



**Flex.Base 2.1.x MySQL  
Getting Started**

# Notices

- **COPYRIGHT**

COPYRIGHT © 2006 by RealFlex Technologies. All right reserved

Note: No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronically, mechanical, magnetic, optical, manual, or otherwise, without prior written permission of RealFlex Technologies.

- **DISCLAIMER**

RealFlex Technologies makes no representation or warranties with respect to the contents hereof and specifically disclaims any implied warranties or merchantability or fitness for any particular purpose. Further, RealFlex Technologies reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation to notify any person of such revision or changes.

- **TRADEMARKS**

Microsoft Windows is a registered trademark of Microsoft Corporation

RealFlex Technologies,  
2218 Northpark Drive, Suite 202,  
Kingwood, Texas, 77339, USA  
Tel: +1 281 348 2341, Fax: +1 281 348 2340  
Email: [sales@realflex.com](mailto:sales@realflex.com)  
<http://www.realflex.com/>

RealFlex Technologies Ltd,  
Limerick Business Complex,  
Raheen Business Park, Limerick, Ireland.  
Tel: +353 61 308884, Fax +353 61 308883,  
Email: [sales@realflex.com](mailto:sales@realflex.com)  
<http://www.realflex.com/>



# Getting Started

<b>CONVENTIONS, SYMBOLS AND TERMS USED .....</b>	<b>5</b>
<b>INSTALLATION GUIDE .....</b>	<b>6</b>
<b>1. STARTING THE INSTALLATION .....</b>	<b>7</b>
1.1.    INSTALL FLEX.WIN HARDWARE KEY .....	7
1.2.    INSTALL FLEX.SERV HARDWARE KEY .....	7
1.3.    INSTALLING FLEX.BASE SOFTWARE FROM CD ON THE MICROSOFT WINDOWS PC .....	8
1.4.    INSTALLING FLEX.SERV SOFTWARE ON QNX/REALFLEX 4 PC .....	16
1.5    CONFIGURE QNX/REALFLEX 4 PC .....	17
1.5.1.    Configure Auto Update to Run Periodically .....	17
Manual Operation.....	17
Automatic Operation.....	17
1.6    STARTING FLEX.BASE.....	18
1.7    EXAMPLE: ACCESS TO DATABASE USING EXCEL .....	24
<b>2. REGISTRATION OF HARDWARE KEYS .....</b>	<b>28</b>
<b>APPENDIX A. UTILITIES.....</b>	<b>28</b>
Reset Project Update Time.....	28
Data Lifetime.....	28
Converting a 1.x project to a 2.0 project .....	28
Retrieving historical events and values from database .....	28
Tuning security settings.....	28
<b>APPENDIX B. DATABASE TABLES.....</b>	<b>28</b>

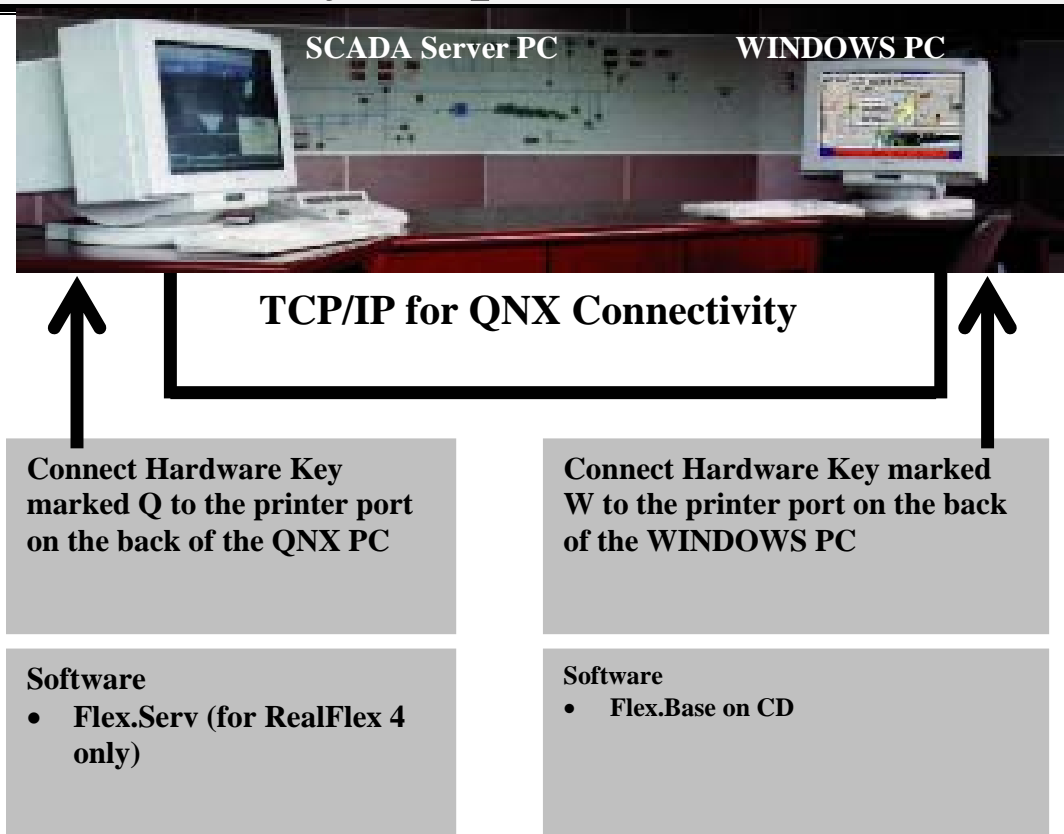


## Flex.Base Overview.

Flex.Base is a high performance database solution for real-time/historical data and event storing. It combines the flexibility of a relational database with an SQL engine based on the MySQL server, enabling inter operational capacity in the Microsoft Windows Environment. It can be used with third party or locally developed report generators or external databases for historical data and events.

Flex.Base communicates with one or more Flex.Serv servers over a TCP/IP network, on a LAN or WAN (28800 bps or more). The functioning of Flex.Base is session-oriented. At the beginning of a session Flex.Base starts the update process, it detects any changes to the project structure on the RealFlex side and applies changes to the SQL database. Then Flex.Base starts the process for historical events, historical values and real-time value processing.

## Connectivity Map



## Conventions, Symbols and Terms Used

Throughout this manual the following conventions are used.

- **Characters** are used to indicate text on screen.
- **CAPITAL** letters are used for the names of options found on the menu strips and to highlight information such as file names.
- *Italic* characters are used to indicate something that has to be typed in or selected.

In the margins there are the following symbols used to highlight important information.

Symbol	Meaning
--------	---------



A useful tip



A warning or cautionary note



An example

Term	Meaning
------	---------

XXXX is used in this manual to refer to a generic project name. Please replace XXXX with YOUR project name.

Test The example used to generate this manual was a project named TEST



## Installation Guide

This section describes the installation process.

### System Requirements

The following table outlines the minimum and recommended system requirements for installing and running the software.

<b>Equipment</b>	<b>Minimum</b>	<b>Recommended</b>
Operating system Computer	Windows 2000 or above IBM compatible Pentium, a mouse or other pointing device and a VGA graphics display	Windows XP Professional Pentium 3 (Dual Pentium or MPS) , 1GHZ, Mouse or other pointing device and a SVGA graphics display
Memory (RAM)	256MB	512MB
Hard Disk space (approx.)	62MB for minimum installation plus: <ul style="list-style-type: none"><li>• 10MB for libraries plus:</li><li>• 2GB for project</li></ul>	62MB for minimum installation plus: <ul style="list-style-type: none"><li>• 10MB for libraries plus:</li><li>• 2GB for project</li></ul>
CD ROM	Required	Required



Flex.Serv documentation is contained within the Flex.Base CD in the documentation folder.



## 1. Starting the Installation

In all there are 7 steps to a successful installation, these include:

- Install Flex.Win Hardware Key (Note: USBkey, please install after Flex.Base)
- Install Flex.Serv Hardware Key
- Installing Flex.Base Software on the Microsoft Windows PC
- Installing Flex.Serv Software on QNX/RealFlex 4 PC (**Not required for RealFlex 6 or RealWin**)
- Configuring QNX/RealFlex PC
- Starting Flex.Base
- Access to database information using Microsoft Excel

The software is protected using a hardware device, which plugs into the parallel port on the PC.

### 1.1. Install Flex.Win Hardware Key

Attach the Hardware key that is labelled with the letter 'W' to the printer port at the back of your Windows PC



If you are using a USB hardware key, insure you do NOT install the key before you install the Flex.View software, as the Flex.View installation installs a driver for the USB key, which is required when the USB key is plugged in.

### 1.2. Install Flex.Serv Hardware Key

Attach the Hardware key that is labelled with the letter 'Q' to the printer port at the back of your QNX PC. In certain cases you may have multiple hardware keys already on your PC. Please add the new key to the existing one.



### 1.3. Installing Flex.Base Software from CD on the Microsoft Windows PC

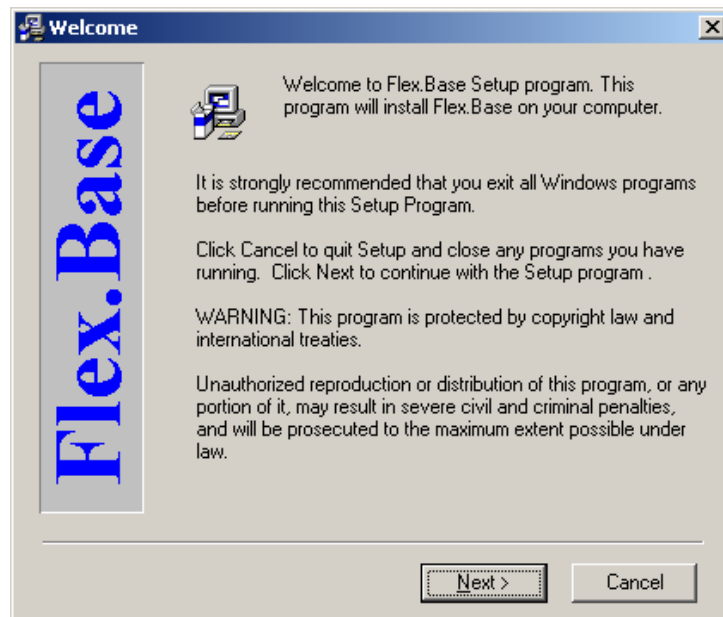
Insert the Flex.Base CD into the PC running Microsoft Windows.

If the CD does not automatically start the installation click the Windows Start button and select Run. The Run dialog box appears.

In the Open field type:

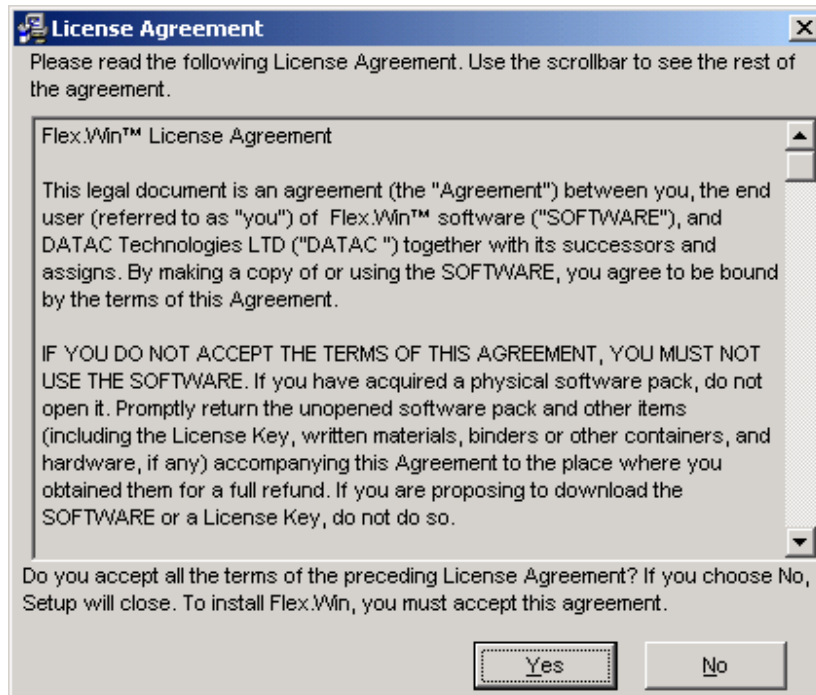
**D:\FBSETUP.EXE** – or the applicable drive letter that is referring to the CD drive

Click OK. The Set-up program starts.



Click "Next"

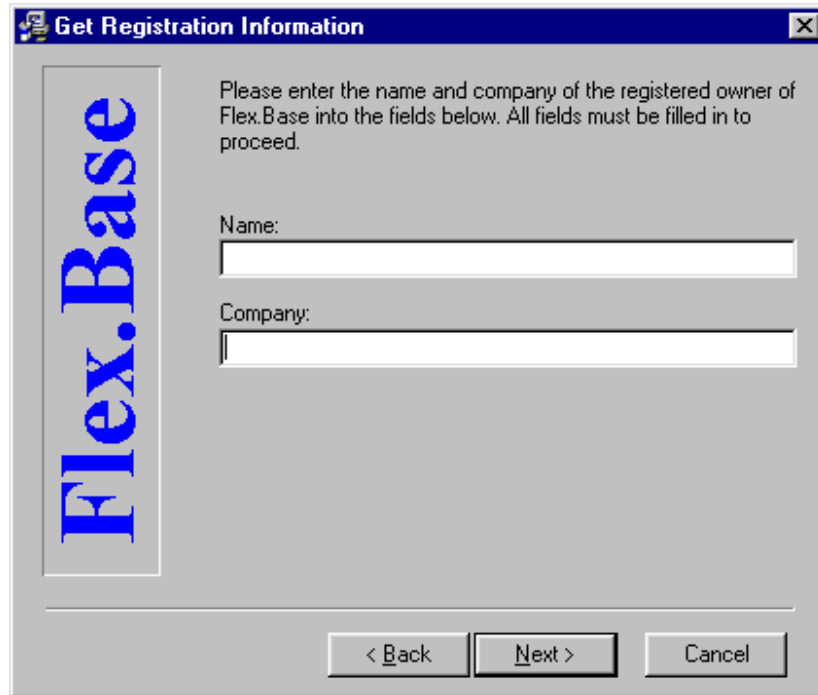




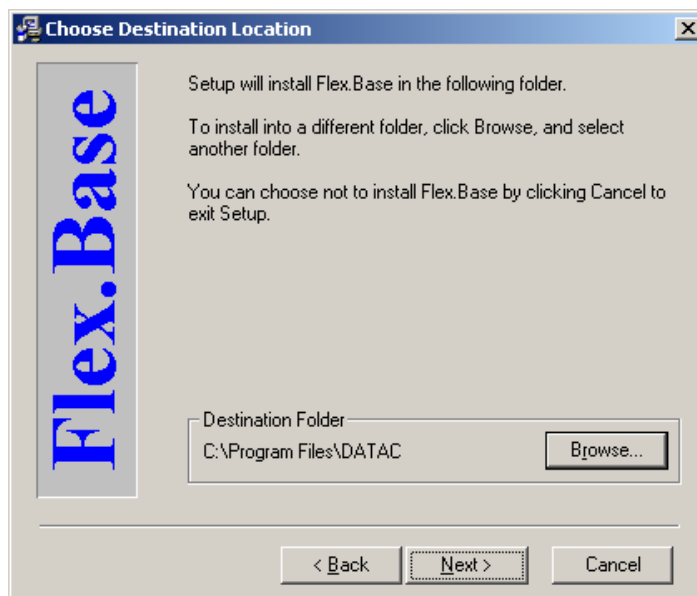
Click "Yes"



Read the license agreement and if you accept click Yes



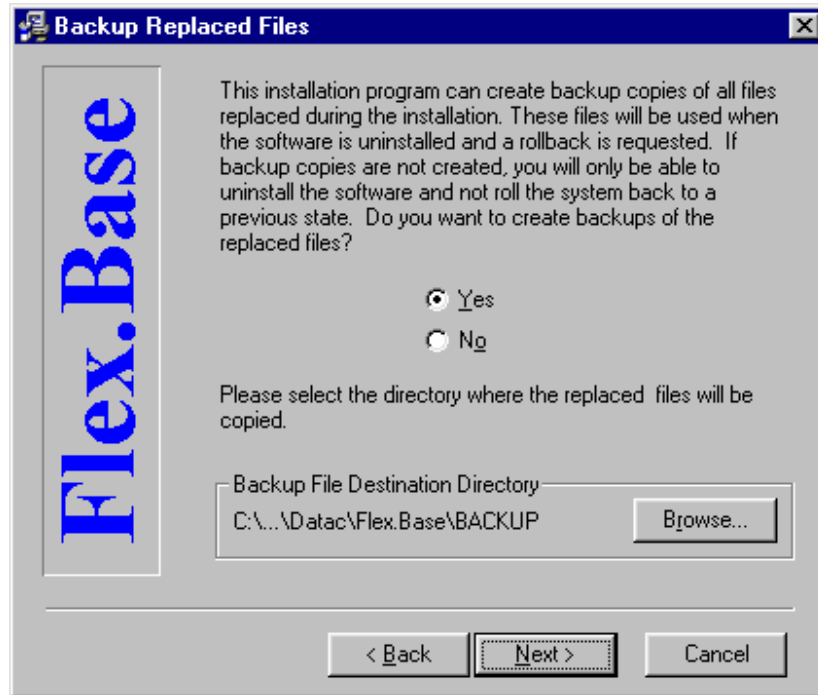
Enter Name and Company and click Next  
NOTE: Both Name and Company must be entered.



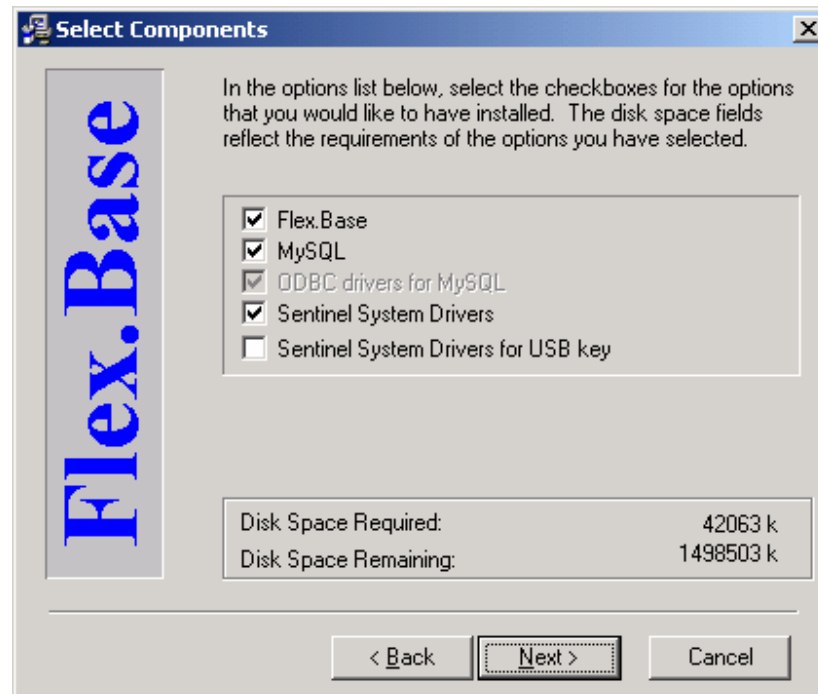
Ensure you have approximately 2.5 Gb free disk space if you are storing historical data as this stores approximately 10 million records.



Click Next



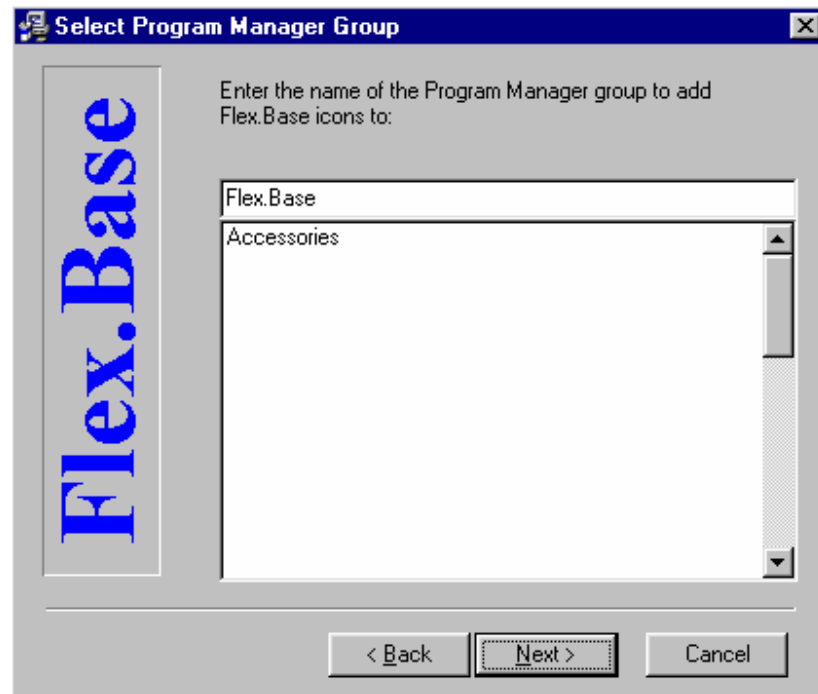
Click Next



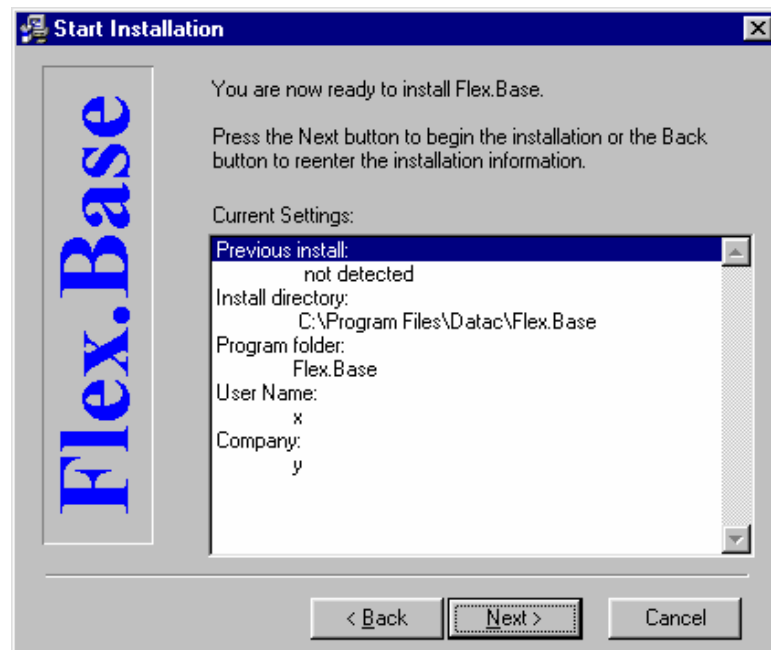
## FlexBase - Getting Started

'ODBC drivers for MySQL' is not required for Flex.Base itself, but it is necessary for connectivity with other application which can access the database using ODBC

Click Next



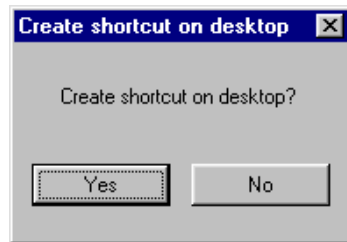
Click Next



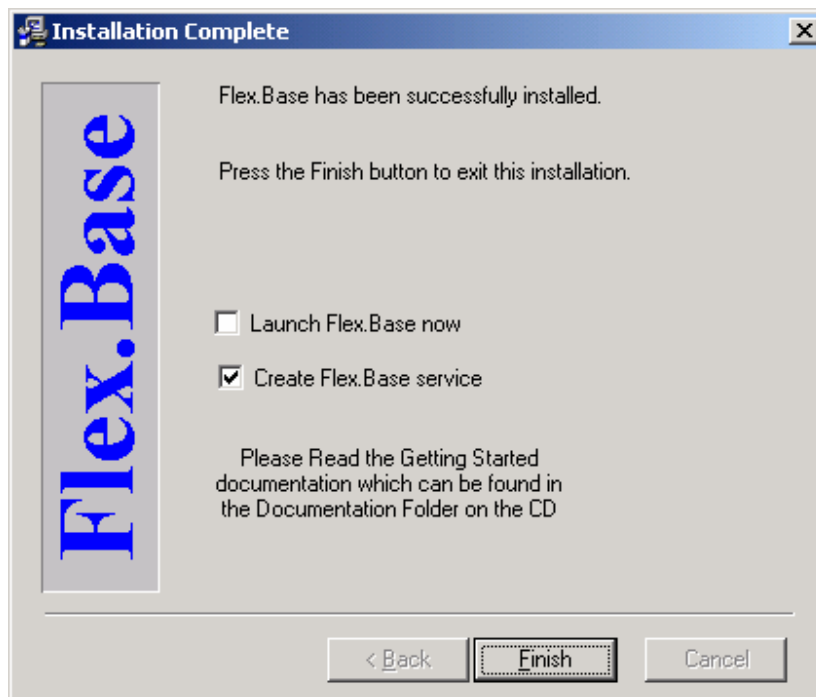
Click Next and wait until all files are installed.



## FlexBase - Getting Started



Click Yes



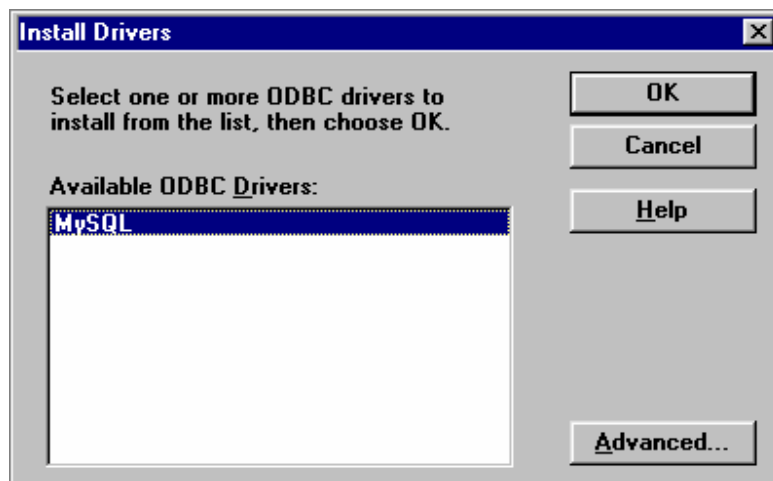
Click Finish



Click Yes

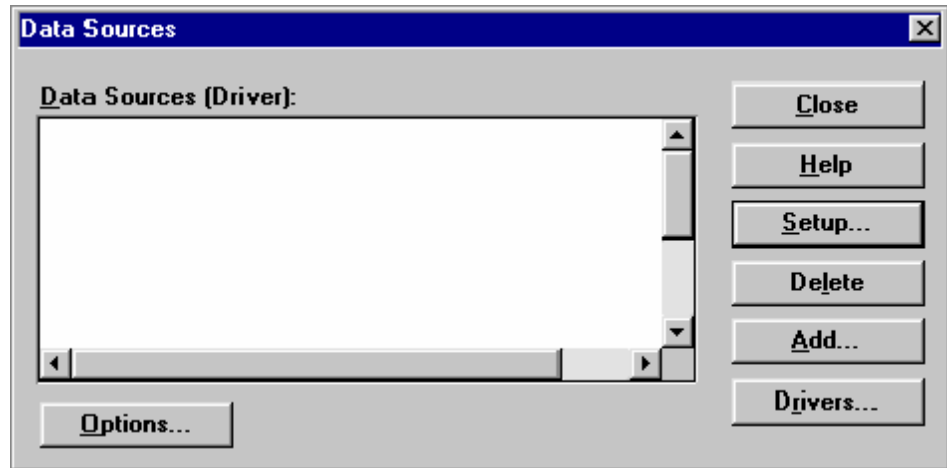


Click Continue

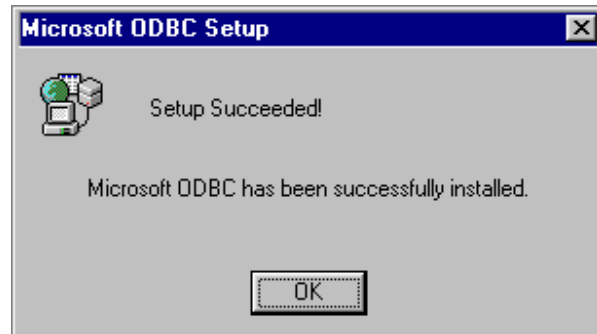


Ensure to click on **MySQL** to select it and click on OK



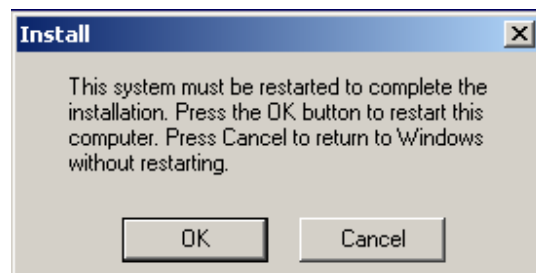


Click Close



Click OK

If you are using Microsoft Windows 2000 or NT you will need to reboot. The following windows will appear. If you select 'Cancel' Flex.Base will installed correctly after the next Reboot

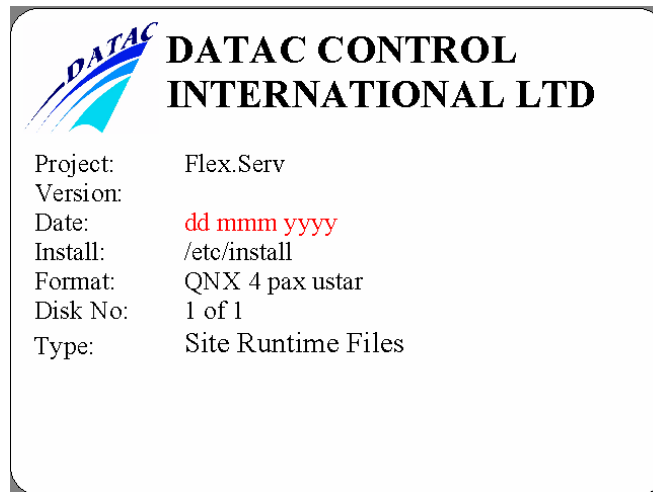


## 1.4. Installing Flex.Serv Software on QNX/RealFlex 4 PC

If you have a RealFlex 4 system, you need to have FlexServ installed on the SCADA server PC i.e QNX/ RealFlex 4 PC.

If you are using RealFlex 6 or RealWin, you do not need to install any extra software on the SCADA server PC.

Insert Flex.Serv disk into QNX/RealFlex PC  
The disk will have a label similar to this



Login as a superuser i.e. type

```
login root  
password :  
#
```

Install the software

```
# cd /  
# /etc/install
```

This is a failover configuration? y/n

Please answer as appropriate.

**Please ensure to remove the Floppy Disk**





## 1.5 Configure QNX/RealFlex 4 PC

### 1.5.1. Configure Auto Update to Run Periodically

AutoUpdate detects changes made to the Realflex system, when run, and saves these changes in a format that allows Flex.View clients to update these changes to the Windows PC, when Flex.View connects or when a "Check for Updates" is done by the Flex.Win user.

AutoUpdate can be run manually at any time, after changes have been made to the Realflex System or it can be configured to run periodically.

**AutoUpdate has to be run with the -f parameter when used in a failover configuration to insure files are updated on both nodes..**

#### Manual Operation

Login as a superuser

Open a shell and type the following command

**AutoUpdate -A <CR>**

or

**AutoUpdate -A -f <CR>** on a failover configuration to insure files are updated on both nodes.

#### Automatic Operation

FlexServ installation automatically configures the rptcron file as that AutoUpdate runs every hour. If you wish to change this :

Edit /realflex/data/rptcron and edit the line to run AutoUpdate every hour on the hour

```
0 * * * * /realflex/bin/AutoUpdate -A > /dev/null 2> /dev/null
```

or to run AutoUpdate once every day at 5 minutes past 7 each morning on a failover configuration

```
5 7 * * * /realflex/bin/AutoUpdate -A -f > /dev/null 2> /dev/null
```

The period at which AutoUpdate is run, has to be setup to suit the individual customer.

When Realflex is restarted this will take effect or if you wish not to restart Realflex, you can use the following command

```
# crontab -u realflex -L /realflex/data/rptcron
```



## 1.6 Starting Flex.Base

The Flex.Base module is the main module that retrieves historical events, historical values, real-time values, flags, updates information and stores it to the database project. It can process multiples up to 10 configurations simultaneously (if planning to use multiprocessor hardware configurations). When Flex.Base is started for the first time, it looks as follows without any project configurations.



Use program menu 'Projects->Add new project configuration'.

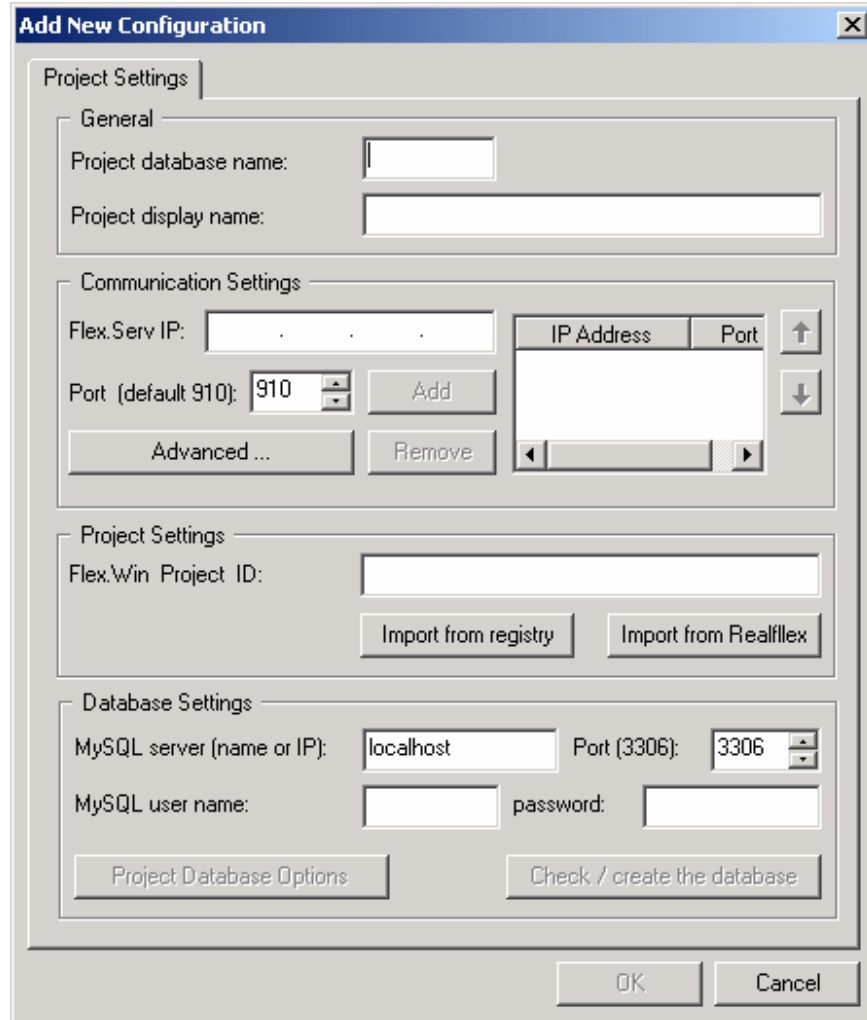


Fig 9

Enter the «Project database name» (Demo in our example) and the «Project display Name» (Demo).

Enter the "Flex.Serv I.P" address and click on "add".



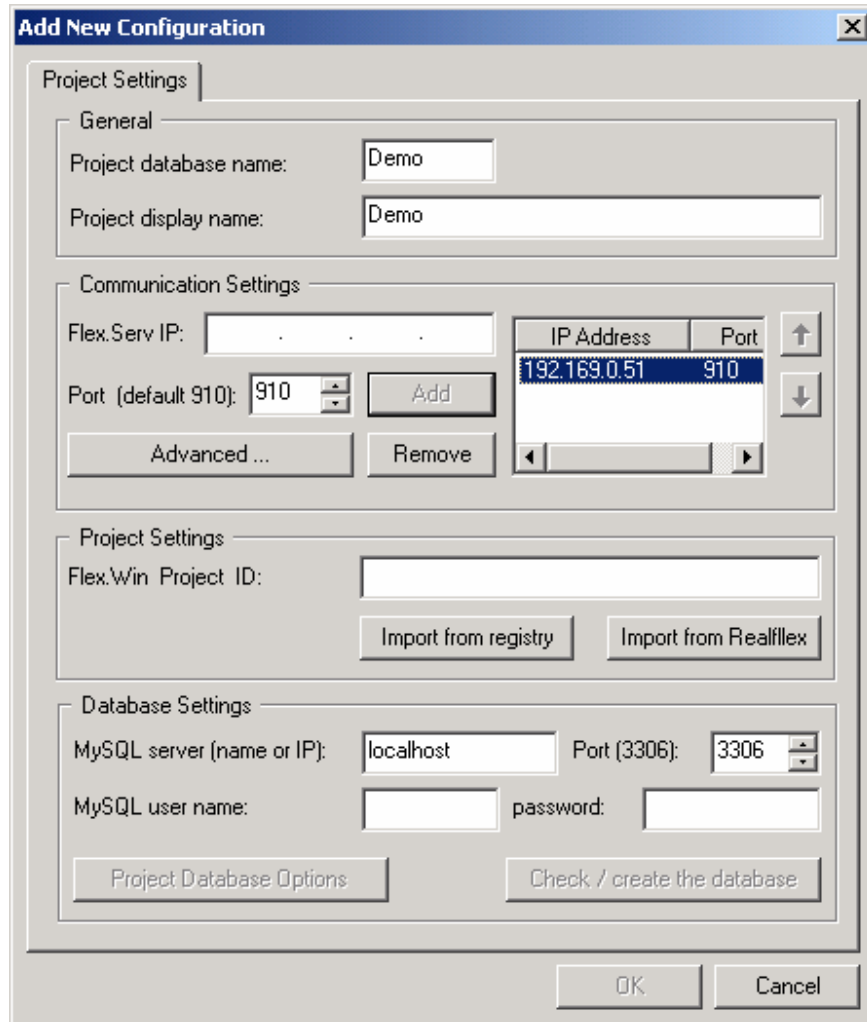
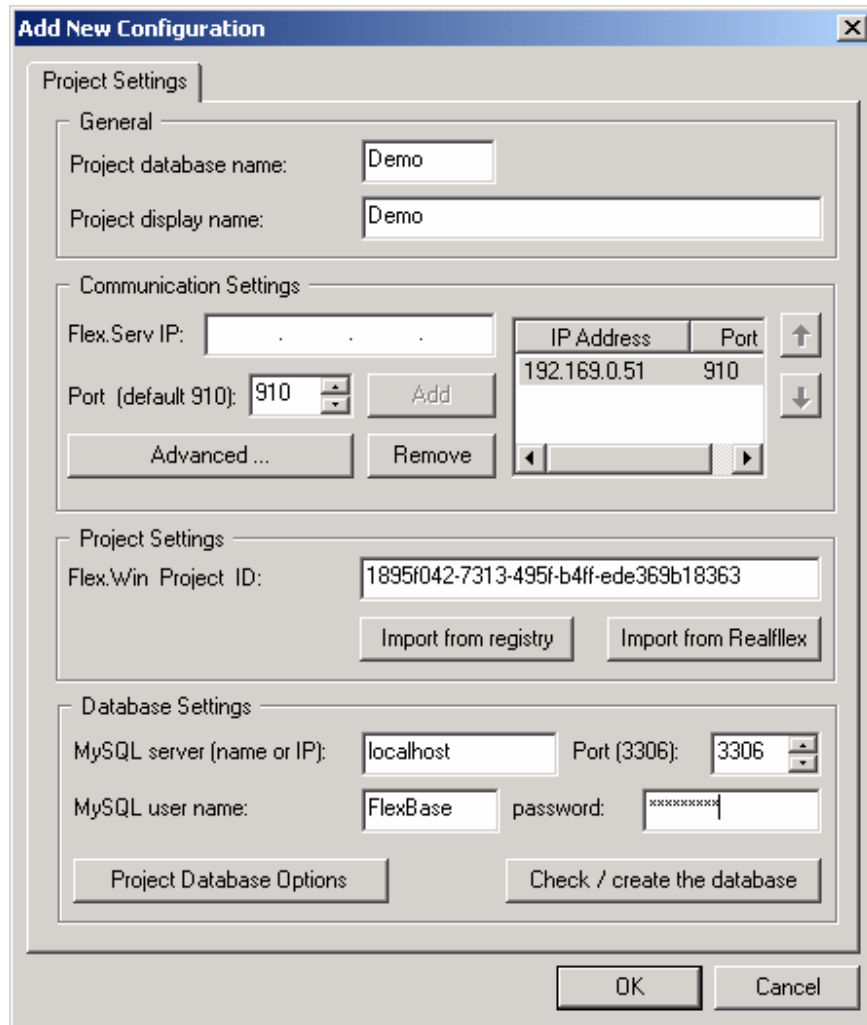
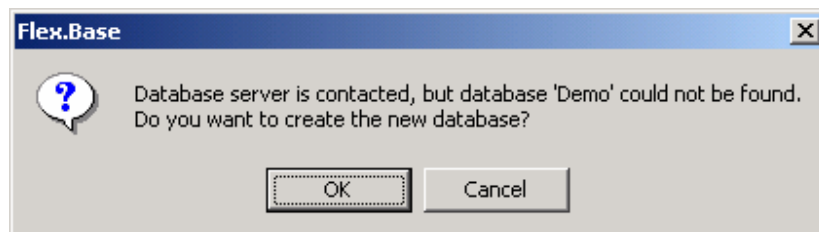


Fig. 10.  
Click on “Import from registry”  
Flex.Win Project ID will be imported



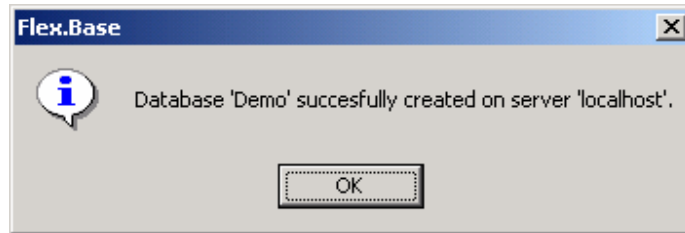


Fill in the “MySQL user name” and “password”. (FlexBase & DATA2000)  
Click on the “Check/Create the database” button.

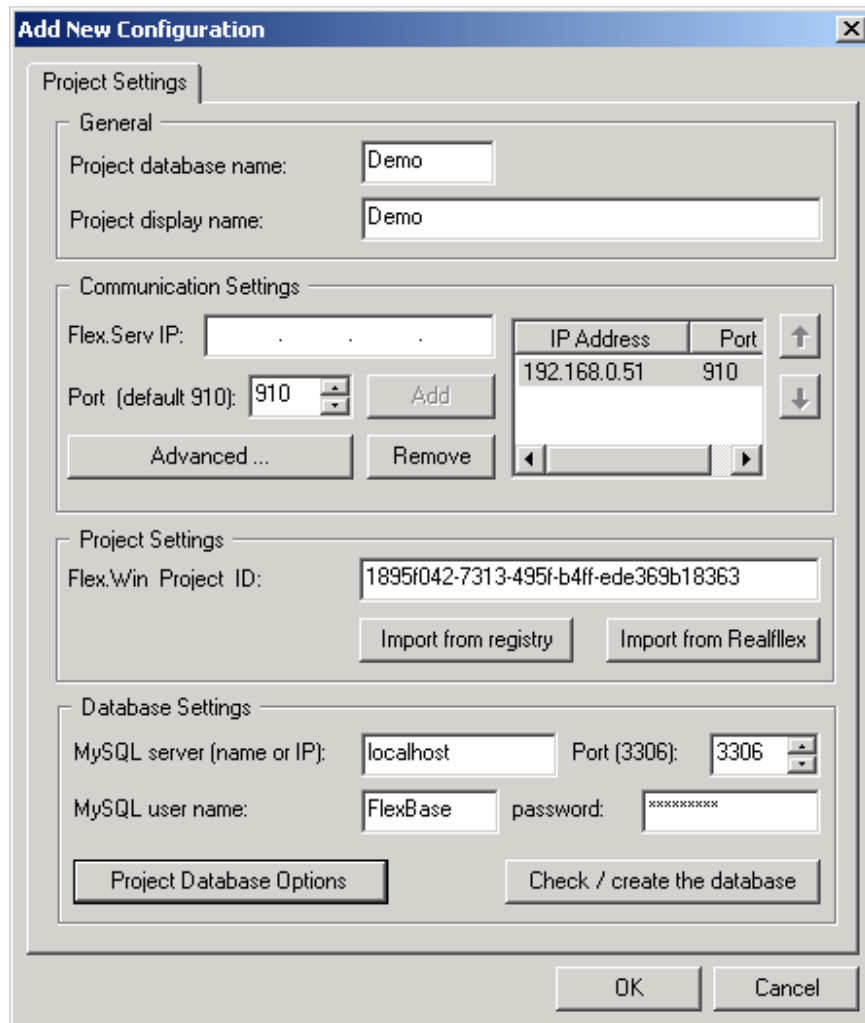


Click “OK”.



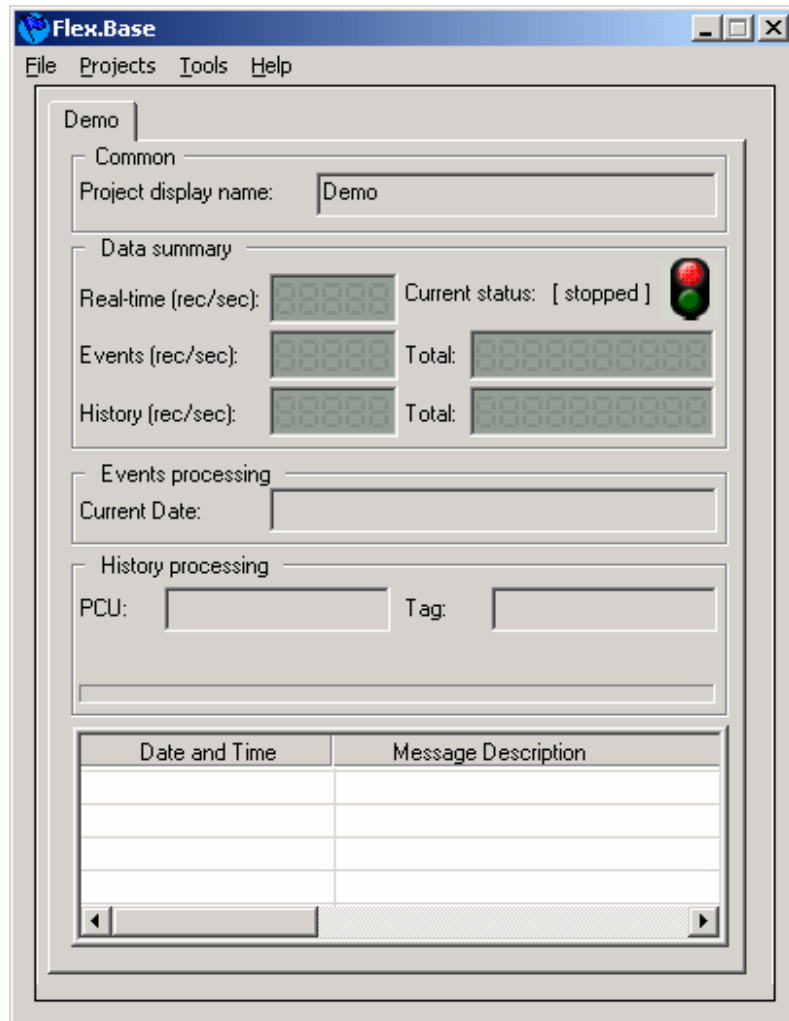


Click "OK".

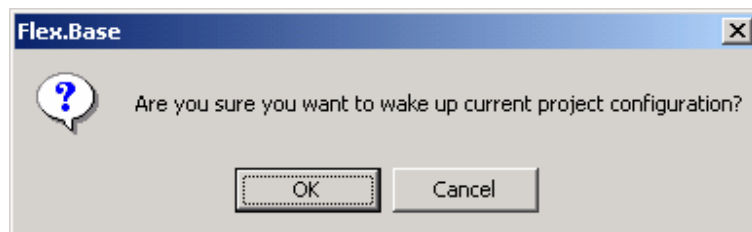


Click "OK"





Click on the “Project” menu and select “Start collecting data”



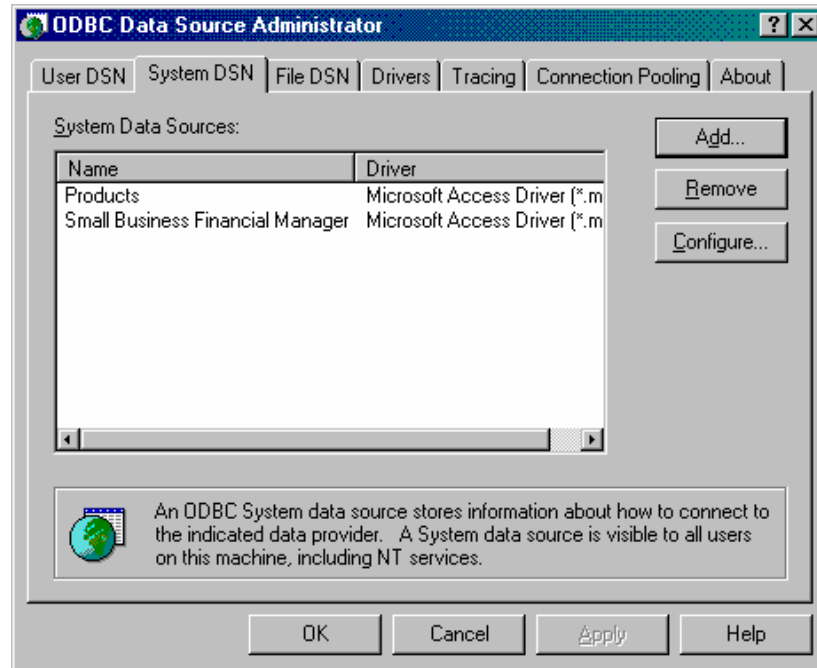
Click “OK”

Flex.Base will begin to retrieve any historical data.

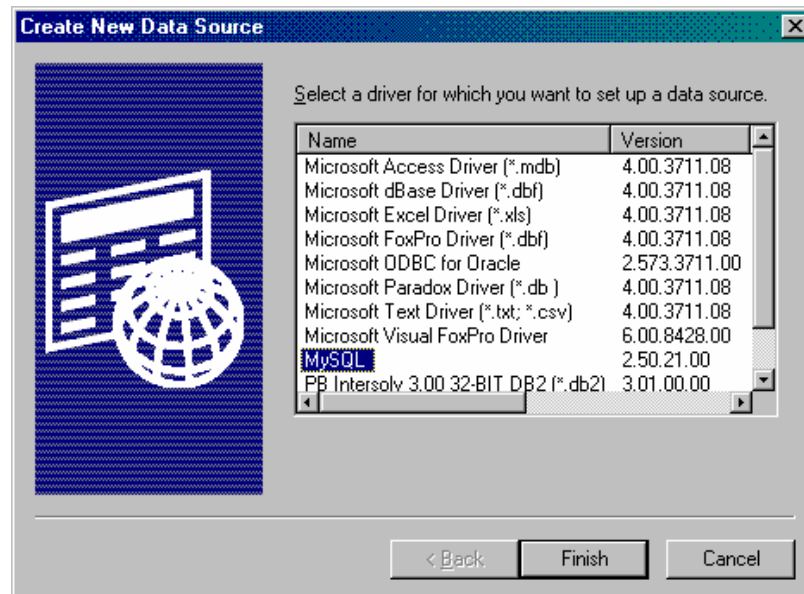
## 1.7 Example: Access to database using Excel

In order to access the MySQL database using ODBC, it is necessary to configure the database as an ODBC data source.

- Click on Start
- Select Settings
- Select Control Panel
- Select Administrative Sources
- Click on ODBC Data Sources (32 bit)



- Select System DSN tab
- Click Add





Select MySQL  
Click Finish

**TDX mysql Driver default configuration**

This is in public domain and comes with NO WARRANTY of any kind  
Enter a database and options for connect

Windows DSN name: FlexBase Test

MySQL host (name or IP): localhost

MySQL database name: Test

User: FlexBase

Password: \*\*\*\*\*

Port (if not 3306):

SQL command on connect:

Options that affects the behaviour of MyODBC

<input type="checkbox"/> Don't optimize column width	<input type="checkbox"/> Pad CHAR to full length
<input type="checkbox"/> Return matching rows	<input type="checkbox"/> Return table names in SQLDescribeCol
<input type="checkbox"/> Trace MyODBC	<input type="checkbox"/> Use compressed protocol
<input type="checkbox"/> Allow BIG results	<input type="checkbox"/> Ignore space after function names
<input type="checkbox"/> Don't prompt on connect	<input type="checkbox"/> Force use of named pipes
<input type="checkbox"/> Simulate ODBC 1.0	<input type="checkbox"/> Change BIGINT columns to INT
<input type="checkbox"/> Ignore # in #.table	<input type="checkbox"/> No catalog (exp)
<input type="checkbox"/> Use manager cursors (exp)	<input type="checkbox"/> Read options from C:\my.cnf
<input type="checkbox"/> Don't use setlocale	<input type="checkbox"/> Safety (Check this if you have problems)

OK Cancel

Fill in the first 5 fields of the configuration

Windows DSN name: Any name you choose

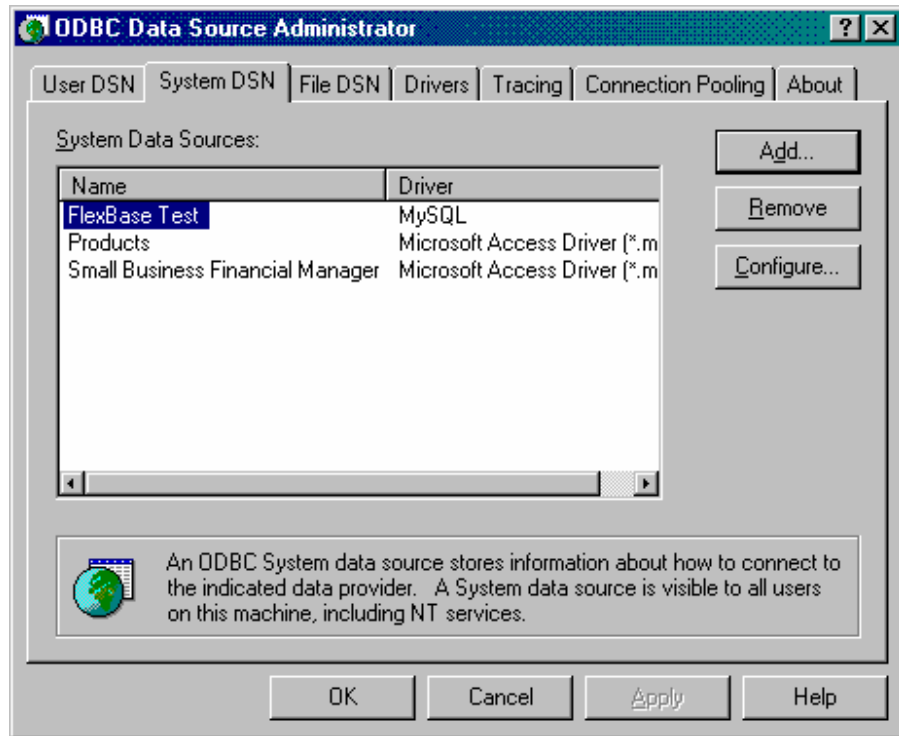
MySQL host: If this configuration is on the same PC as the MYSQL database is stored, then use local host. If the database is stored on another PC, then use the name or IP address of the PC containing the database.

MySQL database name: Name used in Flex.Base Init for the MySQL database e.g. Test

User: FlexBase

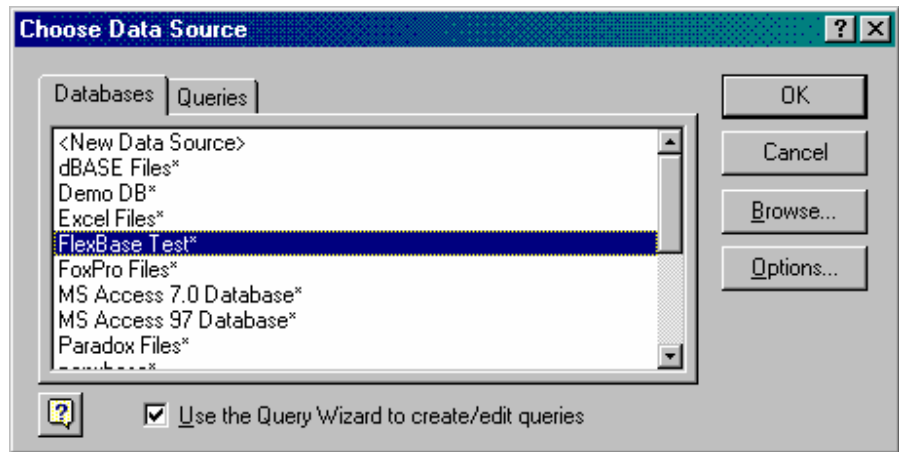
Password: DATAC2000





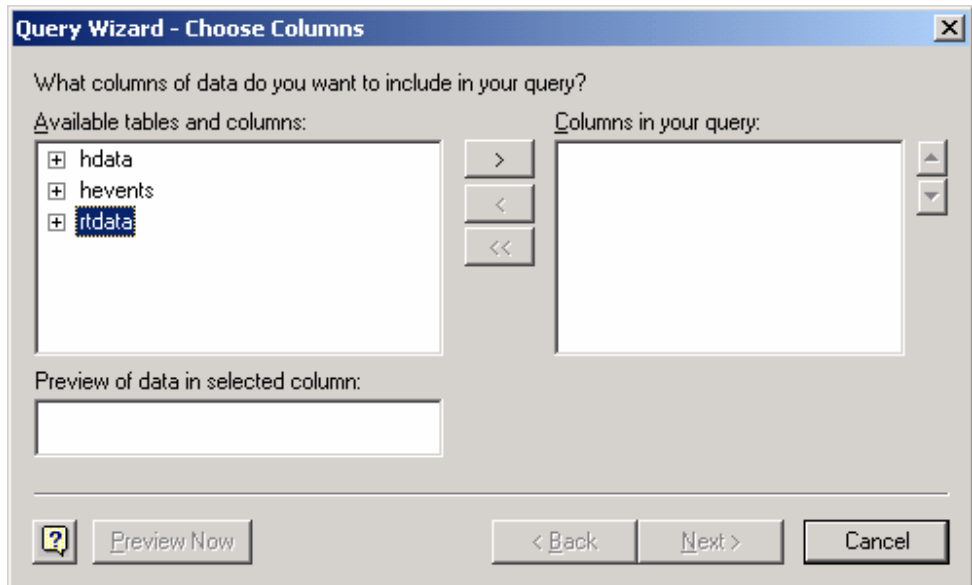
Click OK

- Open Excel
- Select Data
- Select Get External Data
- Select Create New Query

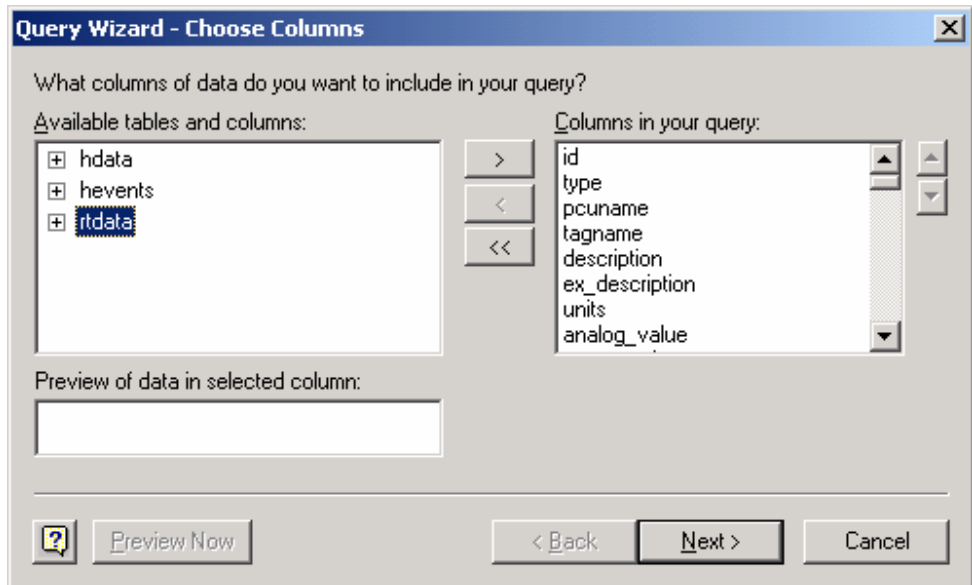


Select your FlexBase Data Source e.g FlexBase Test  
Click OK

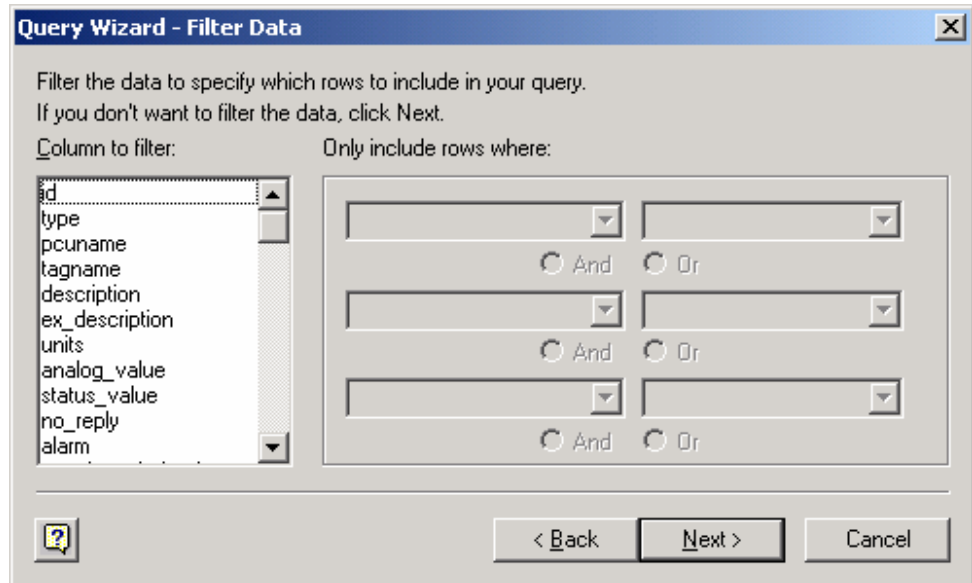




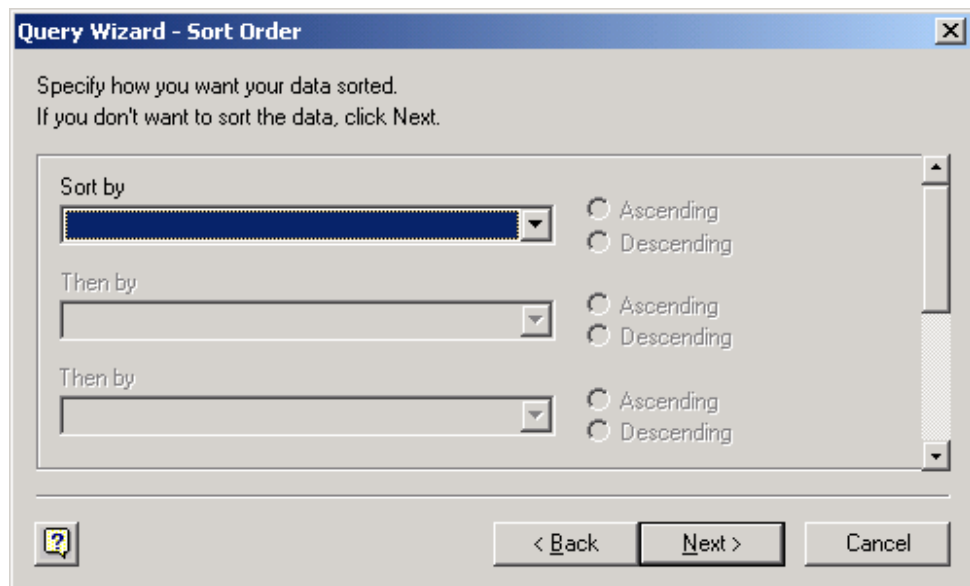
Click on the ">" to add all the fields of the "rtdata" table to the columns in your query.



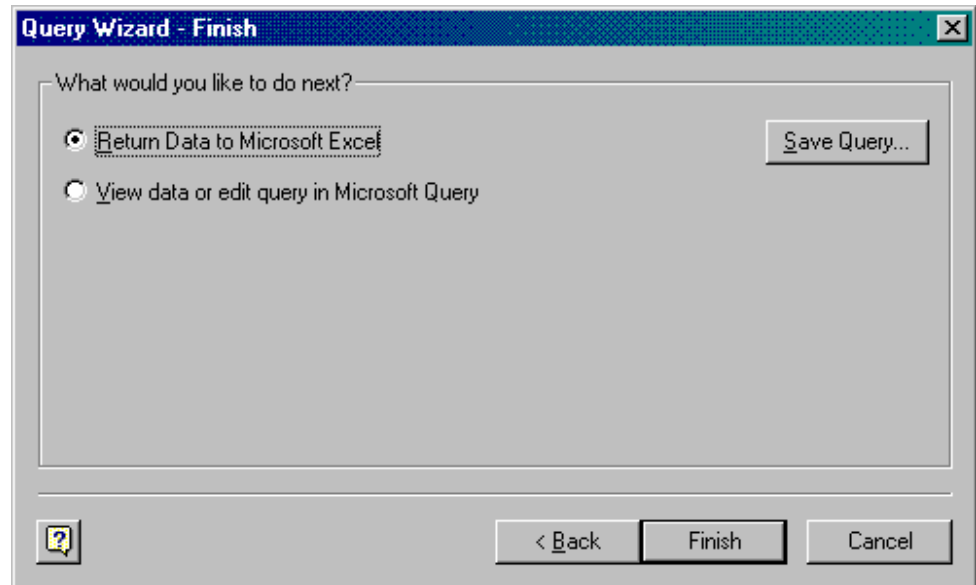
Click "Next".



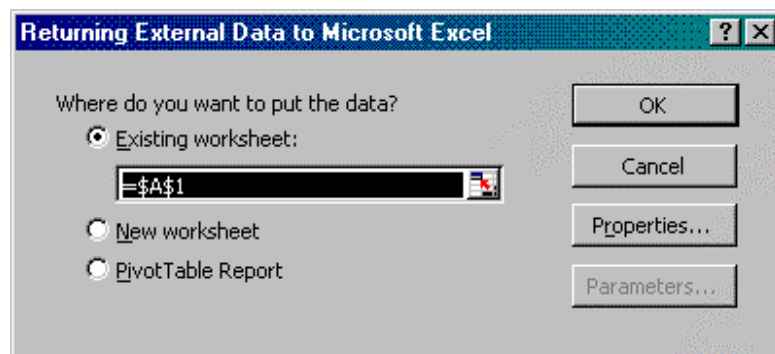
Click "Next"



Click "Next".



Click Finish



Click OK

“Rtdata” information is returned to Excel column A row 1

## FlexBase - Getting Started

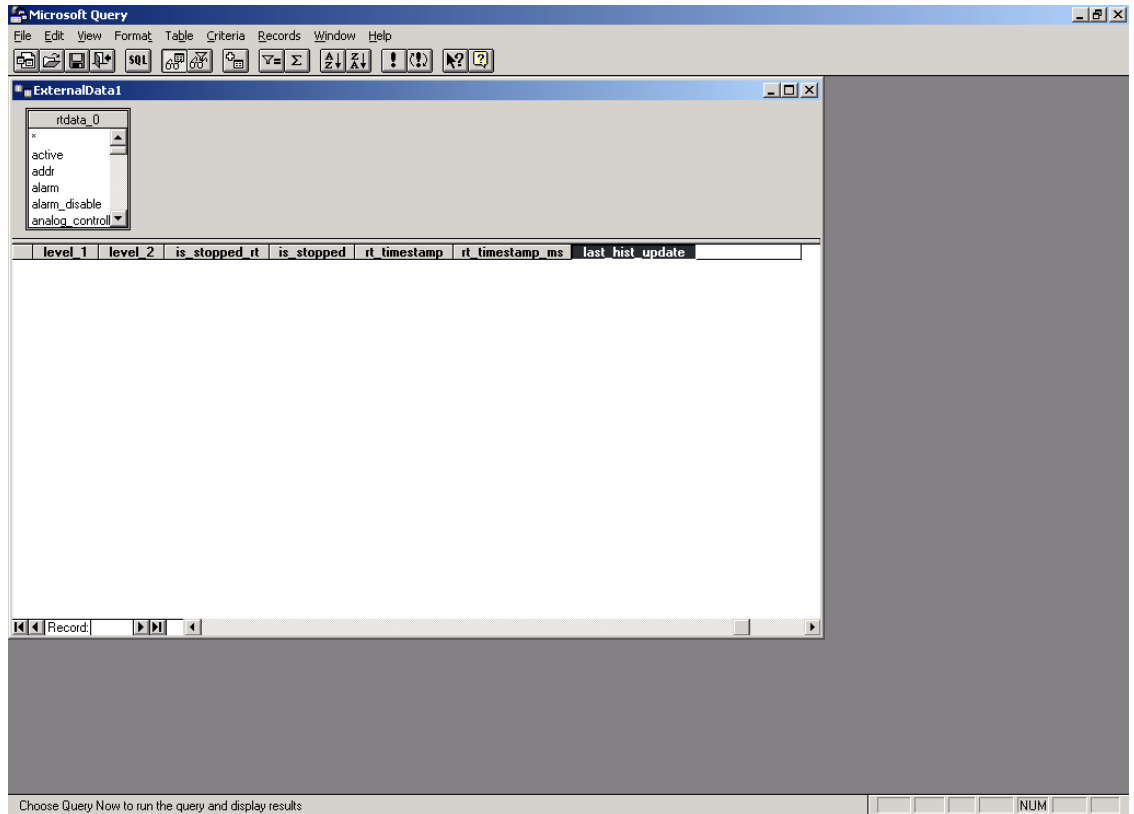
The screenshot shows a Microsoft Excel spreadsheet with the following data:

id	type	pcuname	tagname	description	ex_description	units	analog_value	status_value	no_reply	alarm	unacknowledge
2	3145781	5 FLEXWIN								1	0
3	524301	5 RTU_1								0	0
4	1048597	5 RTU_2								0	0
5	1572893	5 RTU_3								0	0
6	2097189	5 RTU_4								1	0
7	2621485	5 RTU_5								1	0
8	5	5 SYSTEM								1	0
9	524296	0 RTU_1	AI_00_00	TANK 1 LEVEL		M	3.4597			0	1
10	524304	0 RTU_1	AI_00_01	TANK 2 LEVEL		M	3.79003			0	1
11	524312	0 RTU_1	AI_00_02	TANK 1 CAPACITY		%	88.4527			0	0
12	524320	0 RTU_1	AI_00_03	TANK 2 CAPACITY		%	23.0445			0	0
13	524328	0 RTU_1	AI_00_04	TANK 1 DEL FLOW		MLD	22.0696			0	0
14	524336	0 RTU_1	AI_00_05	TANK 2 DEL FLOW		MLD	16			0	1
15	524344	0 RTU_1	AI_00_06	DELIVERY PRESSURE		M	150			0	1
16	524352	0 RTU_1	AI_00_07	OUTFLOW TANK 1		MLD	10.4456			0	0
17	524360	0 RTU_1	AI_00_08	OUTFLOW TANK 2		MLD	50			0	1
18	524368	0 RTU_1	AI_00_09	COMBINED FLOW		MLD	15.8599			0	0
19	524376	0 RTU_1	AI_00_10	ANALOG INPUT 10		NONE	100			0	1
20	524384	0 RTU_1	AI_00_11	ANALOG INPUT 11		NONE	58.5196			0	0
21	524392	0 RTU_1	AI_00_12	ANALOG INPUT 12		NONE	65.5971			0	0
22	524400	0 RTU_1	AI_00_13	ANALOG INPUT 13		NONE	63.2481			0	0
23	524408	0 RTU_1	AI_00_14	ANALOG INPUT 14		NONE	77.3983			0	0
24	524416	0 RTU_1	AI_00_15	ANALOG INPUT 15		NONE	97.0357			0	0
25	524424	0 RTU_1	AI_00_16	ANALOG INPUT 16		NONE	58.8199			0	0
26	524432	0 RTU_1	AI_00_17	ANALOG INPUT 17		NONE	69.8059			0	0
27	524440	0 RTU_1	AI_00_18	ANALOG INPUT 18		NONE	100			0	1
28	524448	0 RTU_1	AI_00_19	ANALOG INPUT 19		NONE	91.0791			0	0
29	524456	0 RTU_1	AI_00_20	ANALOG INPUT 20		NONE	91.3285			0	0
30	524464	0 RTU_1	AI_00_21	ANALOG INPUT 21		NONE	81.2714			0	0
31	524472	0 RTU_1	AI_00_22	ANALOG INPUT 22		NONE	95.248			0	0

- Select cell A1
- Select Data
- Select Get External Data
- Select Edit Query



## FlexBase - Getting Started

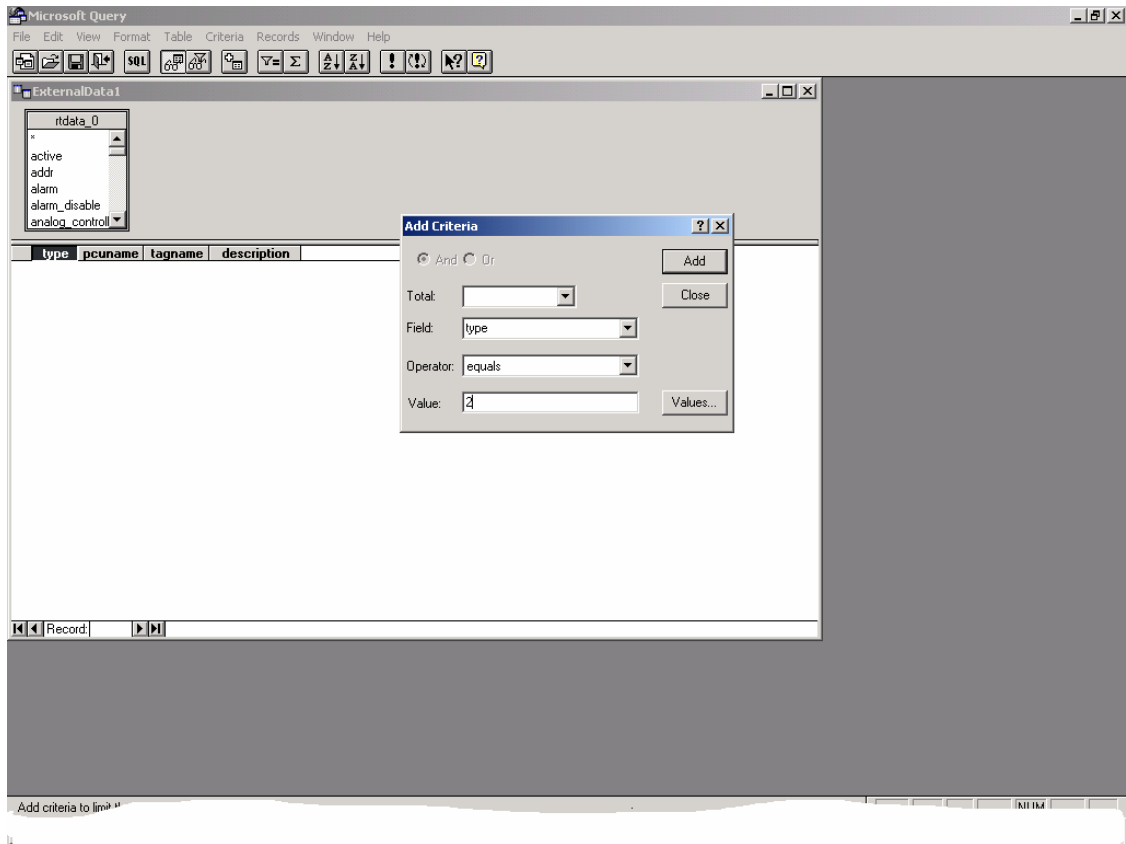


Delete most of the fields (by highlighting the fields and hitting the “delete” button on the keyboard). Leaving just the following fields :

1. Type
2. PCU
3. Tagname
4. Description



## FlexBase - Getting Started



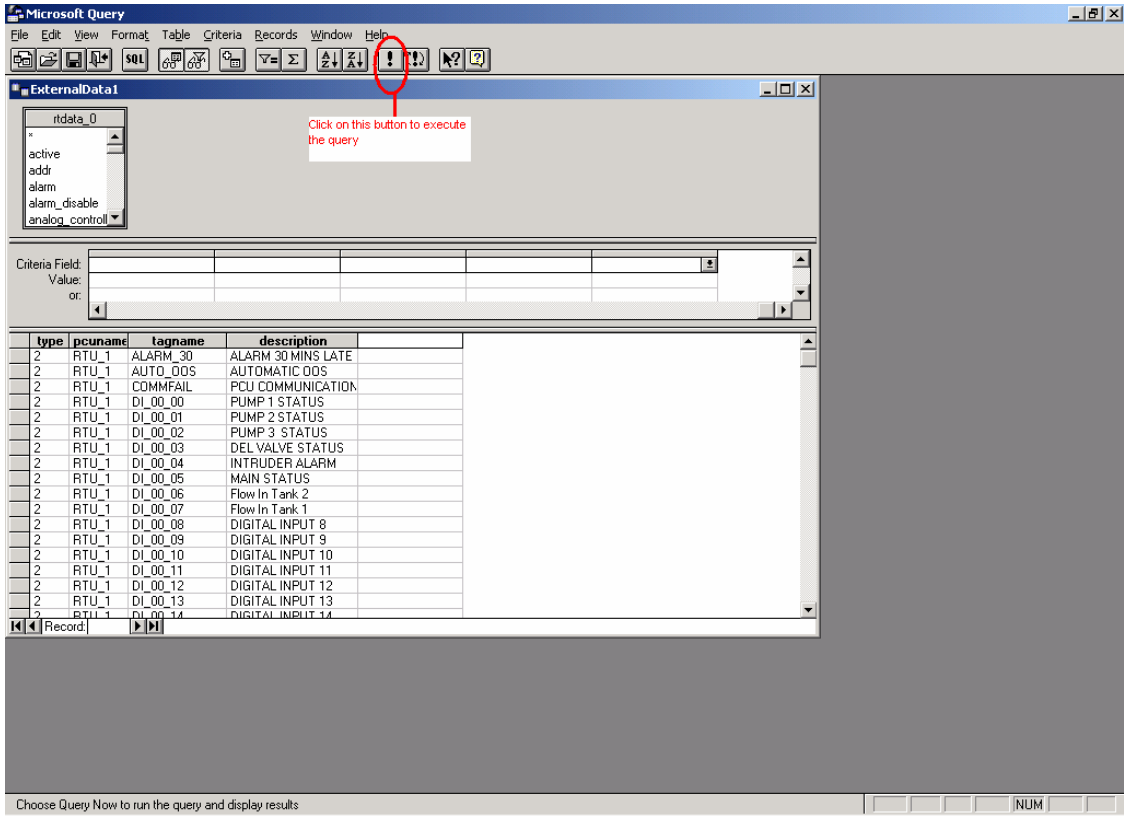
From the “Criteria” menu select “Add criteria”.  
From the “field” drop down box select “type”.  
From the “Operator” drop down box select “equals”.  
From the “Value” drop down box select “2”.

This will narrow down our query to produce a list of all the status points in our real-time database table.





## FlexBase - Getting Started



Now click on the "File" menu and select "Return data to Microsoft Excel".



## FlexBase - Getting Started

Microsoft Excel

File Edit View Insert Format Tools Data Window Help

100%

A1

Book1

1	type	pcuname	tagname	description	E	F	G	H	I	J
2	2	RTU_1	ALARM_30	ALARM 30 MINS LATE						
3	2	RTU_1	AUTO_OOS	AUTOMATIC OOS						
4	2	RTU_1	COMMFAIL	PCU COMMUNICATIONS						
5	2	RTU_1	DI_00_00	PUMP 1 STATUS						
6	2	RTU_1	DI_00_01	PUMP 2 STATUS						
7	2	RTU_1	DI_00_02	PUMP 3 STATUS						
8	2	RTU_1	DI_00_03	DEL VALVE STATUS						
9	2	RTU_1	DI_00_04	INTRUDER ALARM						
10	2	RTU_1	DI_00_05	MAIN STATUS						
11	2	RTU_1	DI_00_06	Flow In Tank 2						
12	2	RTU_1	DI_00_07	Flow In Tank 1						
13	2	RTU_1	DI_00_08	DIGITAL INPUT 8						
14	2	RTU_1	DI_00_09	DIGITAL INPUT 9						
15	2	RTU_1	DI_00_10	DIGITAL INPUT 10						
16	2	RTU_1	DI_00_11	DIGITAL INPUT 11						
17	2	RTU_1	DI_00_12	DIGITAL INPUT 12						
18	2	RTU_1	DI_00_13	DIGITAL INPUT 13						
19	2	RTU_1	DI_00_14	DIGITAL INPUT 14						
20	2	RTU_1	DI_00_15	DIGITAL INPUT 15						
21	2	RTU_1	DI_00_16	DIGITAL INPUT 16						
22	2	RTU_1	DI_00_17	DIGITAL INPUT 17						
23	2	RTU_1	DI_00_18	DIGITAL INPUT 18						
24	2	RTU_1	DI_00_19	DIGITAL INPUT 19						
25	2	RTU_1	DI_00_20	DIGITAL INPUT 20						
26	2	RTU_1	DI_00_21	DIGITAL INPUT 21						
27	2	RTU_1	DI_00_22	DIGITAL INPUT 22						
28	2	RTU_1	DI_00_23	DIGITAL INPUT 23						
29	2	RTU_1	DI_00_24	DIGITAL INPUT 24						
30	2	RTU_1	DI_00_25	DIGITAL INPUT 25						
31	2	RTU_1	DI_00_26	DIGITAL INPUT 26						

Ready NUM



## 2. Registration of Hardware Keys

For security purposes Hardware Keys have to be activated on a product-by-product scenario.

If for some reason your product is not enabled, when you open OPC RTU Configurator, the following screen may appear:



The image shows a 'Product Registration' dialog box with a blue title bar and a close button. The text inside reads: 'If you have purchased this product and received a Product ID from your supplier, please enter your Product ID here EXACTLY as it appears in the instructions. Otherwise, if you want to purchase the program, send this Registration ID to the supplier of this product to receive your Product ID.' Below the text, there is a 'Registration ID' field containing the text '0IHK-1SEU-FUE6-2TX1' and a 'Product ID' field consisting of three empty sub-fields. At the bottom, there are 'OK' and 'Cancel' buttons.

Copy the Registration ID by selecting it and paste into your email facility

Send email to [sales@realflex.com](mailto:sales@realflex.com) with your details and the Registration ID

Upon receiving your request we can validate that you are the correct customer and we will initialize the 'Product ID' for you immediately using in-house software. Upon receiving the email from us please enter the details into the relevant section, therefore initializing the product.



Upgrading Users or IO Limit on Hardware Keys for Flex.Base

If upgrading 'Hardware Keys for either the IO count or the amount of Users please follow the following steps:

- Contact your supplier and place an order for the upgrade to the current system. There is no need to return 'Hardware Keys'

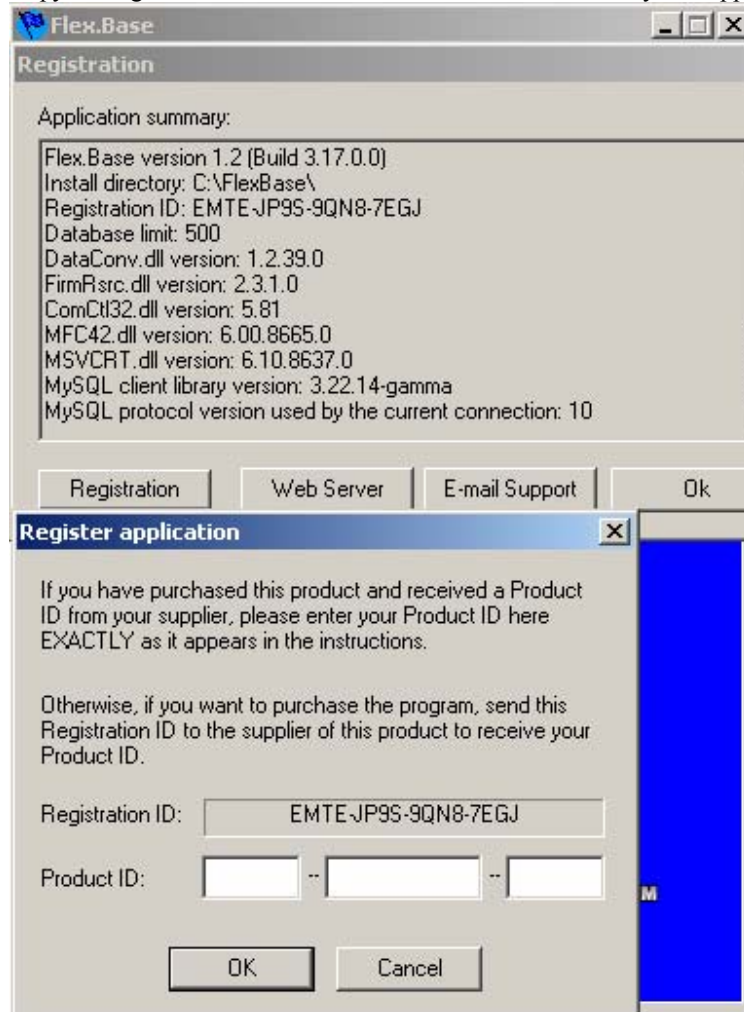
In order to proceed with the order please follow these steps

Go to 'File'

'Registration'

Select 'Registration'

Copy the registration number into an email and forward it to your supplier.



Your supplier will re-code your 'Hardware Key' and email a new 'Product ID for the system.



Upgrading Users or IO Limit on Hardware Keys for Flex.Serv

If you are upgrading 'Hardware Keys for maximum number of Users please follow the following steps:

- Contact your supplier and place an order for the upgrade to your current system. There is no need to return your current 'Hardware Keys'

In order to proceed with the order please follow these steps

- Attach your Hardware Key to the QNX machine and start Flex.Serv.
- Ensure that the QNX machine is connected to the network
- On the Microsoft Windows PC start the "Flex.Serv Registration" utility from the Flex.Base program group (Start button -> Programs -> Flex.Base). This will bring up the configuration window.
- Type in the correct IP address of the QNX PC.
- Copy the Registration ID for the Flex.Serv Key
- Copy the registration number into an email and forward it to your supplier.

Your supplier will re-code the 'Hardware Key' and email a new 'Product ID for the system.



## Appendix A. Utilities

### Reset Project Update Time

This option allows operators to update their databases with earlier database changes

Click on “edit project configuration” from the “Projects” menu.

**Edit Configuration**

Project Settings | Historical filter | Real-time filter

General

Project database name: Demo

Project display name: Demo

Communication Settings

Flex.Serv IP: . . .

Port (default 910): 910 Add

IP Address	Port
192.168.0.51	910

Advanced ... Remove

Project Settings

Flex.Win Project ID: 1895f042-7313-495f-b4ff-ed369b18363

Import from registry Import from Realflex

Database Settings

MySQL server (name or IP): localhost Port (3306): 3306

MySQL user name: FlexBase password: [masked]

Project Database Options Check / create the database

OK Cancel

Click on “Project Database Option”.



Category	Records count	Historical Count	
PCU	0		
ANALOG	0	0	
METER	0	0	
STATUS	0	0	
TANK	0		
Historical events	0		
Historical data	0		

Start date for historical events: 01/01/1980 Set

Start date for historical data: 01/01/1970 Set

Life time (days): 90 check: 01:59:00 Set

Reset update date

Last update time: 01/01/1970 03/11/2003 Set

Erase Records in Database OK

Tick the “Reset update date” box  
Now you will be bring in updates for the project starting from a new specified date.



**Project Database Options** [X]

Category	Records count	Historical Count	
PCU	0		
ANALOG	0	0	
METER	0	0	
STATUS	0	0	
TANK	0		
Historical events	0		
Historical data	0		

Start date for historical events: 01/01/1980 [▲▼] [Set]

Start date for historical data: 01/01/1970 [▲▼] [Set]

Life time (days): 90 [▲▼] check: 01:59:00 [▲▼] [Set]

**Reset update date**

Last update time: 01/01/1970 [▼] 03/11/2003 [▲▼] [Set]

[Erase Records in Database] [OK]

e.g Bring in all the updates since 03<sup>th</sup> November 2003.  
Click on “Set” and then “OK”





## Data Lifetime

This option allows operators to change the duration for which data is kept in the SQL database (Default is 90 Days), before being removed to maintain the database at a reasonable size.

Click on “edit project configuration” from the “Projects” menu.

Category	Records count	Historical Count
PCU	0	
ANALOG	0	0
METER	0	0
STATUS	0	0
TANK	0	
Historical events	0	
Historical data	0	

Start date for historical events: 01/01/1980 Set

Start date for historical data: 01/01/1970 Set

Life time (days): 90 check: 01:59:00 Set

Reset update date

Last update time: 01/01/1970 03/11/2003 Set

Erase Records in Database OK

Life time (days) can be adjusted to suit specific installations.

Check : is the time of the day when the database is checked for data older that the lifetime.

If you change any of these fields, you need to click on Set button to have these changes applied.



## Converting a 1.x project to a 2.0 project

To convert an existing 1.x project to a 2.0 project there are several steps to be followed :

1. Create the 2.0 empty database.
2. Create list of PCU's and Tags in the empty database.
3. Run the FlexBase converter utility.

### 1. Create the 2.0 empty database

Open Flex.Base



From the “Projects” menu select “Add new project configuration”.



Create a new database called “Project2”. (How to add a database is described in section 1.6)

**Add New Configuration**

Project Settings

General

Project database name: Project2

Project display name: Project2

Communication Settings

Flex.Serv IP: . . .

Port (default 910): 910 Add

IP Address	Port
192.168.0.51	910

Advanced ... Remove

Project Settings

Flex.Win Project ID: 14aae70f-5abb-4efb-9583-72ad2e4c5ac5

Import from registry Import from Realflex

Database Settings

MySQL server (name or IP): localhost Port (3306): 3306

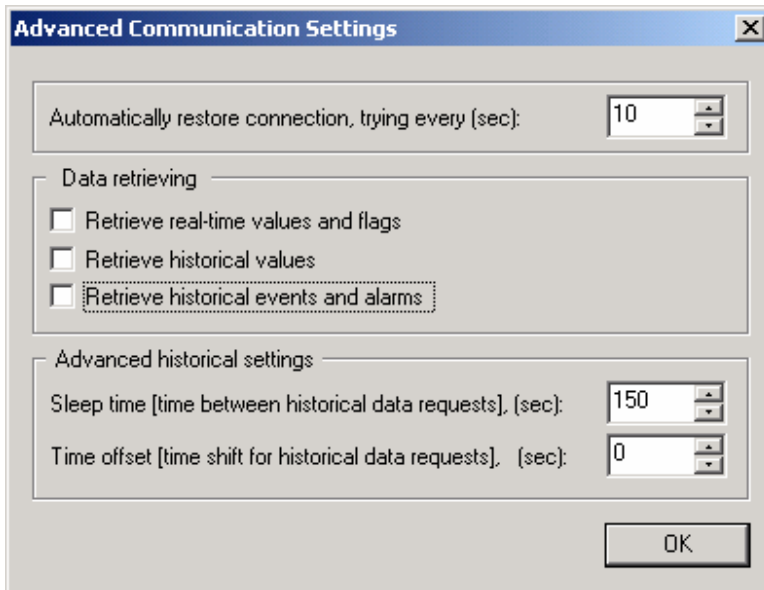
MySQL user name: Administrator password: xxxxxxxx

Project Database Options Check / create the database

OK Cancel

Click on the “Advanced” button and untick the options :

1. Retrieve real-time values and flags
2. Retrieve historical values
3. Retrieve historical events and alarms



Click “OK”.

Click on “Check/create the database” button to ensure “Project2” is created

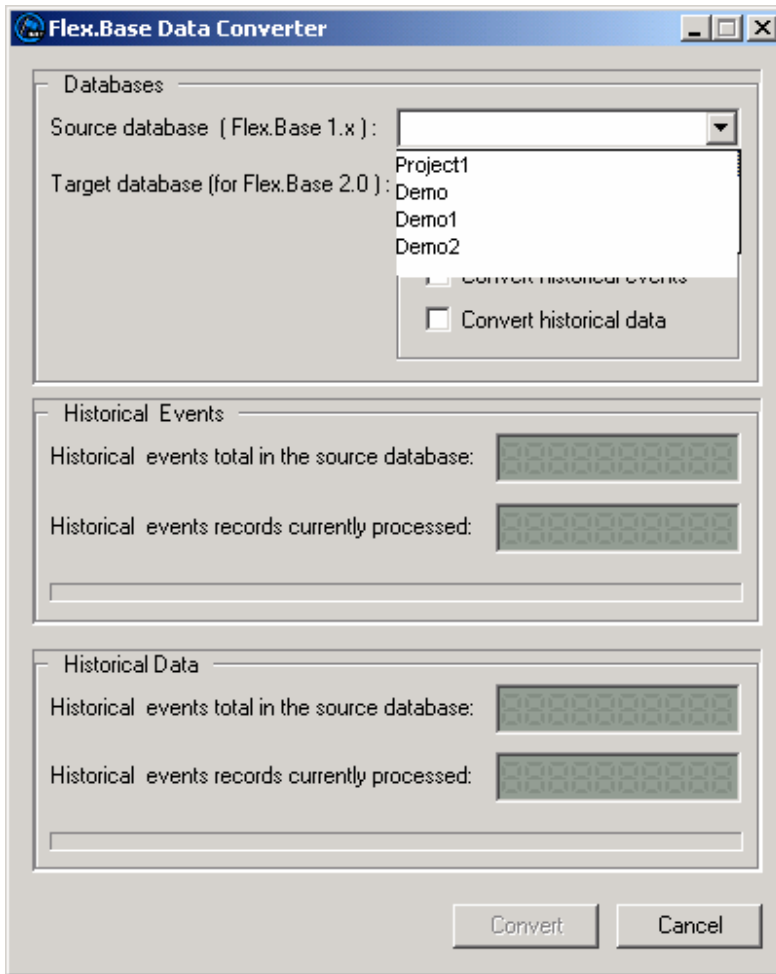
## 2. Create list of PCU’s and Tags in the empty database.

Click on the Project tab, Select “Start data collecting”  
Click on OK  
Wait until the log shows “Project update finished successfully”  
Click on Project tab, Select “Stop data collecting”  
Click on “OK”  
Exit FlexBase

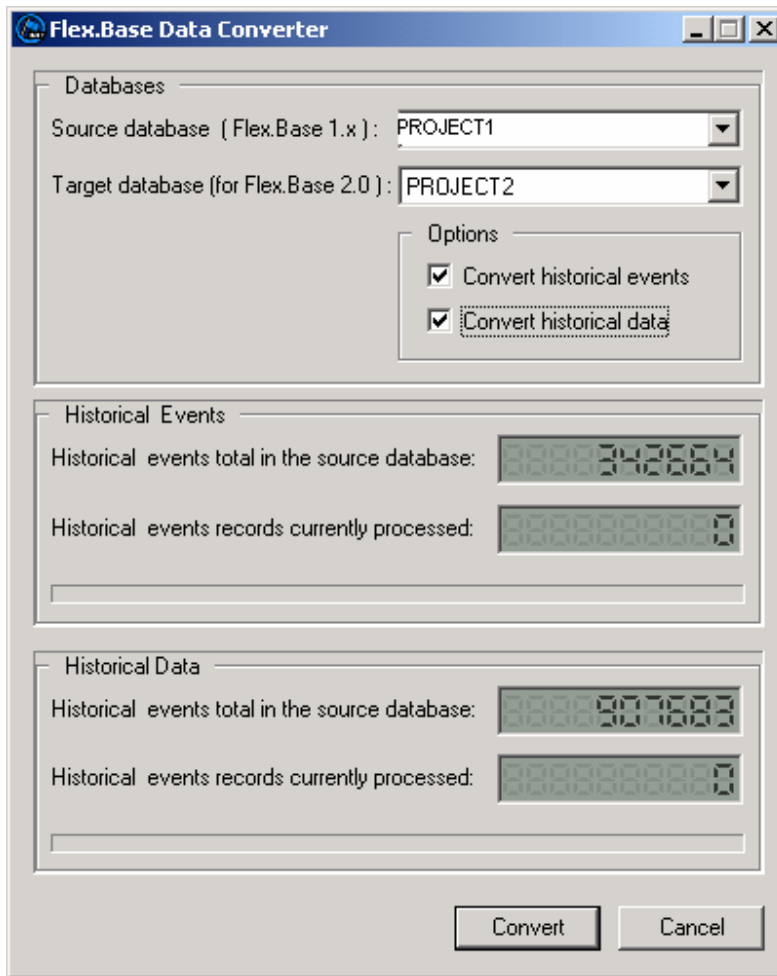
## 3. Run the FlexBase converter utility

From the “Start” menu select “Programs”-> “FlexBase” -> FBConverter.





Select a “Source database” to be converted from the drop down list.  
Select a “Target database” (Project2). Project2 will be a Flex.Base 2.0 version of the  
“Source database”.

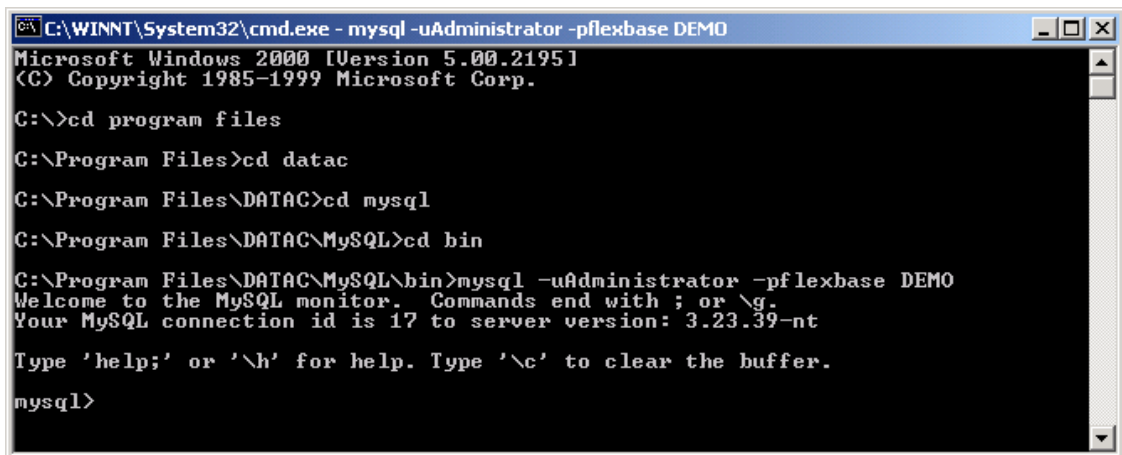


Insure you tick  
“Convert historical events” and  
“Convert historical data”  
Click on the “Convert” button  
Click on OK



## Retrieving historical events and values from database

To retrieve data from the Flex.Base database, use any ODBC client. An example test client is **mysql.exe** utility provided with the Flex.Base package. From the command prompt change to the following directory “c:\Program Files\DATAC\MySQL\bin”. When using the **mysql.exe** utility specify any valid user name and password in the command. Specify database name as third parameter in third command line. Look at the example below.



```

C:\WINNT\System32\cmd.exe - mysql -uAdministrator -pflexbase DEMO
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-1999 Microsoft Corp.

C:\>cd program files
C:\Program Files>cd dataac
C:\Program Files\DATAC>cd mysql
C:\Program Files\DATAC\MySQL>cd bin
C:\Program Files\DATAC\MySQL\bin>mysql -uAdministrator -pflexbase DEMO
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17 to server version: 3.23.39-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>

```

Fig. 12.

After the **MySQL>** prompt appears type any **SQL select** query. Please look at the following example.

Use the following SQL command to retrieve information from the realtime database. In this case we are using the “DEMO” project.

```

Select    rtuname
from      rtdata;

```

For more information refer to the SQL guide.

NOTE: Construct queries paying attention to the following parameters because of performance reasons. i.e. parameters are where a clause must be in the same sequence as in the table indexes (read more about this in your SQL server guide).



## Tuning security settings

Tune Flex.Base security settings using Flex.Base Administrator utility (fig. 7), in addition use 'Database Explorer' and 'Drop database' to remove old data and release disk space.

The Flex.Base Administrator utility is started by selecting "security tuning" from the "Tools" menu in Flex.Base. Logon with any administrator account registered in Flex.Base. After installation the default administrator user name is 'Administrator' and the default password is 'flexbase'.

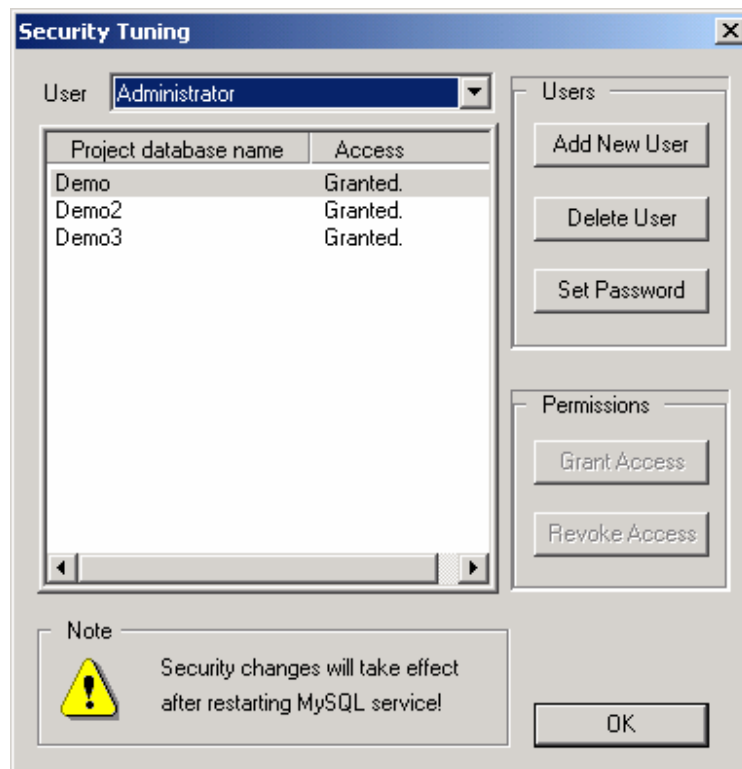


Fig. 7.

There are three pre-defined accounts in Flex.Base: 'Administrator', 'Operator' and 'Viewer'. Create any user account with the administrator rights, operator rights or viewer rights.

- Administrators have no restrictions. Only users with administrator rights can create and remove project databases and tables. Only administrators can create and remove other Flex.Base user accounts, and only administrators can restrict other users access to project databases.





- Operators can read and edit historical events, values and control tables.
- Viewers can only read data.

Note: 'Grant Access' and 'Revoke Access' are useful only on operator and viewer accounts, i.e. you cannot restrict administrator's access to any project databases.

The 'Separate' option gives permission to logically remove old historical events and value records, but to release unused space use the 'Compact Data' option. This is a time consuming operation depending on the database size. Exit Flex.Base before compacting the database. It is essential to free space for storing temporary files on the partition where Flex.Base is installed.

NOTE: Before performing "dropping" any databases within "Database explorer" ensure the MySQL service is stopped using the following command:

*Mysqldadmin.exe -uUSER\_NAME -pPASSWORD shutdown*

Where USER\_NAME and PASSWORD is the administrator's account. Safely backup files in the INSTALLDIR\MySQL\data\DATABASE\_NAME directory, where DATABASE\_NAME is the same as the project database name and INSTALLDIR is the directory name that Flex.Base has been installed onto.



## Appendix B. Database tables.

### Flex.Base 2.0 Database Structure.

#### 1. Table RTDATA (the project table).

Note : Types (AMSTP) means A=Analog M=Meter S=Status T=Tank P=PCU

Field name	SQL type	Types	Description
ACTIVE	INTEGER	AMSTP	1=Tag/PCU is active (fl0)
ADDR	INTEGER	----P	RealFlex Address for PCU/RTU/PLC
ALARM	INTEGER	AMSTP	1=Tag/PCU has Alarm
ALARM_DISABLE	INTEGER	AMST-	1=Alarms are disabled on tag (fl3)
ANALOG_CONTROLLABLE	INTEGER	A----	1=Analog tag is controllable (fl11)
ANALOG_HHLM	DOUBLE	A----	Analog high-high limit
ANALOG_HIGH	INTEGER	A----	1=Analog signal high alarm limit exceeded
ANALOG_HIGHHIGH	INTEGER	A----	1=Analog signal high-high alarm limit exceeded
ANALOG_HLM	DOUBLE	A----	Analog high limit
ANALOG_LLLM	DOUBLE	A----	Analog low-low limit
ANALOG_LLM	DOUBLE	A----	Analog low limit
ANALOG_LOW	INTEGER	A----	1=Analog signal low alarm limit exceeded
ANALOG_LOWLOW	INTEGER	A----	1=Analog signal has low-low alarm limit exceeded
ANALOG_MAX_EU	DOUBLE	A----	Analog max. EU value.
ANALOG_MIN_EU	DOUBLE	A----	Analog min. EU value.
ANALOG_RATEOFCHANGE	INTEGER	A----	1=Analog signal rate of change alarm limit exceeded
ANALOG_VALUE	DOUBLE	A----	Value of Analog tag/Signal
AVAIL_VOL	DOUBLE	---T-	Available tank volume
CNTRL_TYPE	INTEGER	--S--	0=Indicates Status Input and >0 = indicates Status Output
COLDSTART	INTEGER	AMSTP	Flag to indicate tag has been coldstarted
COMM_EFF	DOUBLE	----P	Communications efficiency to



			PCU/RTU/PLC
CONTROL_PENDING	INTEGER	AMSTP	1=Control has been issued but not acknowledged
CONTROL_TAGGED	INTEGER	AMST-	1=Control tag is attached to tag (fl5)
DATA_ERR	INTEGER	----P	Data Errors on Communications to PCU/RTU/PLC
DESCRIPTION	VARCHAR(21)	AMSTP	Description of tag or PCU
EX_DESCRIPTION	VARCHAR(50)	AMST-	Extra description field
FLOW_RATE	DOUBLE	---T-	Tank flow rate
FRACTIONS	DOUBLE	---T-	Tank fractions
GOOD_REQ	INTEGER	----P	Good Communications Scans to PCU/RTU/PLC
GRAVITY	DOUBLE	---T-	Gravity for tank tag
HHLM_COLOR	INTEGER	A-S--	Analog high-high QNX color index
HISTORICAL_LOG	INTEGER	AMS--	1=Tag is historically logging data (fl1)
HLM_COLOR	INTEGER	A-S--	Analog high QNX color index
INFORMATION_TAGGED	INTEGER	AMST-	1=Information tag is attached to tag (fl4)
INVALID	INTEGER	AMSTP	1=Data is invalid
IS_STOPPED	INTEGER	AMS--	1=store historical data from Realflex to the Flex.Base project database
IS_STOPPED_RT	INTEGER	AMSTP	1=store real-time data from Realflex to the Flex.Base project database
LAST_HIST_UPDATE	DATETIME	AMS--	Last history update
LASTVOL	DOUBLE	---T-	Last tank volume
LEVEL_0	DOUBLE	---T-	Tank level 0
LEVEL_1	DOUBLE	---T-	Tank level 1
LEVEL_2	DOUBLE	---T-	Tank level 2
LLLM_COLOR	INTEGER	A-S--	Analog low-low QNX color index
LLM_COLOR	INTEGER	A-S--	Analog low QNX color index
MANUAL_OVERWRITE	INTEGER	AMSTP	1=Signal is manually overwritten
MAX_VOL	DOUBLE	---T-	Maximum tank volume
METER_DAILY	INTEGER	-M---	Meter DAILY value
METER_FACTOR	DOUBLE	-M---	Meter factor
METER_GROSS	INTEGER	-M---	Meter GROSS value
METER_HOURLY	INTEGER	-M---	Meter HOURLY value
METER_LAST_GOOD	INTEGER	-M---	Meter LAST_GOOD value
METER_LAST_HOUR	INTEGER	-M---	Meter LAST_HOUR value
METER_MONTHLY	INTEGER	-M---	Meter MONTHLY value
METER_NET	INTEGER	-M---	Meter NET value
METER_ROLLOVER	INTEGER	-M---	Meter rollover value
METER_TYPE	INTEGER	-M---	Meter type



METER_YEARLY	INTEGER	-M---	Meter YEARLY value
METER_YESTERDAY	INTEGER	-M---	Meter YESTERDAY value
NO_REPLY	INTEGER	AMSTP	1=No Communications to PCU/RTU/PLC
NO_RESPONSE	INTEGER	----P	No Response Communications to PCU/RTU/PLC
NORM_STATE	INTEGER	--S--	Numeric value indicating which state is normal i.e. 0,1,2 or 3
NUM_BITS	INTEGER	--S--	1 or 2 bits in for status point
PCUNAME	VARCHAR(13)	AMSTP	Name of PCU/RTU/PLC
PRODUCT_CODE	INTEGER	---T-	Tank product code
PRODUCT_NAME	VARCHAR(21)	---T-	Tank product name
RETRIES	INTEGER	----P	Retries of Communications to PCU/RTU/PLC
RT_TIMESTAMP	DATETIME	AMSTP	The timestamp for real-time values and flags
RT_TIMESTAMP_MS	INTEGER	AMSTP	Milliseconds for RT_TIMESTAMP field
START_BIT	INTEGER	--S--	Start bit for STATUS_VALUE field
STATES_DESC0	VARCHAR(20)	--S--	Description for status value 0
STATES_DESC1	VARCHAR(20)	--S--	Description for status value 1
STATES_DESC2	VARCHAR(20)	--S--	Description for status value 2
STATES_DESC3	VARCHAR(20)	--S--	Description for status value 3
STATUS_ALARM	INTEGER	--S--	1=Alarm on a status signal
STATUS_OK	INTEGER	--S--	
STATUS_VALUE	INTEGER	--S--	Value of Status tag/Signal
TAG_ID	INTEGER UNSIGNED	AMSTP	Unique Tag index number
TAGNAME	VARCHAR(13)	AMST-	Name of tag or signal
TEMPERATURE	DOUBLE	---T-	Tank temperature
TOTAL_REQ	INTEGER	----P	Total Communications Scans to PCU/RTU/PLC
TYPE	INTEGER	AMSTP	0=Analog 1=Meter 2=Status 3=Tank 5=PCU
UNACKNOWLEDGED	INTEGER	AMSTP	1=Tag/PCU has unacknowledged Alarm
UNAUTHORIZED_STATUS	INTEGER	--S--	
UNITS	VARCHAR(11)	AMST-	Units for tag or signal
VOLUME	DOUBLE	---T-	Current tank volume

## 2. Table HDATA (the history table).



Field name	SQL type	Description
ID	INTEGER UNSIGNED	Unique record index number
TAG_ID	INTEGER	A reference to the corresponding historical tag in RTDATA table (HDATA.TAG_ID = RTDATA.TAG_ID)
SUBTYPE	INTEGER	Subtype for meter tags (0=NET,1=HOUR,2=DAY,3=MONTH,4=YEAR)
VALUE	DOUBLE	Tag value. All tag values are stored as DOUBLE.
TIME	DATETIME	Tag value timestamp.
TIME_MS <sup>1</sup>	INTEGER	Milliseconds for TIME field.

### 3. Table HEVENTS (the historical events table).

Field name	SQL type	Description
DESCRIPTION	VARCHAR(21)	Text from EVENT.YYYYMMDD file
EVENT_ID	INTEGER UNSIGNED	Unique record index number
EVTYPE	INTEGER	
TAG_ID	INTEGER	A reference to the corresponding TAG in RTDATA table (HDATA.TAG_ID = RTDATA.TAG_ID). This field may be -1 for third-party software events.
PCUNAME	VARCHAR(13)	Name of PCU/RTU/PLC
TIME	DATETIME	Event timestamp
TIME_MS <sup>2</sup>	INTEGER	Milliseconds for TIME field.

<sup>1</sup> This field it exists only in Flex.Base 2.0 MySQL Edition databases because MySQL doesn't support milliseconds in DATETIME data type.

<sup>2</sup> This field is exists only in Flex.Base 2.x MySQL Edition databases because MySQL doesn't support milliseconds in DATETIME data type.



RealFlex Technologies,  
2218 Northpark Drive, Suite 202,  
Kingwood, Texas, 77339, USA  
Tel: +1 281 348 2341, Fax: +1 281 348 2340  
Email: [sales@realflex.com](mailto:sales@realflex.com)  
<http://www.realflex.com/>

or

RealFlex Technologies Ltd,  
Limerick Business Complex,  
Raheen Business Park, Limerick, Ireland.  
Tel: +353 61 308884, Fax +353 61 308883,  
Email: [sales@realflex.com](mailto:sales@realflex.com)  
<http://www.realflex.com/>

