ASTUN TECHNOLOGY

Customer Documentation 12/09

Module: DataWatcher

Country: UK

DataWatcher
Installation Guide

TABLE OF CONTENTS

1 OPEN SOURCE MIT LICENCE

2 PREREQUISITES

DataWatcher requires the following to be installed:-

1. Python-2.5.2.msi
   Python. Python module itself, Windows package installer.

2. PyGreSQL-3.8.1.win32-py2.5.exe
   PostgreSQL module for python. For connecting to PostgreSQL.

3. pyodbc-2.1.6.win32-py2.5.exe
   pyodbc module for python. For connecting to ODBC compliant databases.

4. 4Suite-XML-1.0.2.win32-py2.5.exe
   4Suite XML modules for performing XSL Transforms and parsing XML.

5. postgresql-8.1.4-1.zip
   Postgres database server.

6. pgadmin3-1.8.4.zip
   Postgres Administration module.

After these steps have been completed, ensure that **libpq.dll** is in the windows PATH environment variable (usually **C:\Program Files\PostgreSQL\8.1\bin**).

3 INSTALLATION AND CONFIGURATION

3.1 PYTHON SCRIPT INSTALLATION

The following scripts should be placed in a directory, e.g. C:\Python\DataWatcher:

Config.py
Data.py
DataWatcher.py
Output.py
SendMail.py
email.xsl

Create a directory underneath this directory e.g. C:\Python\DataWatcher\Config.

3.2 DATABASE SETUP

The following scripts should be run against the Postgres database:

at_CompareTablesWithKey.sql
at_sys_table_compatible.sql
at_sys_get_columnlist.sql
at_sys_exists_table.sql
at_sys_exists_table2.sql

3.3 CONFIGURATION FILE SETTINGS

3.3.1 SAMPLE CONFIGURATION FILE:

```
# Configuration file for DataWatcher
MyName=DataWatcher
#####################################
# How we will connect to the database
#####################################
PGHost=localhost
PGDBName=DataWatcherTest
PGUser=iShareData
PGPwd=isharedata

Query=LogosTest
Query=EROLRecords

#####################################
# How we will connect to the database
#####################################
LogosTestSQL=SELECT * FROM Logos_Test
LogosTestSourceConn=Driver^={Microsoft Access Driver (*.mdb)};Dbq^=C:\Python\DataWatcher
\Example\Example.mdb;Uid^=Admin;Pwd^=;
LogosTestDestTable=Logos_Test
LogosTestUniqueKey=ID

EROLRecordsSQL="SELECT * FROM EROL"
EROLRecordsSourceDB="Driver^={Microsoft Access Driver (*.mdb)};Dbq^=D:\Astun\Workshop\
Astun\Source Data\Workshop.mdb;Uid^=Admin;Pwd^=;"
EROLRecordsDest=EROL
EROLRecordsUniqueKey=UID

#####################################
# Email Settings
#####################################
#Email details
SMTPServer=mail.servername.com
SMTPFrom=whofrom
SMTPTo=someone@company.com
SMTPCC=someonelse@btinternet.com
SMTPSubject=DataWatcher
```

3.3.2 SETTING UP THE CONFIGURATION FILE

MyName

This is the name of the module. This does not need to be changed.

PGHost

This should be changed to match the Postgres hostname.

PGDBName

Set this value to the name of the database on which comparisons will take place. This should be the same database on which the scripts have been run (see earlier in the document).

PGUser

The username used to connect to the Postgres database.

PGPwd

The password used to connect to the Postgres database.

Query

This will be used to identify a query which will be run against the source database to obtain data for the comparison. This can be any text. Any number of queries can be defined in the configuration file. They should be listed underneath each other as in the example above.

<Query>SQL

Specify what SQL will be run against the source database for the query specified in the section above. Setting name is the Query name and SQL joined together. The value can be any valid SQL for the source database.

E.g. SELECT * FROM Ctax

<Query>SourceConn

This will be used to specify the connection string used to connect to the source database.

PLEASE NOTE THAT ANY OCCURRENCES OF '=' IN THE CONNECTION STRING SHOULD BE ESCAPED USING '^'. FOR EXAMPLE '=' BECOMES '^='

<Query>DestTable

This is the name of the table which will be created on the Postgres database to store the data for comparison.

<Query>UniqueKey

There must be a unique key specified here. Without this it will not be possible for the scripts to establish whether records have been updated.

SMTPServer

This is the name of the server which will be used to send emails from the script.

SMTPFrom

Set this to be the name of the sender.

SMTPTo

This is the email address of the person who will receive data comparison results or error messages.

SMTPCC

Email address of the recipient who will be cc'd results or error messages.

SMTPSubject

Set this to the desired subject of the email.

3.4 STYLING OUTPUT

The xsl stylesheet is used to transform the results dataset from xml to html. This can be modified if required to produce email attachments with a different appearance. The file which should be changed is **email.xsl.**

4 SCHEDULING DATAWATCHER JOBS

The DataWatcher is designed to be run at regular intervals so that frequent updates can be sent to administrators when data is changed. It is suggested that windows scheduled tasks are used to automate comparisons.

The command line syntax for running DataWatcher queries is as follows:-

DataWatcher.py <config> <Query>

5 EXAMPLE OF USE

1. Ensure that all prerequisites mentioned earlier on in this document are met.
2. Extract the **DataWatcher.zip** contents to your hard drive. Suggested directory to extract into is C:\Python, so that the structure will be C:\Python\DataWatcher.
3. Go to the **Example** folder and run **DatabaseInstall.bat**. This will automatically create the Postgres database 'DataWatcherTest' which will be used in this example.
4. Run **TestDataSetup1.py** from the **Example** folder. This can be done by either double clicking on the script or by running from the commandline (e.g. c:\python25\python.exe TestDataSetup1.py). Verify that rows are present in the **Logos_Test** table of the sample database.
5. Go to the **DataWatcher\Config** directory and copy/rename the **DataWatcher.config.example** file to DataWatcher.config. Amend the email settings at the end of the config file to suit. Also, it may be necessary to change the 'LogosTestSourceConn' entry if the Example.mdb file is in a different folder to that specified.
6. Now run the DataWatcher script from the command line:-

   cd C:\Python\DataWatcher (or the directory where DataWatcher.py resides)

   C:\python25\python.exe DataWatcher.py DataWatcher.config LogosTest

   The DataWatcher has now loaded a snapshot of the data for future comparisons.
7. Run the **TestDataSetup2.py** script from the **Sample** folder. This changes some of the data in the Logos_Test table.
8. Now run the DataWatcher script again (see step 6). Check the relevant email account for the results.

6 LIMITATIONS

Each record that you are comparing must have its own unique identifier / key.