The CSVWrite Driver

The CSVWrite driver allows a North device to export data from its Essential Data and Extra Data to delimiter-separated text files. The format of the file can be specified to select which data is included and which delimiter characters to use. This provides support to create comma-separated value files (CSV), tab-separated value files (TSV), and initialisation files (INI). Available for ObSys.

This document relates to CSVWrite driver version 2.0

Please read the *ObSys Manual* alongside this document, available from *www.northbt.com*

Contents

Purpose of CSVWrite Driver

The CSVWrite driver allows a North device to export data from its Essential Data and Extra Data to delimiter-separated text files. The format of the file can be specified to select what data is included and which delimiter characters to use. This provides support to create comma-separated value files (CSV), tab-separated value files (TSV), and initialisation files (INI).

The driver can create up to 32 different files, with a configurable update frequency.

CSVWrite can create files stored on the local PC or network path.

The Essential Data within ObSys holds 1280 values. If necessary, start the Extra Data driver (which requires an interface licence) for an additional 1024 values.

The CSVRead driver is also available, to read delimiter-separated text files.

Values

The driver creates a file with a header line, and a line per selected Essential Data and Extra Data object. Format a line to include the following values:

- Label
- Value
- Alarm state
- Units
- Date and time
- Static text

Prerequisites

The CSVWrite driver requires Essential Data v3.0 (build 01/09/2015) or later.

Determine the files and locations that the CSVWrite driver should create. Be aware that when editing or viewing a file, some software locks the file so that other applications cannot read or write it.

Detailed Operation

File Creation

CSVWrite creates a delimiter-separated text file, such as a CSV file, containing values from the North device's Essential Data and Extra Data database on the local PC or network path.

The driver triggers the creation of the file periodically, and can include a date-time stamp in the filename. Configure the creation trigger in a range from 1 minute to 7 days. Each trigger is time-aligned, for example a 30 minute triggers at :00 and :30 past each hour.

A delay can also be added to the trigger. This allows additional time for Essential Data and Extra Data to obtain the latest values. For example, if CSVWrite creates a file at midnight every day, but the database values collected are only updated at midnight, then a delay of 5 minutes could be used to allow additional time to collect the latest values.

Data Format

The driver writes an optional header line followed by a line per databse object. Set the format of these lines using the driver objects Header Line (HL) and Data Line Format (LD). Using a mixture of static text and variables, these enable a choice of delimiter-separated text files to be created, including:

- Comma-separated value files (CSV)
- Tab-separated value files (TSV)
- Initialisation files (INI)

Variables

The following variables can be included when defining the format of a header or data line:

Variable	Meaning
$(ol)	Label of current object
$(ov)	Value of current object
$(ou)	Units of current object
$(os)	Alarm State of current object (where: 0=No alarm, 1=Alarm, 2=Comms Fault)
$(oi)	Object index number
$(th)	Trigger time: hours (2-digit)
$(tm)	Trigger time: minutes (2-digit)
$(ts)	Trigger time: seconds (2-digit)
$(dd)	Trigger date (2-digit)
$(dm)	Trigger month (2-digit)
$(dy2)	Trigger year (2-digit)
$(dy4)	Trigger year (4-digit)
$(yd)	yesterday's date (2-digit)
$(ym)	yesterday's month (2-digit)
$(yy2)	yesterday's year (2-digit)
$(yy4)	yesterday's year (4-digit)
$(tb)	ASCII tab character
$(obs)	ObSys data folder
$(doc)	User's Documents folder

Example

Let's create a CSV file containing three columns – label, value, units.

First, set the **Header Line** object (HL) to '#Label,Value,Units'. This static text will be written as the first line each time the file is created.

Next, set the **Object Line Format** (LD) to '$(ol),$(ov),$(ou)'. This will write a line for each database object, replacing the variables with the label, value, and units. We have also included a comma between these fields as we are creating a comma-separated value file.

If you wanted to creating a tab-separated value file instead, then replace the commas with $(tb). If you wanted to create an INI file, then the Header Line object would contain a section title, e.g. "[Data]", and the Object Line Format could be 'O$(oi)=$(ol),$(ov)'.

Finally, let's also set the **Filename** (FN) to '$(doc)\CSVWrite-File1-$(dy4)(dm)(dd)-(th)(tm)$(ts).csv'. This will create the file in the local user's documents folder, and include a date-time stamp as part of the filename. E.g. 'CSVWrite-File1-20150515-140000.csv'.

When you first load the driver, this example should already be set in File 1.

Essential Data and Extra Data

The CSVWrite driver exports values from the North device's Essential Data, which contains 1280 values, and Extra data, which contains 1024 values. A total of 2304 values are available.

To select which database values to write to the file, use Object Start (OS), Object End (OE), and Object Increment (OI). The table below illustrates how these can be used:

Object Start	Object End	Object Increment	Actual Objects Written
1	1280	1	1..1280
1	10	1	1..10
1	10	2	1,3,5,7,9
1	100	16	1,17,33,49,65,81,97

Values from Extra Data, if available, appear immediately after Essential Data values, effectively numbered 1281 to 2304.

Example CSV File

The example below shows a comma-separated value file.

CSVWrite-File1-20150515-140000.csv

```
#Label,Value,Units
Floor 1,21.5,kWh
Floor 2,22.2,kWh
Floor 3,25.2,kWh
Floor 4,21.6,kWh
Floor 5,20.7,kWh
```

The file may have a header line, which contains labels for the subsequent rows of data. The file then contains a number of data rows, each of which contains a value from the database.

Using the Driver

On ObSys, the CSVWrite driver is pre-installed.

Once started, you will need to set up the driver before it can write data to delimiter-separated files.

Starting the Interface

⌨ To start an interface using the CSVWrite driver, follow these steps:

→ **Start Engineering** your North device using ObSys

→ Navigate to **Configuration, Interfaces,** and set an unused **Interface** to 'CSVWrite' to start the particular interface

→ Navigate to the top-level of your North device and re-scan it

The driver setup object (M*c*), labelled **CSVWrite Setup**, should now be available.

Setting up the Driver

⌨ To set up the driver, follow these steps:

→ Navigate to the **CSVWrite Setup** object (M*c*). For example, if you started interface 1 with the driver earlier, then the object reference will be 'M1'

→ Navigate to **File 1** (F1) and set the **Filename** (FN) and **Write Trigger Frequency** (WR)

→ Set the **Object Line Format** (LD) to include the data fields required, and the format of the file line

→ Set the **Object Start** (OS), **Object End** (OE) and **Object Increment** (OI) to select the range of Essential Data objects to export to the file

→ Repeat for each file

Object Specifications

Once an interface is started, one or more extra objects become available within the top-level object of the device. As with all North objects, each of these extra objects may contain sub-objects, (and each of these may contain sub-objects, and so on) - the whole object structure being a multi-layer hierarchy. It is possible to navigate around the objects using the ObSys Engineering Software.

Each object is specified below, along with its sub-objects.

Device Top-Level Objects

When an interface is started using the CSVWrite driver, the objects below become available within the top-level object of the device. For example, if interface 1 is started, then the object reference 'M1' becomes available.

Description	Reference	Type
CSVWrite Setup Set up the CSVWrite driver, started on interface c (c is the interface number)	Mc	Fixed Container: *[OSM v20\CSVWrite v20]*

CSVWrite Setup

Object Type: [OSM v20\CSVWrite v20]

The CSVWrite driver contains the following objects.

Description	Reference	Type
File *x* Configuration details for writing File *x*, where *x* is in the range 1..32	F*x*	Fixed Container: *[OSM v20\CSVWrite v20\File]*

CSVWrite File

Object Type: [OSM v20\CSVWrite v20\File]

A CSVWrite File contains the following objects.

Description	Reference	Type
Filename Full path of the file to create. This may include variables, see note below.	FN	Obj\File; Adjustable Examples: C:\CSV\Data.csv $(doc)\Exported\Meter5_$(dy4)(dm)(dd).csv
Write Trigger Frequency The rate at which the file is created. The driver synchronises the write trigger with the platform clock, i.e. a 1hour write will occur on the hour. This trigger time-stamp is used in time variables, see note below.	WR	Obj\ENum; Range: 0..13; Adjustable; Where: 0=None, 1=Once, 2=1min, 3=5min, 4=10min, 5=15min, 6=30min, 7=1hr, 8=2hr, 9=3hr, 10=6hr, 11=12hr,12=1day, 13=7days
Write Trigger Delay (min) Once the write trigger has occurred, an additional delay may be required before creating the file. This allows Essential Data time to refresh its values.	WO	Obj\Num; Range: 0..1000; Adjustable
Header Line The first line to write to the file, typically used for column header information. If blank, no line is written. This may include variables, see note below.	HL	Obj\Text; Max 127chars; Adjustable
Object Line Format Data and format of the line to write for each object specified. This may include variables, see note below.	LD	Obj\Text; Max 127chars; Adjustable
Object Start The first database object index to add to the file. Essential Data starts at 1, Extra Data starts at 1281.	OS	Obj\Num; Range: 1..2304; Adjustable
Object End The maximum database object index to add to the file. Essential Data ends at 1280, Extra Data ends at 2304.	OE	Obj\Num; Range: 1..2304; Adjustable
Object Increment The Essential Data object index to increment between adding each line to the file. Typically 1 for every object, or 16 for each page, etc.	OI	Obj\Num; Range: 1..2304; Adjustable
State	FE	Obj\Enum; Range 0..2

Description	Reference	Type
Last write state of the file		Where: 0=OK, 1=Error, 2=Off
Essential Extra Data Count The maximum number of values available for writing to the CSV file	EDC	Obj\Num; Range 0..2304

Note

The following variables can be used where indicated in the objects above:

Variable	Use
$(ol)	Label of current object
$(ov)	Value of current object
$(ou)	Units of current object
$(os)	Alarm State of current object (where: 0=No alarm, 1=Alarm, 2=Comms Fault)
$(oi)	Object index number
$(th)	Trigger time: hours (2-digit)
$(tm)	Trigger time: minutes (2-digit)
$(ts)	Trigger time: seconds (2-digit)
$(dd)	Trigger date (2-digit)
$(dm)	Trigger month (2-digit)
$(dy2)	Trigger year (2-digit)
$(dy4)	Trigger year (4-digit)
$(yd)	yesterday's date (2-digit)
$(ym)	yesterday's month (2-digit)
$(yy2)	yesterday's year (2-digit)
$(yy4)	yesterday's year (4-digit)
$(tb)	ASCII tab character
$(obs)	ObSys data folder
$(doc)	User's Documents folder

Driver Versions

Version	Build Date	Details
1.0	10/3/2015	Driver released
2.0	1/9/2015	Compatibility with ExtraData added

Next Steps…

If you require help, contact support on 01273 694422 or visit *www.northbt.com/support*

North Building Technologies Ltd
+44 (0) 1273 694422
support@northbt.com
www.northbt.com

Author: GS
Checked by: JF

Document issued 14/12/2016.