

Manual Sensobase

DE 1.01 06/2007



Model
Release

5760x, 5763x, 5761x
1.01, Jun 2007

© 06/07 by Wiesemann & Theis GmbH

Microsoft, MS-DOS, Windows, Winsock and Visual Basic are registered trademarks of Microsoft Corporation

Subject to errors and changes:

Since we can make errors, none of our information should be used without verification. Please inform us of any mistakes or misunderstandings so that we can detect and eliminate them as quickly as possible..

Carry out work on and with W&T products only if it is described here and you have fully read and understood the manual. Unauthorized actions can result in hazards. We are not liable for the consequences of unauthorized actions. When in doubt, please contact us or your dealer first!

Introduction

Sensobase collects data from W&T Web-IO devices and writes the data to any desired database through the ODBC universal interface. The program also makes it possible to export individual sensor measurement series as an Excel spreadsheet, which can serve as the basis for a graphic representation, either complete or reduced to a daily statistic consisting of min., max. and average values.

Table of Contents

Introduction 3

1. ODBC basics 5

2. Selecting a driver 5

3. Special ODBC drivers 6

Text 6

Excel 6

Access 7

4. „True“ databases 7

MySQL 8

Data series export 8

Procedures on the database level 9

5. The user interface 13

6. Automatic record mode after Start 15

1. ODBC basics

A key concept in the ODBC world is the „data source“. This is understood as a named parameter set which points to a database. The data source used by Sensobase is by default called „W&T Sensor Database“. All data sources installed on a computer can be managed using the Windows service „Data sources (ODBC)“. Sensobase will guide you through all necessary steps for creating and configuring your data source.

The most important parameter of each data source is the ODBC driver used, which mainly determines in which format the data are saved and which database operations are available. Additional parameters determine where the data are stored as well as the options which have to do with the special database format. Since the configuration window for entering these parameters is provided by the respective driver manufacturer, these options may turn out to be very different.

2. Selecting a driver

If you have a true database such as MySQL or Oracle, you must first acquire a suitable ODBC driver; for MySQL for example this is a separate download called MyODBC. Otherwise Windows already includes several ODBC drivers, though we recommend the „Access“ driver if there is no existing database.



If you want to replace the driver, note that Sensobase does not currently offer the possibility of carrying over the already collected data from one database to the next.

3. Special ODBC drivers

Text

This driver appends its data to the end of tables. It is not possible to change the contents afterward. Put another way: The only elementary SQL commands available are INSERT and SELECT, no UPDATE or DELETE.

With this driver the table names must end with the file extension .txt. This applies not only to the files in which they are saved, but is also apparent on the level of the SQL commands. You do not get a list of all sensors for example by using „SELECT * FROM SENSORS“, but rather „SELECT * FROM Sensors.txt“.

Before using: The driver asks for a directly in which it can store its tables (a text file for each) and a file named „Schema.ini“. We recommend using a new, empty directory for this.

Excel

The main limitation here is the number of lines, which is limited to 65536 per table. Assuming you had two sensors which are polled at intervals of 10 minutes (144 measurements per day), the capacity of such an Excel database would be sufficient for approx. eight months.

Another problem arises through exclusive access of the ODBC driver to the Excel books. Multiple read-only accesses at the same time are possible, but one write access blocks any others. Put another way: As long as Sensobase is recording sensor values, no one else can view the data.

In contrast to the text driver, the UPDATE command is usable but not DELETE.

Before using: Create a new Excel book. The ODBC driver is fully able if needed to create its own xls file, but the associated

configuration window does not permit any free entries, rather only selection of already existing files.

Access

You can select the Access driver and create mdb databases even on computers on which Access is not installed. To process the data however you must have a full Access installation.

4. „True“ databases

Before using: If you do not have a client for creating a database, you can manually create the database. You should create two new users, one with full access to the database, and one with read-only authorization. With MySQL this is done for example using the following SQL commands:

```
CREATE DATABASE wt_sensor_db;
```

```
GRANT ALL ON wt_sensor_db.* TO Sensobase@'%'
```

```
IDENTIFIED BY 'topsecret';
```

```
GRANT SELECT ON wt_sensor_db.* TO guest@'%'
```

```
IDENTIFIED BY 'guest';
```

The name of the database is then immaterial. Sensobase itself does not make a database selection, rather relies on the corresponding configuration settings of the ODBC data source.

MySQL

Note: With some versions of the driver (3.51.09 to 3.51.11) all the times of day are lost (become 00:00:00). The problem is eliminated in MyODBC 3.51.12, see <http://bugs.mysql.com/bug.php?id=5681>.

Data series export

The exported data series consist of text which is tab delineated, and they can also be simply opened if necessary (e.g. on computers not having Excel) using a text editor. The standard assigned file extension .xls has the advantage however that it can be opened with Excel by simply double-clicking, so that a chart can be created in just a few keystrokes.

The export tables contain the measurement series as well as the complete master data for the respective sensor.

Procedures on the database level

The sensor database consists of two tables, one with the master data for the sensors „Sensors“ used and another continuously expanding onw with the recorded measurements („Recording“). Sensobase creates new tables if they do not already exist using the following commands:

```
CREATE TABLE Sensors (
```

```
    SensorID INT,
```

```
    Host VARCHAR( 64 ),
```

```
    Port INT,
```

```
    Slot INT,
```

```
    Unit VARCHAR( 32 ),
```

```
    Location VARCHAR( 80 ) )
```

```
CREATE TABLE Recording (
```

```
    Reading FLOAT,
```

```
    Calendar DATE,
```

```
    Clock TIME,
```

```
    SensorID INT )
```

When recording mode begins, the Sensor-ID is acquired for all sensors in the list. If these already exist in the database, only the texts for measured value and sensor location are updated:

```
SELECT SensorID FROM Sensors
```

```
WHERE Host='192.168.1.45' AND Port=80 AND Slot=0
```

```
UPDATE Sensors SET Unit='°C', Location='Flur 2. OG'
```

```
WHERE SensorID=5
```

Otherwise a not yet used ID is determined and a new master data record created:

```
SELECT MAX( SensorID ) FROM Sensors
```

```
INSERT INTO Sensors
```

```
(SensorID, Host, Port, Slot, Unit, Location)
```

```
VALUES ( 6, '192.168.1.45', 80, 0, '°C', 'Flur 2. OG' )
```

Recording a value:

```
INSERT INTO Recording (Reading, Calendar, Clock, SensorID)
```

```
VALUES ( 27.5, {d '2005-06-23'}, {t '12:21:06'}, 6 )
```

If no current sensor value can be obtained, instead a NULL value is entered in the database. In such a case you could also enter nothing at all, but then later it would no longer be possible to determine whether the gap is due to the sensor or the data collector itself.

Counting the already recorded measurements begins by obtaining all known sensors in the database and adding them to the current list:

```
SELECT Host, Port, Slot, Unit, Location FROM Sensors
```

Then the Sensor-IDs for the sensors in the current inventory list are determined and the measurements are counted in two passes, each time grouped by Sensor-ID:

```
SELECT SensorID, Host, Port, Slot FROM Sensors
```

```
SELECT SensorID, COUNT(*) FROM Recording
```

```
GROUP BY SensorID
```

```
SELECT SensorID, COUNT(*) FROM Recording
```

```
WHERE Reading IS NOT NULL
```

```
GROUP BY SensorID
```

Filtering is done according to a particular Sensor-ID and if desired by date. Explicit sorting by date and time of day is necessary, since SELECT does not otherwise necessarily output in the order in which they were recorded.

```
SELECT Location, Unit, Host, Port, Slot FROM Sensors
```

```
WHERE SensorID=6
```

```
SELECT Calendar, Clock, Reading FROM Recording
```

```
WHERE SensorID=6 AND Calendar>{d '2005-06-20'}
```

```
ORDER BY Calendar, Clock
```

The daily statistics is the most complicated evaluation, which cannot yet be implemented using a single SQL statement:

```
SELECT Location, Unit FROM Sensors WHERE SensorID=6
```

```
SELECT Calendar, MAX( Reading ), AVG( Reading ),
```

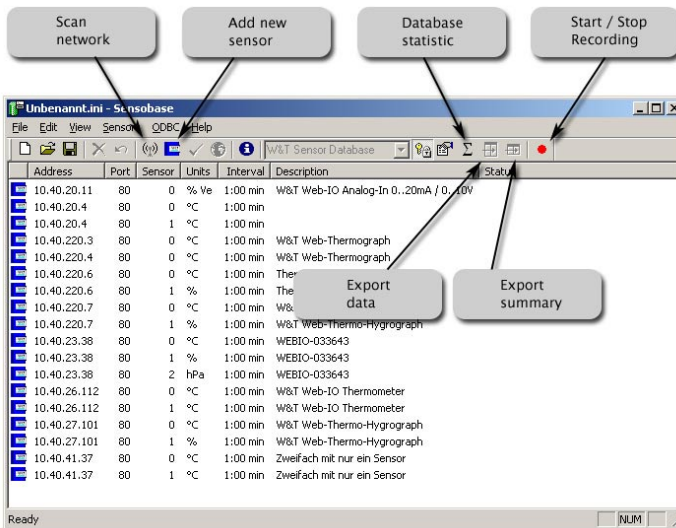
```
MIN( Reading ), COUNT( * ) FROM Recording
```

```
WHERE SensorID=6 AND Reading IS NOT NULL
```

```
GROUP BY Calendar
```

```
ORDER BY Calendar
```

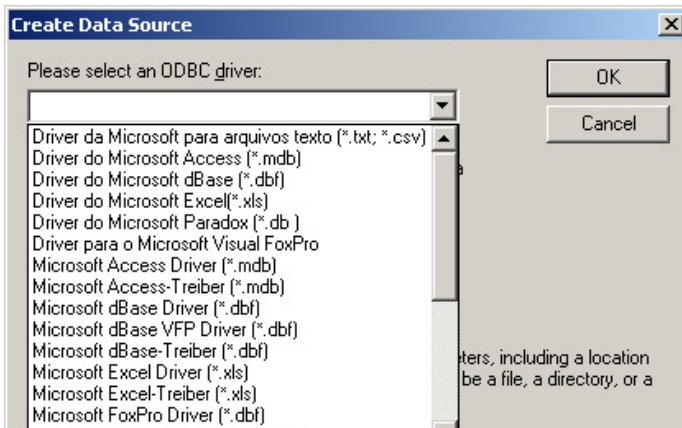
5. The user interface



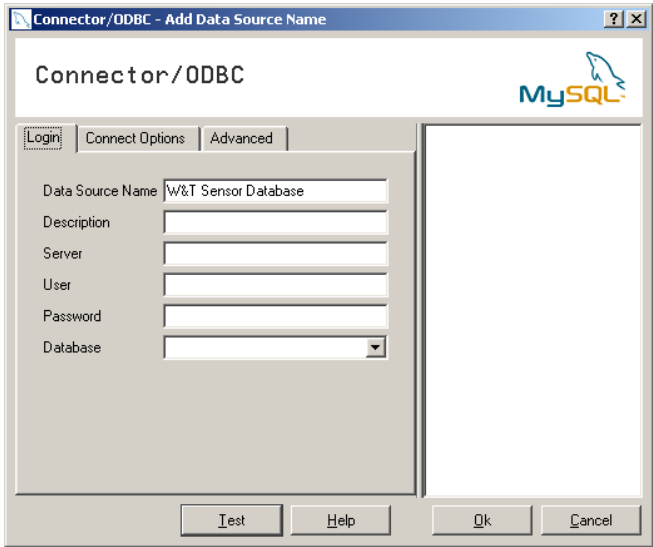
In order to configure Sensorbase for data import, first select

„ODBC -> Select other driver“.

Here you select the desired ODBC driver.



Now the configuration for the selected driver appears. This may vary from manufacturer to manufacturer. The MySQL configuration for example looks as follows:



After you have entered the User and Login information, you may start data recording.

The data can be exported at any time directly as an Excel file for each sensor.

Sensobase also offers the option of exporting a summary (min. temperature, max. temperature, average temperature per day).

6. Automatic record mode after Start

In order to place Sensobase in record mode automatically after startup, you must first have an ini file stored. Now Sensobase can be automatically placed in record mode using the start parameter `/r`.

sensobase.exe /r

or

sensobase_en.exe /r