WebAccess

Driver Configuration Manual

MS Access 2010

BWDBDrv.DLL

Driver date: 2018/3/9

English Version 1.1

Revision History

Date	Version	Author	Reviewer	Description
2018-10-15	1.0	Neal.Chen	Joseph.Chiu	Initial Release
2018-11-6	1.1	Neal.Chen	Neal.Chen	Update Error Code

Table of Contents

1. Introduction to BWDB driver to database

The BWDB Device Driver allows data to be read from a database as if it were a PLC, controller or other device. This may be required if a device driver is not available for direct connection to the device, if the data is manually entered or other reasons.

This is a READ ONLY driver. Use a Script or Pushbutton keymacro to write data to a database. See the Engineering Manual, sections 11.1.96 <SQLDIRECT>, 12.4.37 SQLDIRECT, and 12.4.39 SQLPREPARE for more information on writing data to a database.

Normally, WebAccess logs real-time data to databases using the Log to ODBC feature in SCADA Node properties. Please refer to the Engineering Manual, section 3.2.22 Data Log to ODBC for more information. To read data from databases, WebAccess has Scheduled Reports, Analog Tag Log, Discrete Tag Log and Text Tag Log to allow users to view data from standard database logging (see Engineering Manual, Sections 21.1 and 21.2).

The WebAccess SCADA Node uses an ODBC connection to a Database to mimic a device driver. (ODBC is a Microsoft acronym for Open Data Base Connection).

An ODBC DSN (Database Source Name) must be configured on the SCADA node.

The Database can be on the SCADA node or installed on a Database Server that the ODBC DSN has access to over the network

The WebAccess BWDB driver to a database uses an API type comport. This is a virtual communications port. It does not need to match an actual COM port number. It is recommended to use a comport number above 4 to avoid conflicts with a real comport on the SCADA node (for example a serial comport).

The WebAccess Tags are configured using the Field Names (or Column Names) in the Database as the Address. The Database can be a Microsoft Access, Microsoft SQL Server, Oracle 9i, MYSQL, or PostgreSQL. Other relational databases supported by a Microsoft ODBC DSN may also work.

Unlike other WebAccess drivers, there are no pre-configured Parameters or Blocks in the BWDB device driver. You must configure Parameters before configuring Tags. You must configure parameters and block types before configuring Blocks.

You need to know the Table Name inside the Database. You will need a copy of the Database Software Program to open the Database to determine the Table Name.

You should also note the Field Names (Column Names) so that you can sort the data in the way you want using the SQL SELECT condition.

You need to know the Field Names (Column Names) in your database in order to arrange the data using WHERE or ORDER BY or other conditions to the SQL SELECT command. Most users will want the most recent data, so your database will need one or more date and time fields sorted in descending order.

2.1. ODBC DSN on SCADA Node

The ODBC DSN must be configured on the SCADA node. It is recommended to use a ‘System DSN’ to make the connection available regardless of the user that is logged on to the SCADA node.

The DSN and Table Name are used in the BWDB Device Configuration.

To View an ODBC DSN:

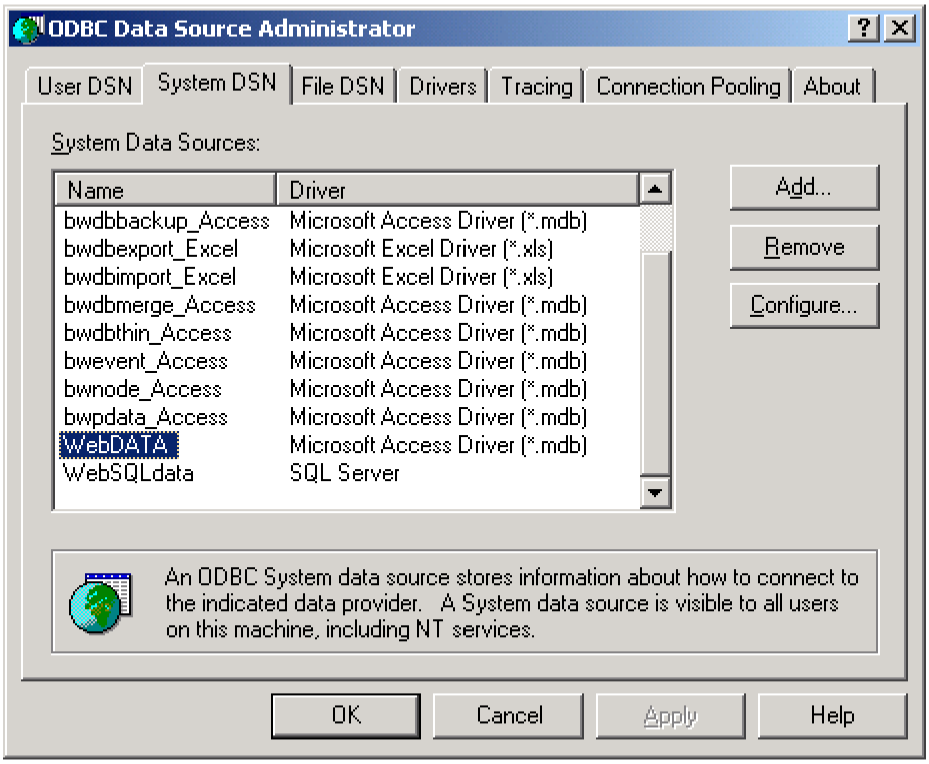
1. Start -> Control Panel -> Administrative Tools -> Data Sources (ODBC)
2. Select System DSN tab.

   The DSN is used in the BWDB Device Configuration.
3. Click OK or CANCEL to close.
4. Select Configure to Modify the DSN, confirm it details (e.g. what database it is connected to).

Note- the Database can be on the SCADA node or installed on a Database Server that the SCADA node has access to over the network.

5. Select ADD to create a new DSN.

The Name of the DSN is required by the WebAccess API Comport configuration.



Name	Driver
bwdbbackup_Access	Microsoft Access Driver (*.mdb)
bwdbexport_Excel	Microsoft Excel Driver (*.xls)
bwdbimport_Excel	Microsoft Excel Driver (*.xls)
bwdbmerge_Access	Microsoft Access Driver (*.mdb)
bwdbthin_Access	Microsoft Access Driver (*.mdb)
bwevent_Access	Microsoft Access Driver (*.mdb)
bwnode_Access	Microsoft Access Driver (*.mdb)
bwpdata_Access	Microsoft Access Driver (*.mdb)
WebDATA	Microsoft Access Driver (*.mdb)
WebSQLdata	SQL Server

An ODBC System data source stores information about how to connect to the indicated data provider. A System data source is visible to all users on this machine, including NT services.

Figure 1.1 ODBC Data Source

2.2. Example Access Database and Table

You need to know the Table Name inside the Database. You will need a copy of the Database Software Program to open the Database to determine the Table Name.

You should also note the Field Names (Column Names) so that you can sort the data in the way you want using the SQL SELECT condition.

The following uses this Access database (named DB2.mdb) and Table1 in that DB2 Database:

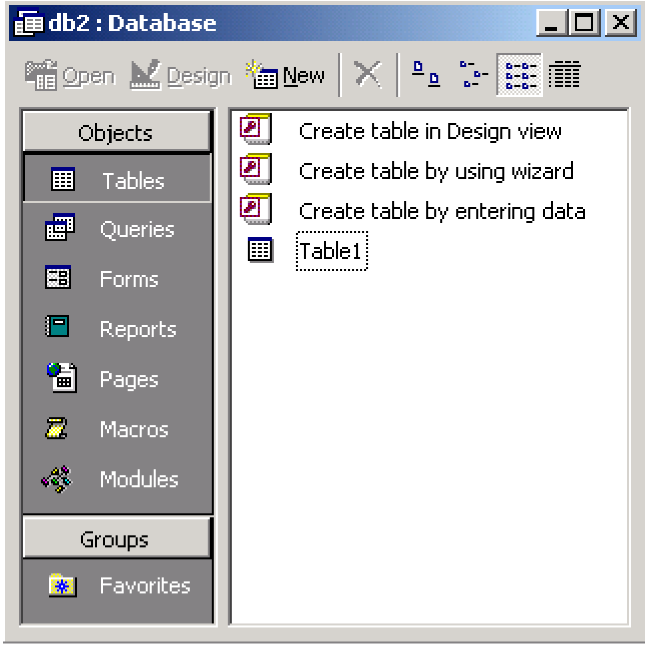
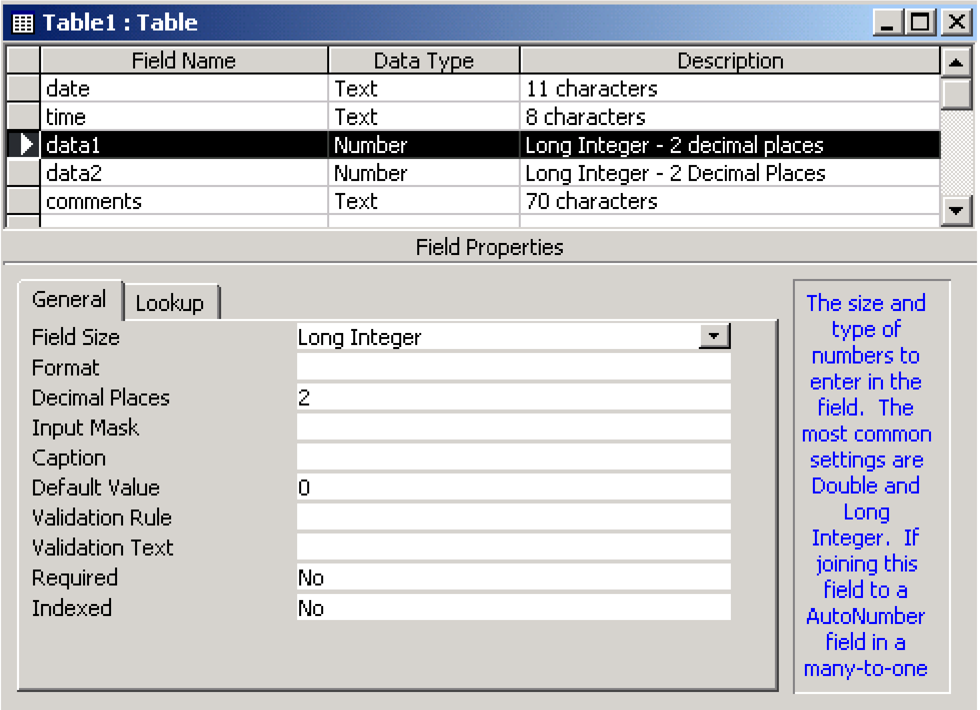


Figure 1.2 Example Access Database – Table named Table1

The Table name will be used in Device Configuration



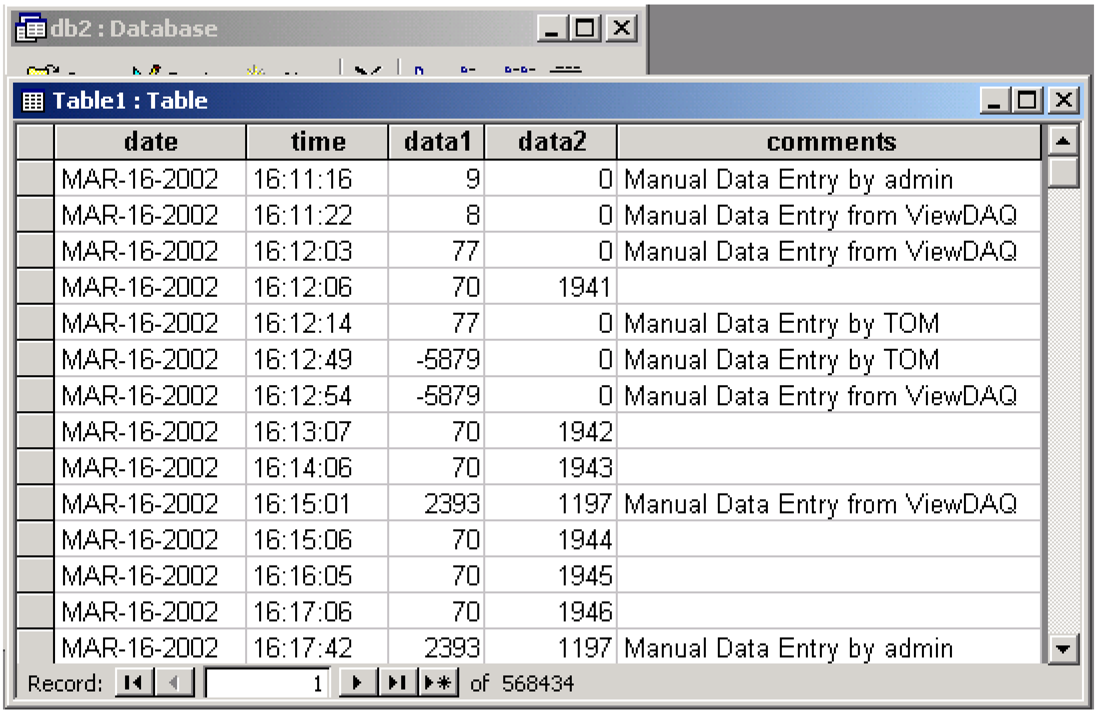
Field Name	Data Type	Description
date	Text	11 characters
time	Text	8 characters
data1	Number	Long Integer - 2 decimal places
data2	Number	Long Integer - 2 Decimal Places
comments	Text	70 characters

Field Size	Long Integer
Format	
Decimal Places	2
Input Mask	
Caption	
Default Value	0
Validation Rule	
Validation Text	
Required	No
Indexed	No

The size and type of numbers to enter in the field. The most common settings are Double and Long Integer. If joining this field to a AutoNumber field in a many-to-one

Figure 1.3 Example Access Database – Table1 Field Names (e.g. Column Names)

The Field Names will be used as the Address for Tag configuration.



db2 : Database

Table1 : Table

date	time	data1	data2	comments
MAR-16-2002	16:11:16	9	0	Manual Data Entry by admin
MAR-16-2002	16:11:22	8	0	Manual Data Entry from ViewDAQ
MAR-16-2002	16:12:03	77	0	Manual Data Entry from ViewDAQ
MAR-16-2002	16:12:06	70	1941	
MAR-16-2002	16:12:14	77	0	Manual Data Entry by TOM
MAR-16-2002	16:12:49	-5879	0	Manual Data Entry by TOM
MAR-16-2002	16:12:54	-5879	0	Manual Data Entry from ViewDAQ
MAR-16-2002	16:13:07	70	1942	
MAR-16-2002	16:14:06	70	1943	
MAR-16-2002	16:15:01	2393	1197	Manual Data Entry from ViewDAQ
MAR-16-2002	16:15:06	70	1944	
MAR-16-2002	16:16:05	70	1945	
MAR-16-2002	16:17:06	70	1946	
MAR-16-2002	16:17:42	2393	1197	Manual Data Entry by admin

Record: 1 of 568434

Figure 1.4 Example Access Database – contents of Table1

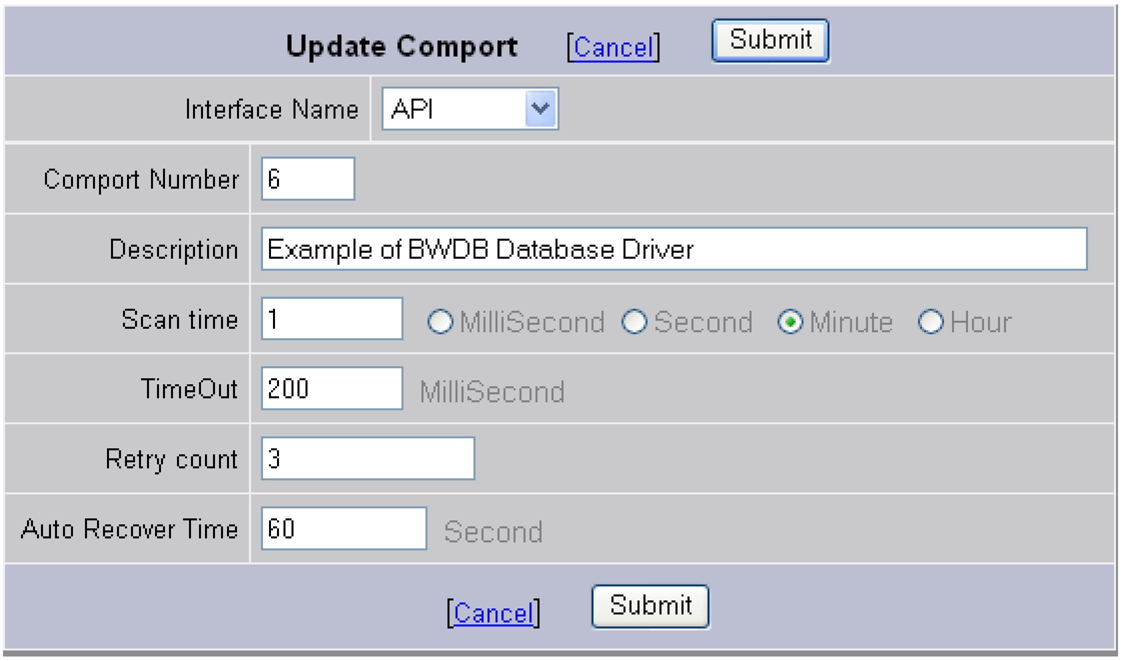
2. API Comport and BWDB device

2.1. Configure an API Comport and a BWDB Device

The TCPIP Comport is usually associated with an Ethernet Network Interface Card on the SCADA Node PC. Any TCPIP compatible medium is supported as long as it complies with Microsoft TCPIP protocol stack. The user should give the setting of comport number, scan time, timeout, retry count, auto recover time & scan devices in parallel by the actual connection requirements.

The steps, in summary, are:

1. Start Internet Explorer **Web Browser**.
2. Enter IP address of the **Project Node**.
3. Use **WebAccess Configuration**.
4. Open or Create a **Project**.
5. Configure a **SCADA node** (the PC that will connect to the automation hardware).
6. Configure a **Comport** for the SCADA Node that is an API type Comport by selecting ADD Comport from SCADA Node Properties.
7. For the **Interface Name**, select **API**. Wait for the page to update.



Interface Name	API
Comport Number	6
Description	Example of BWDB Database Driver
Scan time	1 ○ MilliSecond ○ Second ● Minute ○ Hour
TimeOut	200 MilliSecond
Retry count	3
Auto Recover Time	60 Second

Figure 2.1 API type comport used with BWDB driver

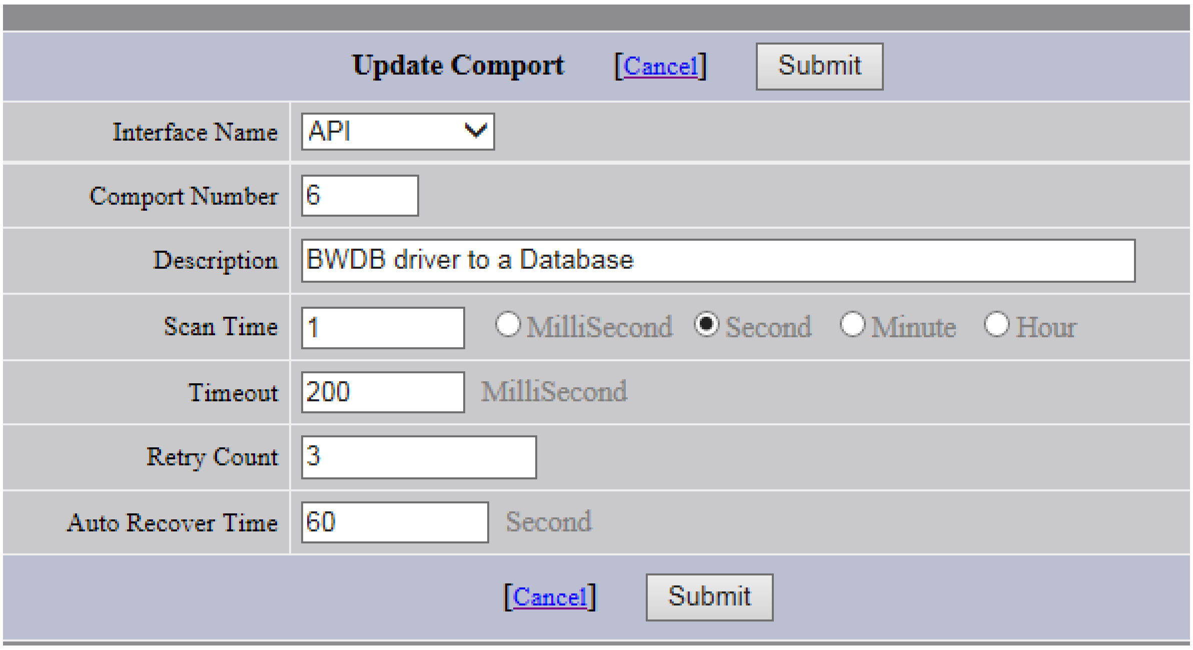
8. Enter a **Comport Number**. This is a virtual communications port. It does not need to match an actual COM port number. It is recommended to use a comport number above 4 to avoid conflicts with a real comport on the SCADA node (for example a serial comport).

9. Configure **Scan Time**, **Timeout**, **Retry** and **Auto Recover**.

See 1.3.2 API Comport Properties in the following section for more information.

10. Press **Submit** to save the API Comport properties.

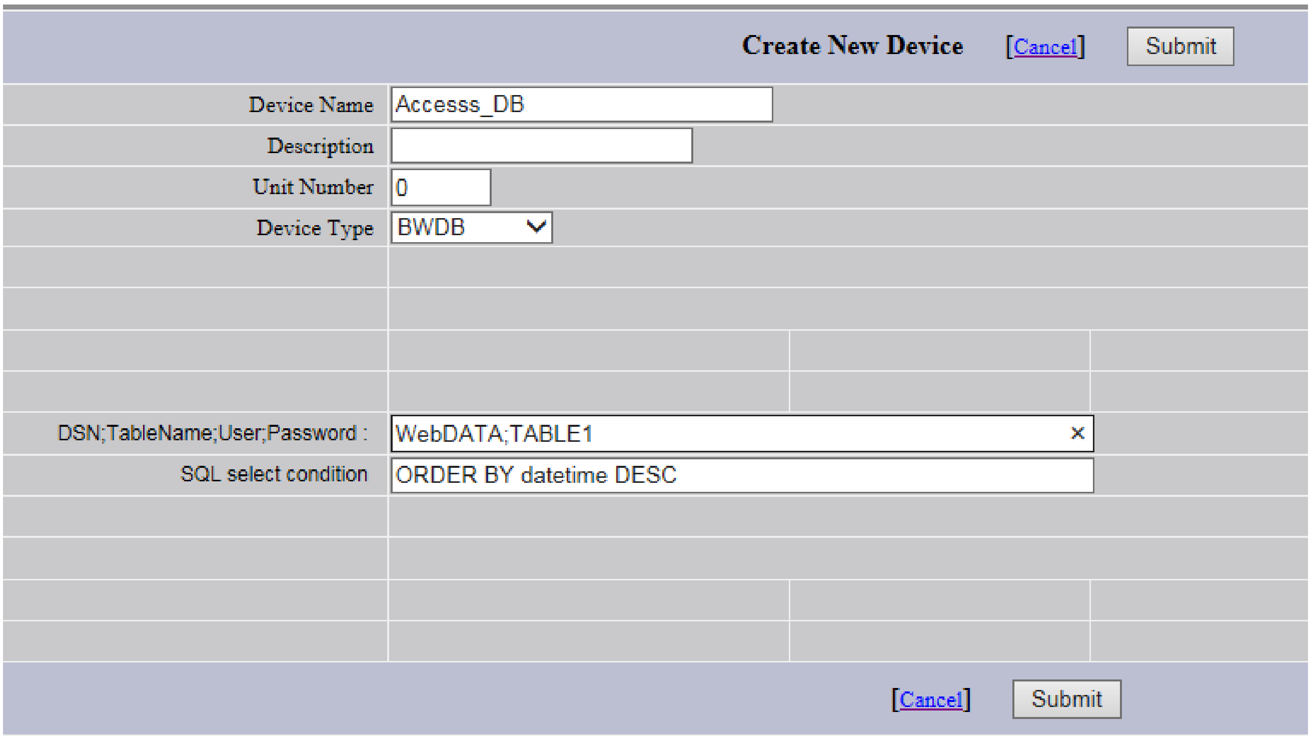
11. Select the API Comport (Port 6 in the Example) from the left menu list.



Interface Name	API
Comport Number	6
Description	BWDB driver to a Database
Scan Time	1 ○ MilliSecond ● Second ○ Minute ○ Hour
Timeout	200 MilliSecond
Retry Count	3
Auto Recover Time	60 Second

Figure 2.2 Example BWDB driver on Port 6

12. Configure a BWDB Device (determines the communications Protocol or Device Driver) using **Add Device**.



Device Name	Accesss_DB
Description	
Unit Number	0
Device Type	BWDB
DSN;TableName;User;Password :	WebDATA;TABLE1
SQL select condition	ORDER BY datetime DESC

Figure 2.3 Example Access Database – DSN named WebData. Table named Table1

13. Select the **Device Type** is **BWDB**. Wait for the Page to update.

14. Enter a **Unit Number**. This is a virtual number. Use whatever makes sense to your end user.

15. Enter **DSN** (Data Source Name) and **Table Name** separated by semi-colon (;). For example WebDATA;TABLE1.

See section 1.1 ODBC DSN on SCADA Node for determining the DSN. This is a DSN on the SCADA node.

16. If you need a User Name and Password, append these to the DSN name and Table name separated by semi-colons (;). The password is usually case sensitive.

Microsoft Access Databases usually do not need a User and Password.

However, SQL Server usually requires a User and Password. If you are using SQL server, enter **DSN;TableName;User;Password**.

17. Enter the **SQL SELECT condition**. This is part of an SQL SELECT command, usually a WHERE or ORDER BY condition.

Most commonly this is an 'ORDER BY fieldname DESC' statement used to arrange the data to be the most recent. Note that the default of most databases is the oldest record is first (so called Ascending order).

If you have a single timestamp, to get the most recent record, an example might be:

ORDER BY timestamp DESC

In the above example database DB2.MDB, Table1 has two columns Date and Time. To get the most recent record the Select condition would be:

ORDER BY Date DESC, Time DESC

Note that Date and Time are field names (column names) in the DB2.MDB Table1. DESC is the SQL syntax for Descending (i.e. last record first).

You have to use the actual Field Names (column names) in your database.

18. Press **Submit** to save the BWDB device properties.
19. You must create Parameters before you can Add Tags or Add Blocks.

If you or someone has already created parameters you can jump to configure a TAG.

Otherwise go to Step 20 Configure Parameters.

Note – You have to create parameters and blocks before you can use Add Tag or Add Block.

2.2. API Comport Properties

2.2.1 Comport Number

This is a virtual communications port. It does not need to match an actual COM port number. It is recommended to use a comport number above 4 to avoid conflicts with a real comport on the SCADA node (for example a serial comport).

2.2.2 Description

This is an optional field used for user reference.

2.2.3 Scan Time

This is the time in milliseconds, seconds, minutes or hours to scan the DSN and the Database. This must match the ability of the network and Database to respond. Very large databases, with many records may take a long time to respond.

If the network, DSN and database cannot respond as fast as the SCAN Time entered, WebAccess will scan at a slower rate.

2.2.4 Timeout

Timeout is the time waited before re-sending a communications packet that did not have a reply. Timeout is in milliseconds.

Timeout specifies how long the software waits for a response to a data request, specifically to wait for a reply from one packet. A recommended value is 3 seconds, longer if the communication device is slow. This is protocol dependent: some protocols do not allow changes in time out.

Combined with Retry count, Timeout also determines time to consider a device or port as BAD.

Timeout is the time to wait since last communication packet sent without a reply. Time is in milliseconds. The slow or poor quality communications require longer timeout. The faster the communications network or device, the shorter the timeout required. Shorter timeouts notify operators of communications failure more quickly.

2.2.5 Retry Count

It means number of times to retry communications if no reply is received from a device. Combined with Timeout, also determines time to consider a device or port as BAD.

In addition, Indicates the number of times after the first attempt has failed that communication should be attempted before indicating a failure. Specifically, how many times to send a single packet after the DSN or Database fails to respond to the first packet. After the retry count is exceeded, all the tags in the packet are marked with asterisks and the next packet of requests is sent. A reasonable value is 3 to 5 times. After this number of tries, the tags in this packet are marked as "fail to respond" (i.e. asterisks) and are disabled. In reality, increasing the number of retries hides failures on the part of the DSN or Database to respond to a request. Essentially, increasing the retries gives the DSN or Database more chances to reply.

2.2.6 Auto Recover Time

Auto Recover Time is the time to wait before attempting to re-establish communications with a BAD device or port.

If communications to the Database is unusually slow due to hardware, communications or network issues, you might consider increasing this value. If communications to the Database fails frequently, you may want to decrease this number in order to have WebAccess try to re-establish communications sooner.

If communications to the DSN Fails (i.e. exceeds Timeout), WebAccess will wait the 'Auto Recover Time' before trying to re-establish communications.

2.3. BWDB Device Properties

2.3.1 Unit Number

This is a virtual number. Use whatever makes sense to your end user. This will appear in Detail Displays and Communication Status Display.

2.3.2 DSN

DSN is the Data Source Name. See section 1.1 ODBC DSN on SCADA Node for determining the DSN. This is a DSN on the SCADA node.

2.3.3 Table Name

You need to know the Table Name inside the Database. You will need a copy of the Database Software Program to open the Database to determine the Table Name.

Append the Table name to the DSN using a semicolon (;). For example: **WebDATA;Table1**

2.3.4 User and Password

Some Databases require a user and password (SQL SERVER using SQL SERVER security, for example). If you need a User Name and Password, append these to the DSN name and Table name separated by semi-colons (;).

For Example:

WebData;Table1;User;Password

The password is usually case sensitive.

Microsoft Access Databases usually do not need a User and Password.

SQL Server usually requires a User and Password.

2.3.5 SQL SELECT condition

This is usually required to arrange the data to read the most recent DATA.

The **SQL SELECT condition** is part of an SQL SELECT command, usually an ORDER BY condition. This is the part of the SELECT statement that follows **SELECT * from Tablename**.

The SELECT Condition is up to 63 characters long, on a single line and is in the format:

[WHERE condition] [and|or condition...]

[GROUP BY column-list]

[HAVING conditions]

[ORDER BY column-list [ASC | DESC]]

Most commonly this is an '**ORDER BY fieldname DESC**' statement used to arrange the data to be the most recent. Note that the default of most databases is the oldest record is first (so called Ascending order).

If you have a single timestamp, to get the most recent record, an example might be:

ORDER BY timestamp DESC

In the above example database DB2.MDB, Table1 has two columns Date and Time. To get the most recent record the Select condition would be:

ORDER BY Date DESC, Time DESC

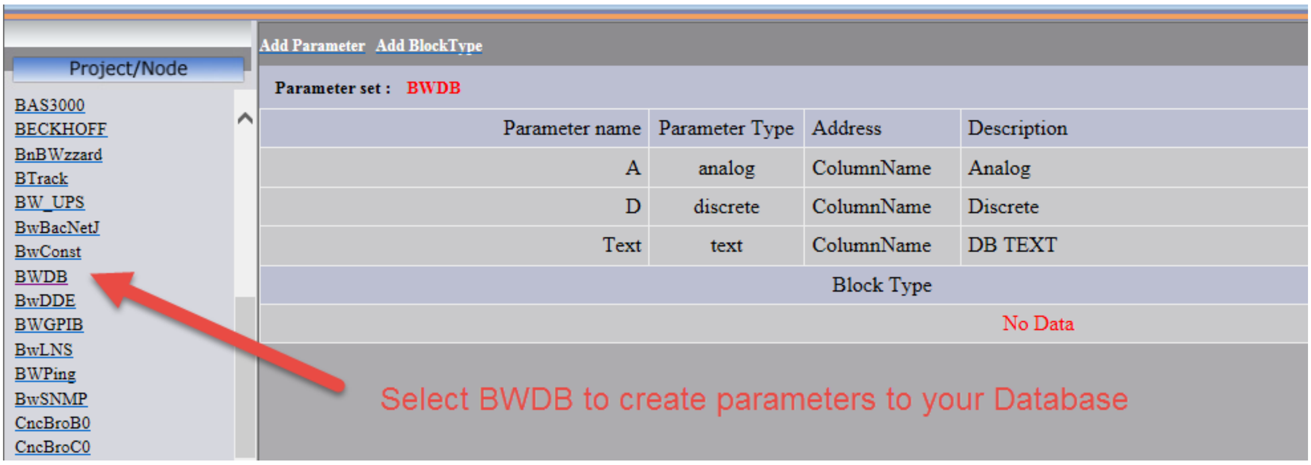
Note that Date and Time are field names (column names) in the DB2.MDB Table1. DESC is the SQL syntax for Descending (i.e. last record first).

You have to use the actual Field Names (column names) in your database. You will need a copy of the Database Software program to open the Database to see the Field Names (Column Names). For an Access Database example, see section 1.2 Example Access Database and Table.

2.4. Configure Parameters

Unlike other WebAccess drivers, there are no pre-configured Parameters or Blocks in the BWDB device driver. You must configure Parameters before configuring Tags. You must configure parameters and block types before configuring Blocks. Parameters act as a template for building tags.

20. Under **Device Type**, in the left frame, select **BWDB**.
21. The Parameter Set for the BWDB device opens



Parameter set : BWDB

Parameter name	Parameter Type	Address	Description
A	analog	ColumnName	Analog
D	discrete	ColumnName	Discrete
Text	text	ColumnName	DB TEXT

Block Type

No Data

Select BWDB to create parameters to your Database

Figure 2.4 Create the parameter in BWDB driver

22. Select **Add Parameter**

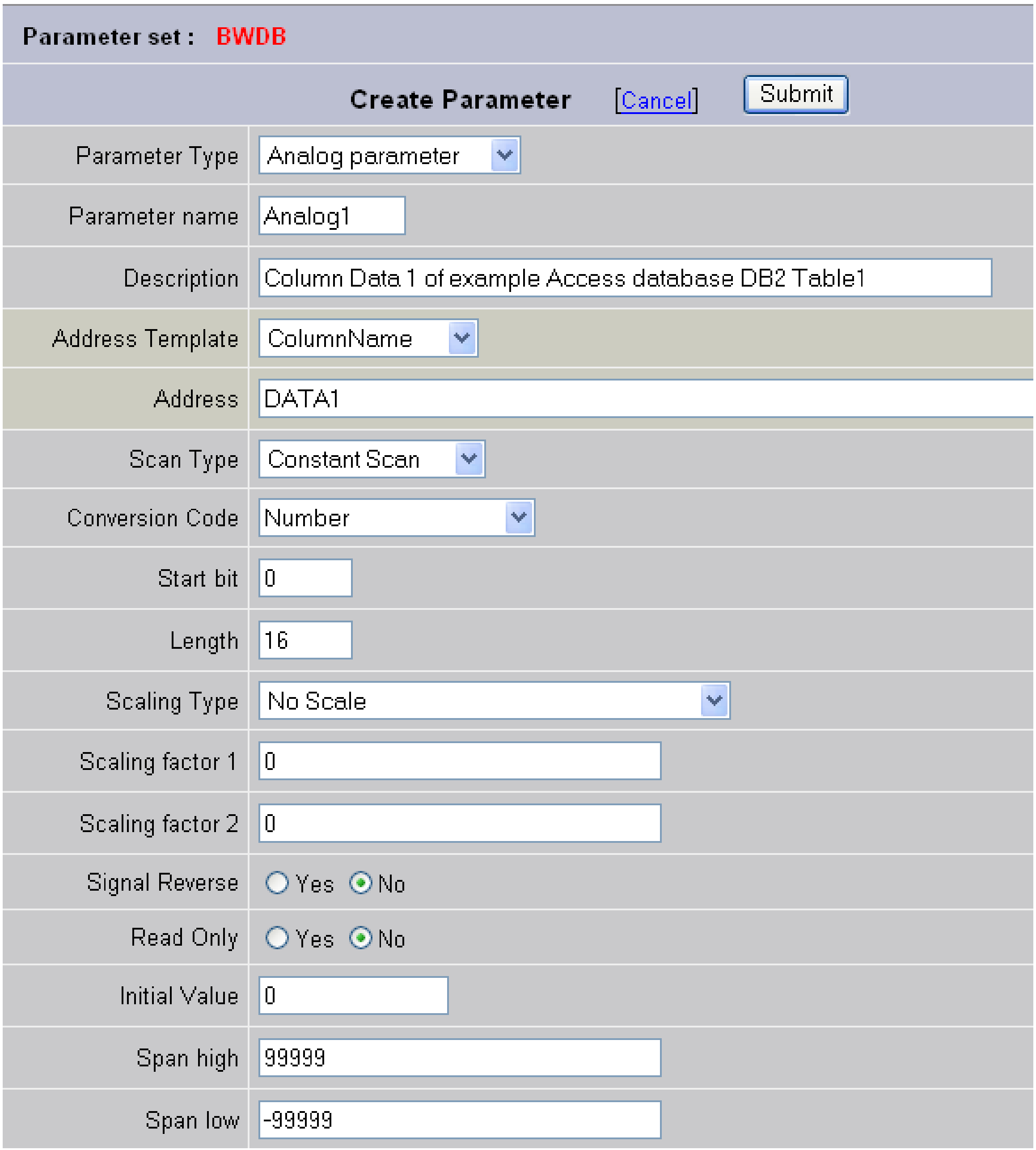
23. Select a **Parameter Type** (Analog, Discrete or Text). This is how WebAccess will treat the value read from the Database.

24. Enter a **Parameter Name**. This is how others will reference this parameter as a Template for building Tags or Block Types.

This is the minimum to create a Parameter. You press Submit now, if you are in a hurry.

You need at least one Parameter of each Data Type (Analog, Discrete and Text) if you are creating Analog, Discrete and Text type tags. You can modify everything except the Data Type later when creating a tag. You can save effort later by creating multiple Parameters with the other fields customized to match the most common entries. Also, to create Block Types, you need multiple Parameters for each Block Type.

2.4.1 Create Analog Parameter



Parameter set : BWDB

Create Parameter [Cancel] Submit

Parameter Type	Analog parameter
Parameter name	Analog1
Description	Column Data 1 of example Access database DB2 Table1
Address Template	ColumnName
Address	DATA1
Scan Type	Constant Scan
Conversion Code	Number
Start bit	0
Length	16
Scaling Type	No Scale
Scaling factor 1	0
Scaling factor 2	0
Signal Reverse	○ Yes ● No
Read Only	○ Yes ● No
Initial Value	0
Span high	99999
Span low	-99999

Figure 2.5 Create the analog parameter

25. Optionally enter a **Description**.

26. Optionally enter an **Address**. This will act as a Template and can be modified when the Tag or Block is created

The Address is the Field Name or Column name in the Database Table.

You have to use the actual Field Names (column names) in your database. You will need a copy of the Database Software program to open the Database to see the Field Names (Column Names). For an Access Database example, see section 1.2 Example Access Database and Table.

27. Optionally enter a **Scan Type**.

Constant Scan will scan at the configured Scan Rate of the Com Port.

Display Scan will read the Tag only if it appears in a Graphic Display. Real-Time Trend, Detail Display or Dialog Box(e.g. Point Info Dialog Box).

28. Enter a **Conversion Code**. For a number use **Number**.

In theory, the Microsoft DSN converts all data to Text Type, but this does not always return the correct value. You may have to experiment.

29. Optionally enter a **Scaling Type**.

No Scale uses the value as it is entered in the database.

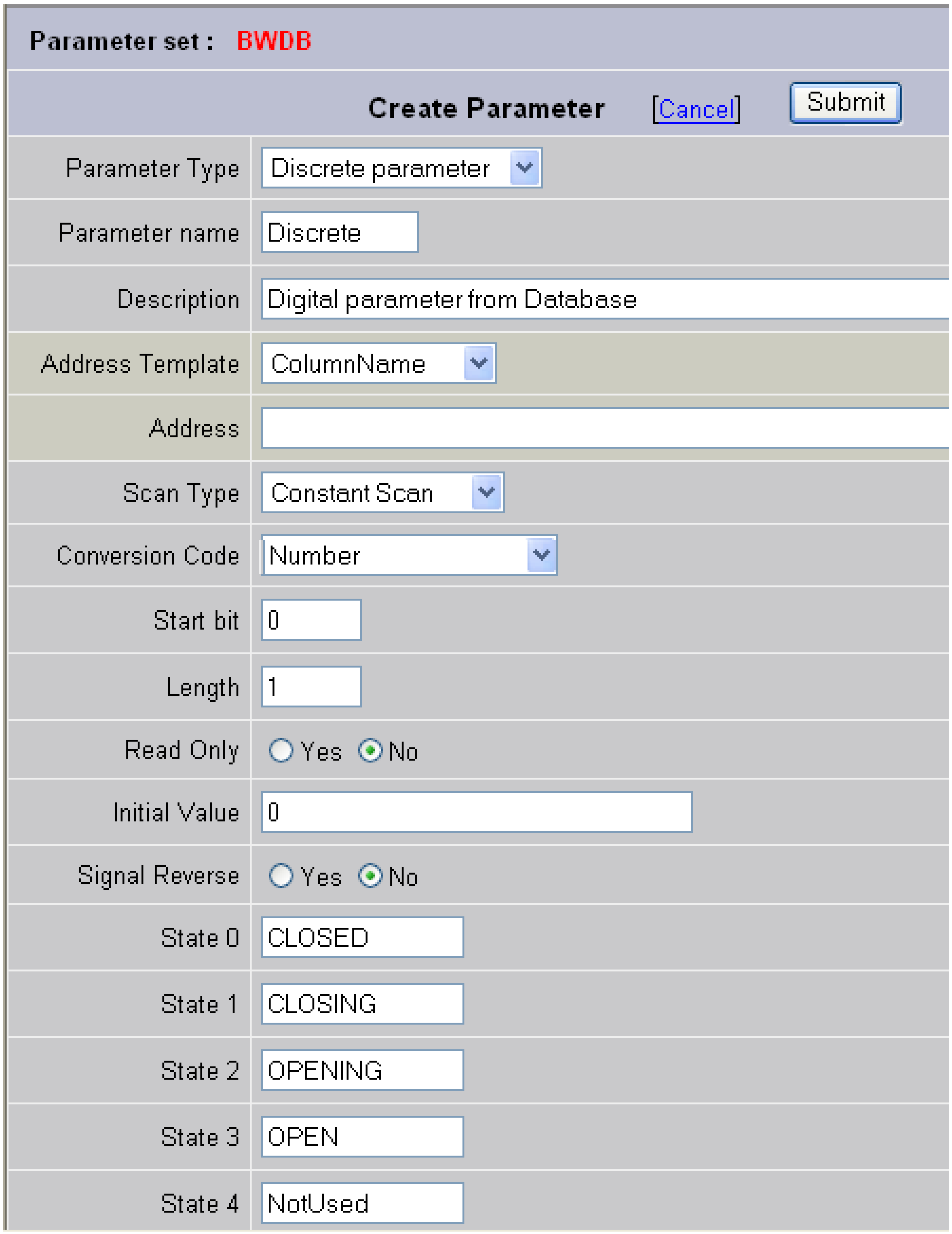
For more information see the Engineering Manual, section 4.2.13 (Click on Help in the Project Manager to open the Engineering Manual).

30. Refer to the Engineering Manual, section 5.2.4 for more information on other Analog Parameter Properties.

31. Press **Submit** to save this Analog Parameter.

Note – the BwDB device driver ignores Start Bit and Length

2.4.2 Create Discrete Parameter



Parameter set : BWDB

Create Parameter [Cancel] Submit

Parameter Type	Discrete parameter
Parameter name	Discrete
Description	Digital parameter from Database
Address Template	ColumnName
Address	
Scan Type	Constant Scan
Conversion Code	Number
Start bit	0
Length	1
Read Only	○ Yes ● No
Initial Value	0
Signal Reverse	○ Yes ● No
State 0	CLOSED
State 1	CLOSING
State 2	OPENING
State 3	OPEN
State 4	NotUsed

Figure 2.6 Create the discrete parameter

1. Select **Add Parameter** from the BWDB Device Type.
2. For the **Parameter Type**, select the **Discrete parameter**.
3. Enter a **parameter name**. This is how others will reference this parameter as a Template for building Tags or Block Types.

This is the minimum to create a Parameter. You press **Submit** now, if you are in a hurry. (Although. it is recommended to enter 'Text Type' as the Conversion Code).

4. Optionally enter a **Description**.
5. Optionally enter an **Address**. The Address is the Field Name or Column name in the Database Table.

This will act as a Template and can be modified when the Tag or Block is created

6. Optionally enter a **Scan Type**.

Constant Scan will scan at the configured Scan Rate of the Com Port.

Display Scan will read the Tag only if it appears in a Graphic Display, Real-Time Trend, Detail Display or Dialog Box (e.g. Point Info Dialog Box).

7. For the **Conversion Code**, select **Number** from the drop-down menu.

8. Optionally enter a State Descriptors for State 0, State 1, State 2, etc.

The 'NotUsed' entry disables that state. States must be continuous.

If STATE 0 and STATE1 are used, then this is a Digital Tag (e.g. 2-State, one bit).

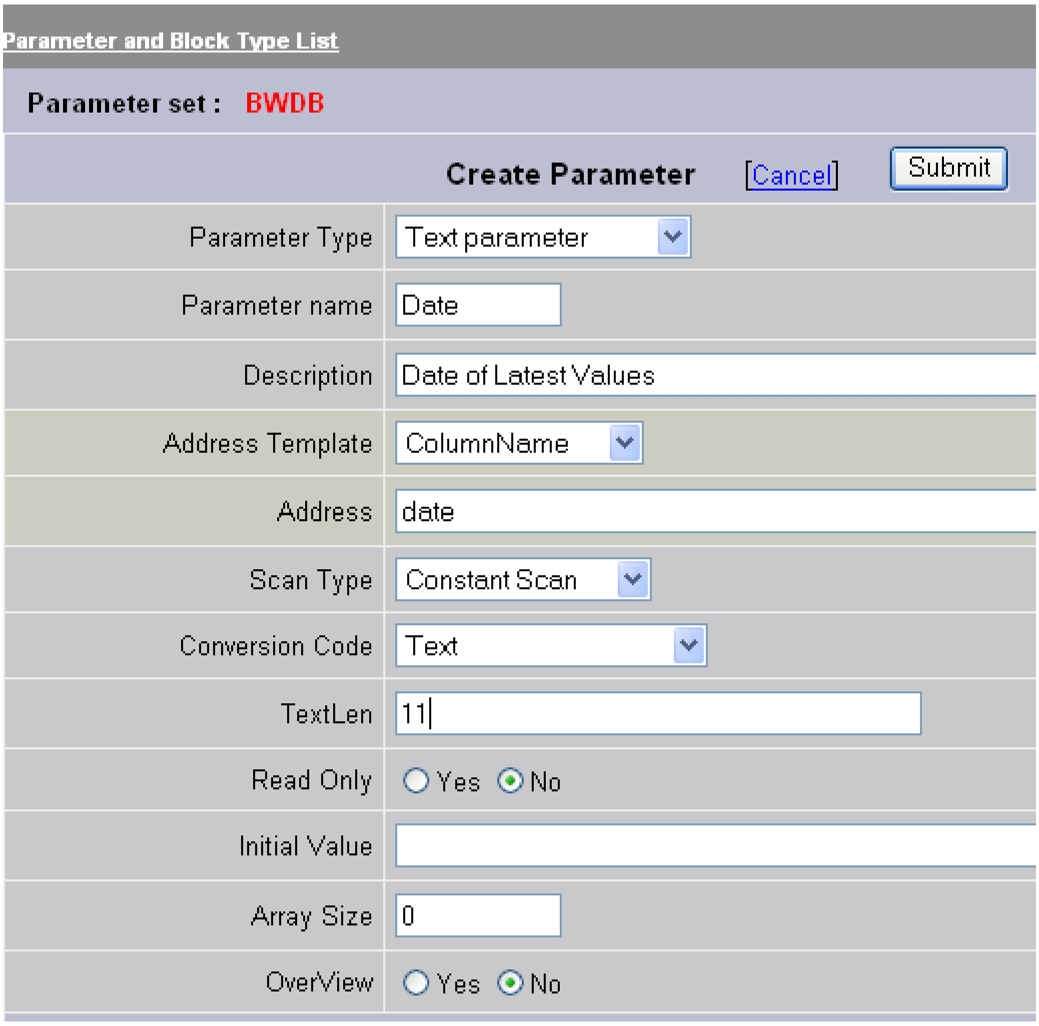
If STATE 0, STATE 1 and STATE 2 are used, then this is a 3-State Tag, (behaves as if two bits are read).

Note – the BwDB device driver ignores Start Bit and Length.

9. Press **Submit** to save the Discrete Parameter.

Refer to the Engineering Manual section 5.2.5 Discrete Parameter Properties for additional information on other fields.

2.4.3 Create Text Parameter



Parameter and Block Type List

Parameter set : BWDB

Create Parameter [Cancel] Submit

Parameter Type	Text parameter
Parameter name	Date
Description	Date of Latest Values
Address Template	ColumnName
Address	date
Scan Type	Constant Scan
Conversion Code	Text
TextLen	11
Read Only	○ Yes ● No
Initial Value	
Array Size	0
OverView	○ Yes ● No

Figure 2.7 Create the text parameter

1. Select **Add Parameter** from the BWDB Device Type.
2. For the **Parameter Type**, select **Text parameter**.
3. Enter a **parameter name**. This is how others will reference this parameter as a Template for building Tags or Block Types.

This is the minimum to create a Parameter. You press **Submit** now, if you are in a hurry. (Although. it is recommended to enter 'Text Type' as the Conversion Code).

4. Optionally enter a **Description**.
5. Optionally enter an **Address**. The Address is the Field Name or Column name in the Database Table.

This will act as a Template and can be modified when the Tag or Block is created

6. Optionally enter a **Scan Type**.

Constant Scan will scan at the configured Scan Rate of the Com Port.

Display Scan will read the Tag only if it appears in a Graphic Display, Real-Time Trend, Detail Display or Dialog Box (e.g. Point Info Dialog Box).

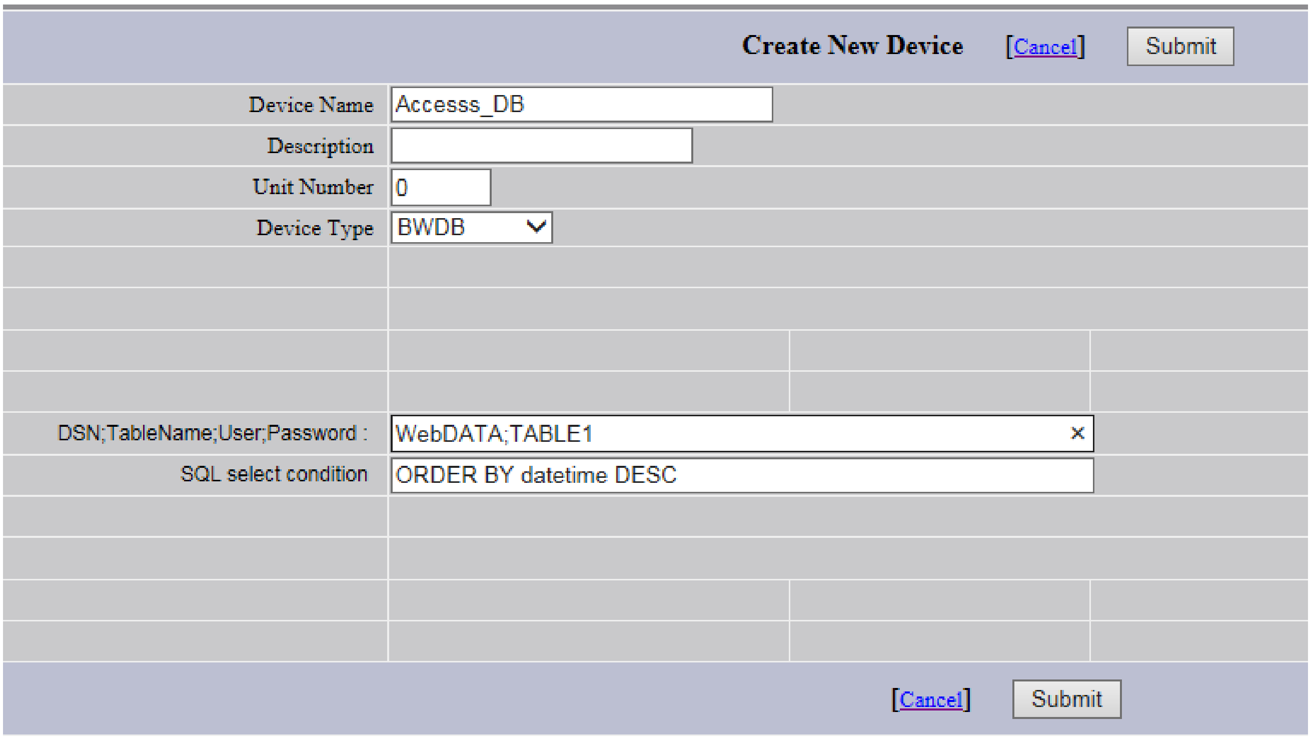
7. For the **Conversion Code**, select **Text** from the drop-down menu.

Refer to the Engineering Manual section 5.2.6 Text Parameter Properties for additional information on other fields. (Click Help at the top right in the Project Manager, to open the Engineering Manual).

8. Optionally enter a Text Length (**TextLen**). This is the number of characters the Tag will hold as its value that 70 characters maximum.
9. Press **Submit** to save the Text Parameter.

2.5. Configure a TAG

1. Use **Add Tag** to create tags from Device Properties page.



Create New Device

Device Name: Accesss_DB

Description:

Unit Number: 0

Device Type: BWDB

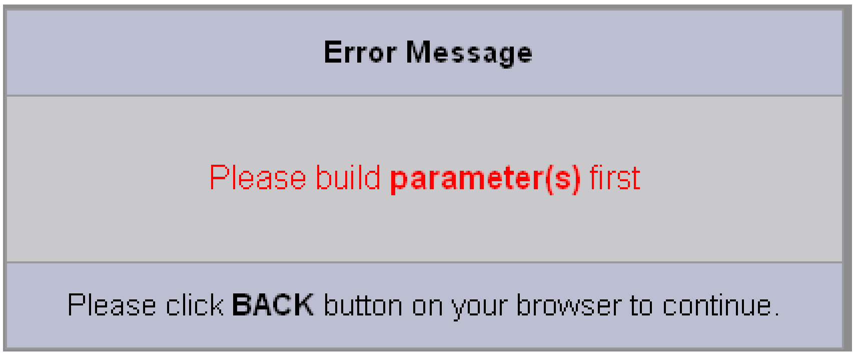
DSN;TableName;User;Password : WebDATA;TABLE1

SQL select condition: ORDER BY datetime DESC

Figure 2.8 Use Add Tag or Add Block to create Tags in BWDB device

Note – You have to create parameters and blocks before you can use Add Block.

If you have not built any parameters you will get an error message. Please refer to the previous section on building parameters,

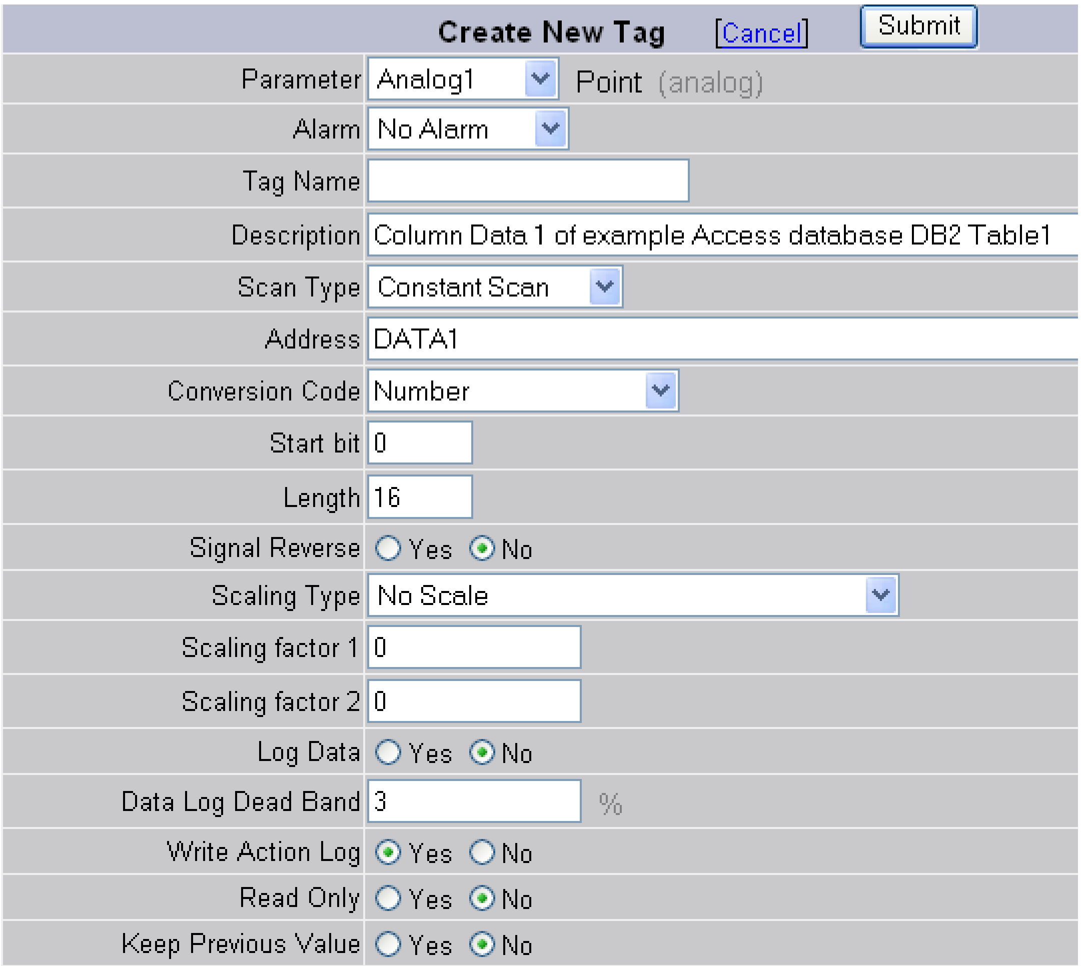


Error Message

Please build **parameter(s)** first

Please click **BACK** button on your browser to continue.

2. The **Create New Tag** page opens.



Create New Tag [Cancel] Submit

Parameter	Analog1 Point (analog)
Alarm	No Alarm
Tag Name	
Description	Column Data 1 of example Access database DB2 Table1
Scan Type	Constant Scan
Address	DATA1
Conversion Code	Number
Start bit	0
Length	16
Signal Reverse	○ Yes ● No
Scaling Type	No Scale
Scaling factor 1	0
Scaling factor 2	0
Log Data	○ Yes ● No
Data Log Dead Band	3 %
Write Action Log	● Yes ○ No
Read Only	○ Yes ● No
Keep Previous Value	○ Yes ● No

3. Select a **Parameter** to match the type of data to be read or written (from the example above, select Analog1 or Date).

4. Select ALARM to enable alarming for Analog or Discrete Tags. The page will refresh if ALARM is selected, so it is best select alarm first before entering any other data. You can always enable alarming after saving the Tag.

5. Enter a Tag Name. This is how users, graphic displays, scripts and dialog boxes will refer to the information.

6. Optionally enter or modify the **Description**.

7. Modify the **Address**. The **Address** is the Field Name or Column in the Database Table.

You have to use the actual Field Names (column names) in your database. You will need a copy of the Database Software program to open the Database to see the Field Names (Column Names). For an Access Database example, see section 1.2 Example Access Database and Table.

8. Optionally modify the **Scan Type**.

Constant Scan will scan at the configured Scan Rate of the Com Port.

Display Scan will read the Tag only if it appears in a Graphic Display, Real-Time Trend, Detail Display or Dialog Box (e.g. Point Info Dialog Box).

9. Optionally modify the **Conversion Code**.

For a number (e.g. Analog tag), use the **Number**.

For a Discrete or Digital Tag, use the **Number**.

For a Text or character data, use **Text**.

2.5.1 Analog Tag

10. **No Scale** uses the value as it is entered in the database.

For more information see the Engineering Manual, section 4.2.13 (Click on Help in the Project Manager to open the Engineering Manual).

Note – the BwDB device driver ignores Start Bit and Length.

11. For an Analog Tag, assign Alarms, Scaling, Engineering Units, Description and other features

Refer to the Engineering Manual, section 4.2 Analog Tag Properties, for more information on configuring Analog Tags.

12. Press **Submit** to save this Tag.

13. **Download** to the SCADA node.

2.5.2 Discrete Tag

1. Optionally enter a **Scan Type**.

Constant Scan will scan at the configured Scan Rate of the Com Port.

Display Scan will read the Tag only if it appears in a Graphic Display, Real-Time Trend, Detail Display or Dialog Box (e.g. Point Info Dialog Box).

2. For the **Conversion Code**, select **Number** from the drop-down menu.

3. Optionally enter a **State Descriptors** for State 0, State 1, State 2, etc.

The ‘NotUsed’ entry disables that state. States must be continuous.

If STATE 0 and STATE1 are used, then this is a Digital Tag (e.g. 2-State, one bit).

If STATE 0, STATE 1 and STATE 2 are used, then this is a 3-State Tag, (behaves as if two bits are read).

Note – the BwDB device driver ignores Start Bit and Length.

4. Press **Submit** to save the Discrete Parameter.

2.5.3 Address

The **Address** is the Field Name or Column in the Database Table.

You have to use the actual Field Names (column names) in your database. You will need a copy of the Database Software program to open the Database to see the Field Names (Column Names). For an Access Database example, see section 1.2 Example Access Database and Table.

2.5.4 Conversion Code

The conversion code is used to interpret the data as number, discrete state, or text.

For a number (e.g. Analog tag), use **Number**.

For a Discrete or Digital Tag, use **Number**.

For a Text or character data, use **Text**.

2.5.5 Start Bit

The BWDB driver ignores the Start Bit field.

2.5.6 Length

The BWDB driver ignores the Length field. The Analog and Discrete Tags always read the full number in the Database. The Discrete Tags use the number of State Descriptors that are used to determine the number of States.

2.5.7 State 0, State 1, State 2, etc.

State Descriptors for State 0, State 1, State 2, through State 7 are used to determine the number of states for a discrete tag

The ‘NotUsed’ entry disables that state. States must be continuous.

If STATE 0 and STATE1 are used, then this is a Digital Tag (e.g. 2-State, one bit).

If STATE 0, STATE 1 and STATE 2 are used, then this is a 3-State Tag, (behaves as if two bits are read).

If STATE 0 through STATE 7 are used, then this is a 7-State Tag, (behaves as if three bits are read).

Note – the BwDB device driver ignores Start Bit and Length.

2.5.8 Blocks

Currently, there are no pre-configured block types in the BWDB driver. You must configure parameters and blocks

3. Error Code

***[8000]**

Possible causes are:

- An illegal Column Name (Field Name). This usually is the case if the error occurs after adding another tag when previously there was no error. This often results in *[8000] for all tags.
- ODBC DSN miss-configured.
- Database not accessible by DSN
- DSN not on SCADA Node.
- Wrong User or Password for Database.

***[8100], *[8011], *[8012], *[8013], *[8021], *[8022], *[8023], *[8024], *[8031], *[8032], *[8033]**

Read data error (execute SQL command error)