

MyODBC 3.51 Local database:

```
"DRIVER={MySQL ODBC 3.51 Driver};SERVER=localhost;DATABASE=myDatabase;  
USER=myUsername;PASSWORD=myPassword;OPTION=3;"
```

MyODBC 3.51 Remote database:

```
"DRIVER={MySQL ODBC 3.51 Driver};SERVER=data.domain.com;PORT=3306;  
DATABASE=myDatabase; USER=myUsername;PASSWORD=myPassword;OPTION=3;"
```

1.4.6 Interbase

Local computer:

```
"Driver={Easysoft IB6 ODBC};Server=localhost;  
Database=localhost:C:\mydatabase.gdb;Uid=username;Pwd=password"
```

Remote Computer:

```
"Driver={Easysoft IB6 ODBC};Server=ComputerName;  
Database=ComputerName:C:\mydatabase.gdb;  
Uid=username;Pwd=password"
```

1.4.7 IBM DB2

Standard:

```
"driver={IBM DB2 ODBC DRIVER};Database=myDbName;  
hostname=myServerName;port=myPortNum;protocol=TCPIP;  
uid=myUserName; pwd=myPwd"
```

1.4.8 Sybase

Standard Sybase System 12 Enterprise Open Client:

```
"Driver={SYBASE ASE ODBC Driver};Srvr=Aron1;Uid=username;Pwd=password"
```

Standard Sybase System 12.5 Enterprise Open Client:

```
"Driver={SYBASE ASE ODBC Driver};NA=HOSTNAME,PORT_NUMBER;  
UID=username;PWD=password"
```

Standard Sybase System 11:

```
"Driver={SYBASE SYSTEM 11};Srvr=Aron1;  
Uid=username;Pwd=password;Database=mydb"
```

For more information check out the Adaptive Server Enterprise Document Sets Intersolv 3.10:

```
"Driver={INTERSOLV 3.10 32-BIT Sybase};  
Srvr=Aron1;Uid=username;Pwd=password;"
```

Sybase SQL Anywhere (former Watcom SQL ODBC driver):

```
"ODBC; Driver=Sybase SQL Anywhere 5.0; DefaultDir=c:\dbfolder\  
Dbf=c:\mydatabase.db;Uid=username;Pwd=password;Dsn="" "" "
```

Note! The two double quota ("") following the DSN parameter at the end are escaped quotas (VB syntax), you may have to change this to your language specific escape syntax (\") or maybe single quota ('). The empty DSN parameter is indeed critical as not including it will result in error 7778. Read more in the Sybase SQL Anywhere User Guide (see part 3, chapter 13)

1.4.9 Informix

Informix 3.30:

```
"Dsn="";Driver={INFORMIX 3.30 32 BIT};Host=hostname;  
Server=myserver;Service=service-name;Protocol=olsoctcp;Database=mydb;  
UID=username;PWD=myPwd"
```

Informix-CLI 2.5:

```
"Driver={Informix-CLI 2.5 (32 Bit)};Server=myserver;Database=mydb;
Uid=username;Pwd=myPwd"
```

1.4.10 Ingres

```
DSN-less "Provider=MSDASQL.1;DRIVER=Ingres;SRVR=xxxxx;DB=xxxxx;
Persist Security Info=False;uid=xxxxx;pwd=xxxxx;
SELECTLOOPS=N;Extended Properties=""SERVER=xxxxx;
DATABASE=xxxxx;SERVERTYPE=INGRES"""
```

1.4.11 Mimer SQL

Standard Security:

```
"Driver={MIMER};Database=mydb;Uid=myuser;Pwd=mypw;"
```

1.4.12 PostgreSQL

Standard:

```
"DRIVER={PostgreSQL};SERVER=ipaddress;port=5432;
DATABASE=dbname;UID=username;PWD=password;"
```

1.4.13 Paradox

5.X:

```
"Driver={Microsoft Paradox Driver (*.db )};
DriverID=538;Fil=Paradox 5.X;
DefaultDir=c:\pathToDb\;
Dbq=c:\pathToDb\;CollatingSequence=ASCII"
```

7.X:

```
"Provider=MSDASQL.1;Persist Security Info=False;
Mode=Read;Extended Properties='DSN=Paradox;
DBQ=C:\myDb;DefaultDir=C:\myDb;
DriverId=538;FIL=Paradox 7.X;MaxBufferSize=2048;
PageTimeout=600;';Initial Catalog=C:\myDb"
```

1.4.14 DSN

DSN:

```
"DSN=myDsn;Uid=username;Pwd=;"
```

File DSN:

```
"FILEDSN=c:\myData.dsn;Uid=username;Pwd=;"
```

Sage Line 50 v11

```
"DSN=sageline50v11;Uid=username;Pwd=password;"
```

1.4.15 Excel

```
"Driver={Microsoft Excel Driver (*.xls)};DriverId=790;
```

```
Dbq=C:\MyExcel.xls;DefaultDir=c:\mypath;"
```

TIP! SQL syntax: "SELECT * FROM [sheet1\$]" - i.e. worksheet name followed by a "\$" and wrapped in "[]" brackets.

1.4.16 Text

```
"Driver={Microsoft Text Driver (*.txt; *.csv)};
```

```
Dbq=c:\txtFilesFolder\;Extensions=asc, csv, tab, txt;"
```

1.4.17 DBF/Foxpro

```
"Driver={Microsoft dBASE Driver (*.dbf)};
```

```
DriverID=277;Dbq=c:\mydbpath;"
```

1.4.18 AS/400

```
"Driver={Client Access ODBC Driver (32-bit)};
```

```
System=my_system_name;Uid=myUserName;Pwd=myPwd"
```

1.4.19 Visual Foxpro

Database container (.DBC):

```
"Driver={Microsoft Visual FoxPro Driver};SourceType=DBC;
```

```
SourceDB=c:\myvfpdb.dbc;Exclusive=No;
```

```
NULL=NO;Collate=Machine;BACKGROUNDFETCH=NO;DELETED=NO"
```

Free Table directory:

”Driver={Microsoft Visual FoxPro Driver};SourceType=DBF;

SourceDB=c:\myvfpdbfolder;

Exclusive=No;Collate=Machine;NULL=NO;DELETED=NO;BACKGROUNDFETCH=NO”

”Collate=Machine” is the default setting.

1.4.20 More

More information can be found at <http://www.connectionstrings.com>.