



For all eWON versions

## 1 Purpose

This document describes the usage of Export Block Descriptors. The different situations in which Export Block Descriptors (EBD) are useful will be illustrated by small examples.

### 1.1 Structure

A first part provides an overview about EBDs. The next section describes the different ways to use Export Block Descriptors. The last part provides some practical examples.

## 2 Usage

Export Block Descriptors are used when exporting data from the eWON to describe the exported content. EBDs are text strings concatenations that describe:

- What to export
- How to format the output
- How much to export

### 2.1 What to export

Different kinds of data can be exported from the eWON:

- Log files
- User files
- Configuration files
- Scheduled Actions Status
- Script expressions
- Firmware

Data type description starts with **\$dtxx**, where “xx” is the type descriptor.

#### 2.1.1 Log Files

Log files can contain:

- All or one Tag(s)'s changes of values history
- Real-time Tags values changes
- Alarms status history
- Events history
- Alarm Real-time
- Tag's Instant Values (available since eWON version 4.xx)

For log files, the amount of exported data can be specified. By this way, it is possible to export only some part of the log that is stored in the eWON. The method used to specify the period of time to be exported will be explained later. The following table describes the different data types:

EBD String	Meaning
\$dtHL	Tag value history
\$dtRL	Real-time Tag logging
\$dtAH	Alarm history
\$dtEV	Events history
\$dtAR	Alarm Real-time
\$dtIV	Instant Value (available since eWON version 4.xx)

Table 1: EBD log files data types

For Tag and alarm history, it is possible to export the data for a single or all of the Tags. The string **\$tnXXXX** is used to specify the Tag to export the values of. If the Tag's name is not specified, then all the Tags data will be exported.

## 2.1.2 User Files

User files (by opposition with system files) can be stored in the /usr directory (or in one of its sub-directories). The content of these files can be exported, as for the log files we have seen in the previous section. A user file data type is specified using the **\$dtUF** string. Of course the file name must be specified as well, this is done by using the **\$fnXXXXX** string, where "XXXXX" is the file name, relative to the /usr directory (When doing a PUTFTP, then \$fn does not need to be specified, because the PUTFTP command manages the name of the destination file).

For example, the string **\$dtUF\$fn/pierre/file1.txt** can be used to export the file /usr/pierre/file1.txt.

## 2.1.3 Configuration files

It is possible to export the eWON's configuration files. The following table describes the description strings that can be used:

EBD String	Meaning
<b>\$dtTL</b>	Exports the Tags list (one or all of the Tags)
<b>\$dtPG</b>	Exports the BASIC program
<b>\$dtCF</b>	Exports the configuration

Table 2: EBD configuration files data types

- The Tags list contains the whole configuration of one or all of the Tags. **\$tnXXXXX** can be used in this case to specify a Tag name, as described in the previous chapter. If the Tag name is not specified, then the configuration from all Tags will be exported. Please refer to the User Manual for a more detailed descriptions of Tags.
- The eWON can contain programs written in Basic language. The string **\$dtPG** can be used to export the programs in a file.
- The eWON configuration can also be exported, this time by using the **\$dtCF** string. The content of the export is described in detail in the eWON User Manual.

## 2.1.4 Status

Some actions like sending SMS, Emails, or putting a file on an FTP server, are referred as **scheduled actions** because they are put on a stack and executed one after the other. It is possible to get the status of every action that is posted on the stack by means of the **\$dtSS** string.

## 2.1.5 Script Expressions

If some variables exist in a BASIC script (string, float or integer), it is possible to get the current value of them by using the **\$dtSE** Export Block Descriptor. For example, if a BASIC program section contains a string variable **a\$**, we can know the current value of **a\$** by using **\$dtSE \$se"a\$"**. The second **\$se** (with se in lower case) must be used as the header of the variable we want to know the value of.

## 2.1.6 Firmware

It is possible (though it is scarcely used) to export the firmware the eWON is running. The **\$dtFW** string is used to achieve this.

## 2.2 Output Formatting

Following the kind of data output, it is possible to choose between one or more of the following output formats:

- **ASCII**

ASCII formats the output as comma separated values (CSV). This output format is useful because it can be directly imported by every spreadsheet program and is readable by humans. The drawback of ASCII formatting is the place, since, for example, the number 125 is coded on three bytes (three characters) instead of one.

- **Binary**

Binary formats the output as a succession of bytes. For the same amount of information, it is a lot shorter than the ASCII formatting, but it forces the user to decode the output with a specific program. The binary structure of the different exports can be found in the Technical Notes that can be downloaded from the eWON web site (*Support/Documentation/Technical Notes*).

- **HTML**

HTML formatting is the most user readable format, because it allows to present the data as tables. The price to pay for such a nice formatting is a big overhead of data transmission, due to the layout information that is contained in the output.

- **Graph**

The Graph formatting can be applied only to real-time logging data types. It outputs a PNG format image.

The style is declared in an Export Block Descriptor with the **\$ftX** field, where "X" stands for:

Parameter	Format
<b>B</b>	Binary
<b>G</b>	Graph
<b>T</b>	Text
<b>H</b>	HTML

Table 3: Output formats types

The following table shows the available formatting styles for each of the data types:

Data type	String	Available formatting
<b>Real-time Tag value logging</b>	\$dtRL	B,G,T,H
<b>Tag value history</b>	\$dtHL	B,G,T,H
<b>Alarm history</b>	\$dtAH	T,H
<b>Events</b>	\$dtEV	T,H
<b>Tag List</b>	\$dtTL	T,H
<b>Alarm Real Time logging</b>	\$dtAR	T,H
<b>Program</b>	\$dtPG	T
<b>Configuration</b>	\$dtCF	B
<b>Firmware</b>	\$dtFW	B
<b>Scheduled Actions Status</b>	\$dtSS	T,H
<b>Script Expression</b>	\$dtSE	B,T,H
<b>User File</b>	\$dtUF	B,T,H
<b>Instant Value</b>	\$dtIV	B,T

Table 4: Formatting styles available for data types

## 2.3 How Much to Export

For logging data types, the amount of data to transfer can be specified. It consists in setting the timespan during which the data must be exported. The defined time can be absolute, it can be relative to the current time, or it can be a marked time.

The starting time is set with the string **\$stXXX** for absolute time, **\$st\_XXX** for relative time (notice the underscore) and **\$stL** for last marked time (see next subsection).

The ending time is set with the string **\$etXXX** for absolute time and **\$et\_XXX** for relative time.

The time itself is specified with one or several fields, and is different for relative and absolute time reference. The string is **[sXX][mXX][hXX][dXX]** for relative time. Minutes are considered by default if not specified. For absolute time, the string is **DDMMYYYY[\_HHMMSS][\_mmm]**, where “DDMMYYYY” stands for DAY-MONTH-YEAR, “HHMMSS” is HOURS-MINUTES-SECONDS and “mmm” stands for milliseconds.

Examples:

Time range description	String
From 10 minutes ago until 5 seconds ago	\$st_m10\$et_s5
From 10 minutes ago until now	\$st_m10\$et_m0
From Wednesday, the 30 <sup>th</sup> of April 2003 at 04:03:25 PM until now	\$st30042003_160325\$et_m0
From Wednesday, the 30 <sup>th</sup> of April 2003 at 04:03:25 PM until Thursday, the first of May at 10:00:00 AM and 354 millisec.	\$st30042003_160325\$et01052003_100000_354

Table 5: Timespan definitions examples

### 2.3.1 Marking Time

It is often useful to set the output time range starting at a previously marked time. To start or stop at the previously marked time, we can use the strings **\$stL** and **\$etL**. To mark the current time, the string **\$ut** must be appended in the Export Block Descriptor.

Examples:

Time range description	String
From marked time until now	\$stL\$et_m0
From marked time until now and mark current time	\$stL\$et_m0\$ut

Table 6: Marking time examples

## 2.4 More Examples

<b>Export the history of all Tags in HTML format from 10 minutes ago until now</b>
--

\$dtHL\$ftH\$st_m10\$et_m0
----------------------------

Table 7: Historical logging export - example 1

<b>Export the history of the Tag named "Test" in binary format from 10 minutes ago until now</b>
--

\$dtHL\$tnTest\$ftH\$st_m10\$et_m0
------------------------------------

Table 8: Historical logging export - example 2

<b>Export the events in text format from last marked time until now, and mark current time</b>
--

\$dtEV\$ftT\$stL\$et_m0\$ut
-----------------------------

Table 9: Events file export

<b>Export the real-time logging of Tag "Test" in graph format starting 10 minutes and 30 seconds ago and finishing now (mark also the current time)</b>
---

\$dtRL\$tnTest\$ftG\$st_m10s30\$et_m0\$ut
---

Table 10: Real-time logging example

## 3 Different Uses of Export Block Descriptors

Export Block Descriptors can be used in different situations, such as alarm actions. When a Tag's alarm status becomes active, an Email can be sent or an FTP file can be uploaded on the configured FTP server. The content of the Email or the FTP file can be specified using EBDs. Basic scripts can also use EBDs, for example to send a report every day at midnight, or to automatically calculate and communicate the past month's phone bill.

### 3.1 Alarm actions

#### 3.1.1 Sending Emails

To send an Email on alarm, the following configuration is needed:

- **A SMTP server must be defined (Main Configuration)**
- **If needed, the Internet connection parameters must be set (Dialup configuration)**
- **The alarm must be enabled for the Tag**
- **The Email action must be enabled (see below):**

<b>Email upon</b>	<input checked="" type="checkbox"/> ALM	<input type="checkbox"/> ACK	<input type="checkbox"/> RTN	<input type="checkbox"/> END
Short Message:	<input type="checkbox"/> Format as short message			
eMail TO:	yourmail@yourmailserver.net		ex: usr1@dom.ci,usr2@dom.ci	
eMail CC:				
eMail Subject:	eWON alarm: Tag value history			
eMail Attachment(s):	[ \$dtHL\$ftT\$stL\$et_m0\$ut ]		ex: &[ \$dtEV ]	

Figure 1: Email on alarm action configuration

The recipient(s), the carbon-copy (CC) recipients if any, the Email subject and the Email content must be defined. Concerning the Email content, there are two solutions: send the output of the Export Block Descriptor inside the message, or send it as an attachment. The above shown message content sends the output inside the message. To send it as an attachment, the Export Block Descriptor string must start with an "&". The above content would then have been written: **\$[dtHL\$ftT\$stL\$et\_m0\$ut]**.

#### 3.1.2 Uploading file by FTP

The following configuration is required to upload a file to a FTP server:

- **An FTP server must be set up (Main Config)**
- **If needed, the Internet connection parameters must be set (Dialup configuration)**
- **The alarm must be enabled for the Tag**
- **The Put FTP action must be enabled**

For all eWON versions

The following figure shows the Put FTP configuration:

Put FTP upon		<input checked="" type="checkbox"/> ALM	<input type="checkbox"/> ACK	<input type="checkbox"/> RTN	<input type="checkbox"/> END
Destination File Name:	Tag_History.txt				
File Content :	[\$dtHL\$ftT\$stL\$et_m0\$ut]				

Figure 2: Put FTP alarm action configuration

The destination file name (the file name on the FTP server) and the file content must be specified. You can also use the Instant Value export format (available since version 4.xx) to export the current value from a Tag, by setting the following Export Block Descriptor in the **File Content** field from the **Put FTP upon** field in the Tag's Alarm Notification Page:

`[$dtIV$ftT$flA]`

In the above example the current value from the Group A Tags is uploaded in text format to the FTP server that is defined in the eWON (providing that the name of the export file has been entered in the **Destination File Name** field from PUT FTP on alarm config).

## 3.2 Basic Script

The BASIC interpreter provided in the eWON firmware is a powerful tool that allows to access every feature of the eWON. The provided routines allow to send Emails and upload FTP files, as well as the alarm actions. The commands described below can be entered in the **Script Setup** page or they can be included in a program. Information about Programming the eWON can be found in the eWON User Manual.

### 3.2.1 Sending Emails

The BASIC command to send Emails is **SENDMAIL**. The command takes four mandatory parameters separated by commas. It has the following syntax:

`SENDMAIL S1, S2, S3, S4`

Where:

- **S1 is the Email address(es) of the recipient(s). If several addresses are provided, they must be separated by a semi-colon.**
- **S2 is the Email address(es) of the carbon-copy recipient(s). If several addresses are provided, they must be separated by semi-colons.**
- **S3 is the message subject**
- **S4 is the message content**

Example: the following command can be given to send all Tags value history from the last marked time up to now to **user1@actl.be**:

`SENDMAIL user1@actl.be", "" , "ewon Tags history", "[$dtHL$ftT$stL$et_m0$ut]"`

### 3.2.2 Uploading Files

The BASIC command used to upload files to a FTP server is **PUTFTP**. The command takes two arguments separated by commas. It has the following syntax:

`PUTFTP S1, S2`

Where:

- **S1 is the destination file name**
- **S2 is the file content**

**Example:** To upload the previous content to the server under the name **ewonTagsHistory.txt**, the following command can be used:

`PUTFTP "ewonTagsHistory.txt", "[$dtHL$ftT$stL$et_m0$ut]"`