Ebd csv2kml converter User Guide

The ebd_CSV2KML Application – Over View

This is an application is designed to convert files downloaded from the dataTaker DT8X range of data loggers, (specifically the DT 85 in our application) that use a GPS reciever, such as a Garmin 18 to datalog information of parameters against GPS coordinates, to Keyhole Markup Language (.kml file), used with Google Earth. This enables you to log & track parameters for testing and development with the added data of where they occoured.

This can be very useful in the design & development of new products, especially vehicle development, where you need to be able to demonstrate geographically where certain conditions are exhibited. - For example, at a certain altitude, or specific sequence of hillclimbing or after a set of corners etc...

The latest installer file can be downloaded here current V1.2.0.0
To update to V1.3.0.2, download this exe, then browse to the location storing the program file & overwrite the file.

Screen shots of the application & Google Earth outputs:
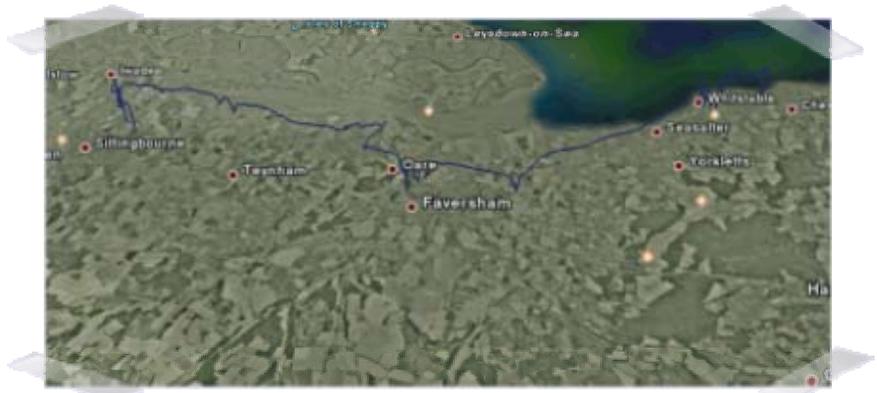
## Release Notes for ebd CSV2KML Version 1.3.0.2

Items updated:
- Filename extension of the source file is now not case sensitive.
- This is an update to the package that can still be downloaded by clicking on the logo below (Version 1.2.0.0 installer)
- dbd file (DataTaker Binary data) support added - copy the "dumpdbd.exe" application (found at the dataTaker Website here to the directory that the ebdCSV2KML.exe application is located
- Added ability to drag & drop source file onto the program icon - similar to gplot
- Fixed runtime error 5 when trying to use an invalid filetype

Usage instructions for ebd_CSV2KML Converter application (Version 1.3.0.2)

1) Ensure Excel is installed on your system, *but not running…*
   i) This can be checked by using Task manager to look for an orphaned or ghost existence of Excel. (If, in the unlikely event that this application bombs then there is a possibility that it can leave a ghost of Excel behind) - Something that should be tidied up!
2) Launch the ebd CSV2KML converter.
3) Drag & Drop either a ".csv" (Comma Separated Value) file or ".dbd" (DataTaker Binary Data) file onto any of the following:
   i) The Logo at the bottom,
   ii) The Text window for "input file":
   iii) The form background.

Note: You will have to wait briefly at this point as csv2kml application examines the status of the file. If it determines it is a multiple schedule file, then a dialogue box will appear. If there is only one schedule, you will not see this box & the parameters will be available to select from the drop down list.

4) Choose a Schedule: (If required – If not continue to select variable)
   i) Simply select the Schedule you require from the Dropdown box.
   ii) The Application will down populate the Variable dropdown box from the column headers in the source dtat csv or dbd file.
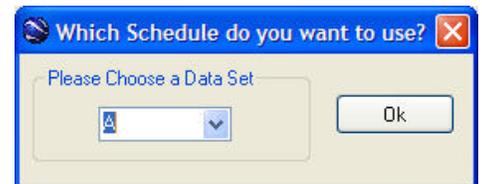
5) Select a Variable: - Now, the Variable / parameter selection box should be active:
   i) Select which parameter you wish to graph in Google Earth from the list.

6) If you want a date range, (and your file has data recorded from more than one date), then click the check box.
   i) A drop down list will appear with the possible selections you can make for date:
      a) This is information contained in the source file
   b) Select the date you require from the drop down list.
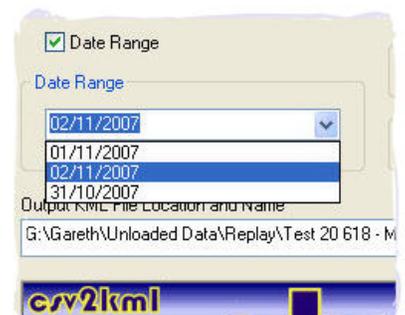
7) Now, using the controls on the right hand side of the form; select your parameters for the output Google Earth KML file (explained below)

8) Click "Generate KML" button to produce KML file(s)

Configuring Your Output File Parameters:

**1) Set up the parameters for the output kml file:**

   i) To change the colour of the line you will see:
   a) Click the "Colour picker" Button
   b) Select which colour you want,
   c) Click OK
   d) You should see the colour you selected in the box next to the "Colour Picker" Button; - this will be the colour of the line in Google Earth.
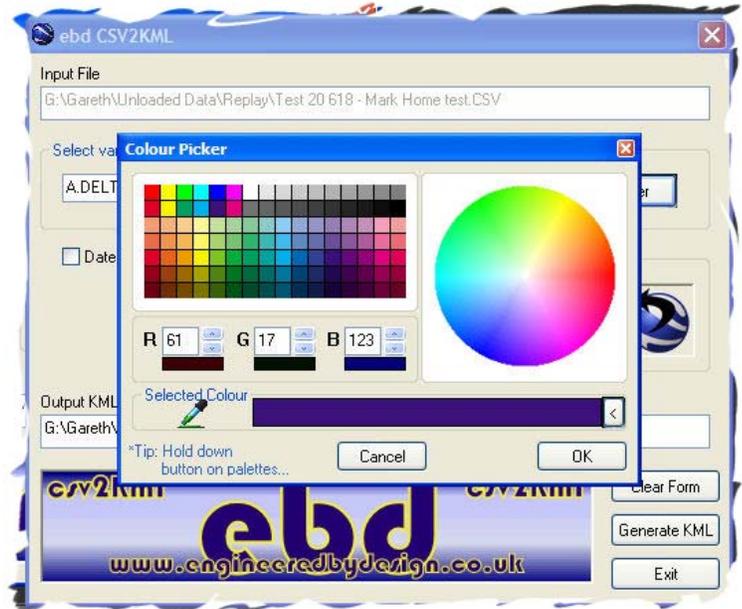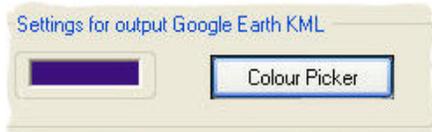
   ii) Change the Line Width:
   a) Click in the text box underneath the "line width" tag, type in the value required
   b) Note, This box cannot be empty, and will tell you so if you try & delete the value instead of overwriting it.
   c) Also, this value cannot be "alpha" such as a letter or character.
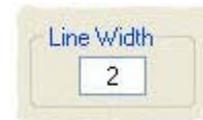
   Tip: - A line that is too thin will be difficult to see in Google Earth. A line that is too thick might mean that you cannot distinguish where the vehicle has traveled as the lines blur together. And example of this would be if the vehicle has traveled up & down the same road within the time frame you are plotting. If you are using a high enough resolution, and a pretty good GPS receiver, like the Garmin 18LVC, then you can even tell what side of the road the vehicle was on.

   iii) Scale Factor:

   Tip: - Why do I want a scale factor? I have found this particularly useful when comparing a "digital" value, ay denoting the state or position of an item, via a relay or position switch, so the data is 1 or 0, against a large analogue value. To make it more visually useful you can increase the amplitude of the effectively square wave, to one closer to the analogue value you are comparing against.
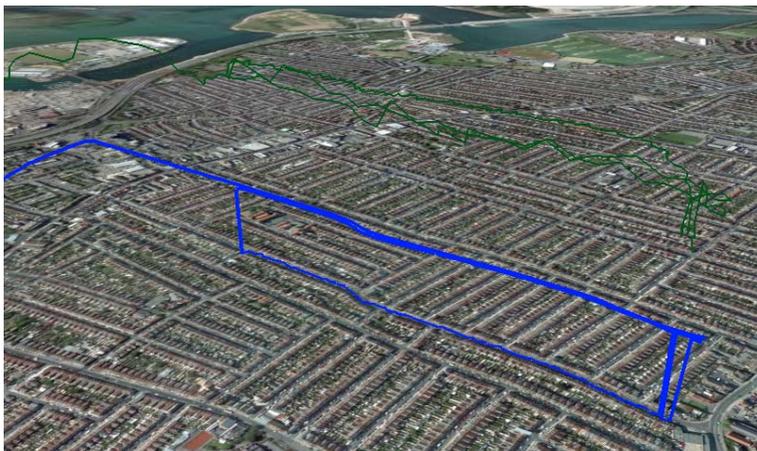
   In the picture here, the blue line represents a digital signal i.e. 1's & 0's, scaled up by a factor of 100. The Green line is a analogue value whose median is around the 200 level. This makes it easier to visually compare values.

3

a) To use, simply select the value in the text box, - your selected parameter will be multiplied by this number when the output file is generated.
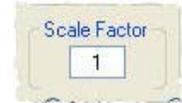
Important note: your source file is never saved, overwritten or modified – Source Data is Sacred!

b) The alternative is to scale something down to make the visual comparison easier, so to reduce by half, enter 0.5 as the scale factor.

iv) Path or Graph?

a) Select the Radio Button for which type of output you require.

Path: - This means the output will have no height or 'Z' Value, and only a path will be generated, effectively just geographical trace of where whatever you are monitoring has travelled.

Graph: - This meansthe output will be a line that follows the geographical path, (Lat / Long or X & Y Co-ordinates, but will plot a height above sea level (or Z Value) in relation to the value of the selected parameter at that particular location.

v) Output file location & Name.

a) Currently (V1.3.0.2) this cannot be changed due to the way it is generated when you click the "Generate KML" button. Thus you need to ensure that there are no "illegal characters" in the parameter name in the header of the column.

Illegal Filename Characters are: < > : * ? " / | |

(i) That is to say ensure there are no illegal filename characters in the DeLogger Program written for your dataTaker Data logger, as the header of the CSV / DBD file is taken from the Parameter Label from the Variable channel

(ii) This is set by the program you have written in DeLogger,
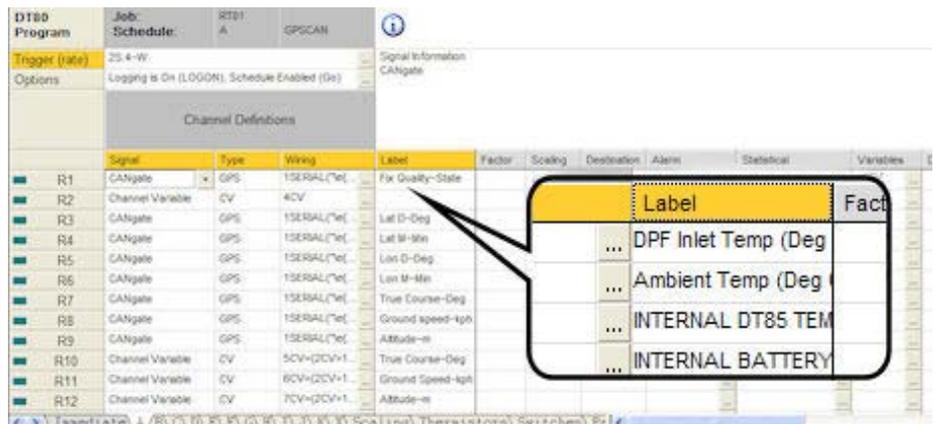
If the dataTaker Program Channel Labels already contain illegal characters. You can edit the source .csv or .dbd file to take them out after the event.f

4

b) The KML filename is currently concatenated from the source csv filename, the parameter you select for graphing & the type of graph you select, e.g. graph or path.

c) If you select max & min point options, the ebd csv2kml application will generate separate files in the same directory as the source file, with the same name as the graphed output, but with the extra extension of Max.KML or Min.kml - These files are not shown in the filename window - see below, nor are they opened automatically and the end of the KML generation process

vi) Max & Min points:

These options will create files containing push pins in the map at the coordinates that the selected conditions occurred.

a) If you wish to highlight the points on that journey where your selected parameter was at its highest & lowest, then select one, or both, of the check boxes here.
b) As described above, a file for each option selected will be created for the max & or Min points in the same directory as the source csv file.
c) These files are not opened automatically. To open:
   (i) Browse to the folder with the source csv file in it.
   (ii) Select the *max.kml or *min.kml (where * denotes the rest of your generated filename)
   (iii) Either double left click, or drag & drop into Google Earth.
   (iv) You should be flown in Google Earth to where a push-pin has been placed.
   (v) To see the data associated with that Push-Pin, left click on the pin in Google earth. A speech bubble will appear showing you the date & time it occurred, the coordinates, The Value, & The header / parameter name.

If you have any questions or comments on the installation, use, application, bugs, recommendations for improvements, support, or anything else pertaining to the use of the ebd_CSV2KML application, then I would suggest signing up to our newly opened forum, which can be found here: ebdForum.

Sample Sample DeLogger / Datataker Programme

```
     'JOB=RT01
     'COMPILED=2007/11/12 20:16:34
     'TYPE=dt80
     DT=\d
001: BEGIN"RT01"
002: CATTN
003: 'Spans and polynomial declarations
004: S1=0,1000,0.51,4.5"mBar"
005: S2=0,600,0,100"Bar"
006: S3=1,60,0,100"MPa"
007: 'Thermistor declarations
008: 'Switches declarations
009: 'Parameter declarations
010: P3=10
011: P15=2
012: P53=0
013: 'Global declarations
014: RS1S
015: ' Initialise port that CANgate is connected to
016: PS=RS232,57600,N,8,1,HWFC
017: ' Initialise the CANgate device
018: IF(1SERIAL("\e{VERBOSE OFF;VERSION^M}%f",2)<1.23)"CANgate not found or firmware version less than 1.23"
019: ' Initialise CANgate ports
020: 1SERIAL("{CONNECT 1 250;SETADDR 1 33^M}",W) 'J1939
021: 1SERIAL("{CONNECT 2 0;SETADDR 2 0^M}",W)
022: 1SERIAL("{GPSBAUD 4800^M}",W) 'Garmin GPS18 5Hz
023: ' Initialise selected CAN parameters
024: 1SERIAL("{BEGIN^M}",W)
025: 1SERIAL("{1 GPS \034GPGGA\034 6 N^M}",W)
026: 1SERIAL("{2 GPS \034GPGGA\034 2 D^M}",W)
027: 1SERIAL("{3 GPS \034GPGGA\034 2 M^M}",W)
028: 1SERIAL("{4 GPS \034GPGGA\034 4 D^M}",W)
029: 1SERIAL("{5 GPS \034GPGGA\034 4 M^M}",W)
030: 1SERIAL("{6 GPS \034GPVTG\034 1 N^M}",W)
031: 1SERIAL("{7 GPS \034GPVTG\034 7 N^M}",W)
032: 1SERIAL("{8 GPS \034GPGGA\034 9 N^M}",W)
033: 1SERIAL("{13 RECVJ 1 61444 4 5 256 3 0 FORMAT un 0.125 0 \034%%.3f\\n\034 MIN^M}",W)
034: 1SERIAL("{14 RECVJ 1 61444 3 3 256 3 0 FORMAT un 1 -125 \034%%.0f\\n\034 MAX^M}",W)
035: 1SERIAL("{12 RECVJ 1 65266 1 2 256 6 0 FORMAT un 0.05 0 \034%%.2f\\n\034 AVE^M}",W)
036: 1SERIAL("{9 RECVJ 1 65262 1 1 256 6 0 FORMAT un 1 -40 \034%%.0f\\n\034 ^M}",W)
037: 1SERIAL("{15 RECVJ 1 65262 1 1 256 6 0 FORMAT un 1 -40 \034%%.0f\\n\034 ^M}",W)
038: 1SERIAL("{11 RECVJ 1 61444 4 5 256 3 0 FORMAT un 0.125 0 \034%%.3f\\n\034 ^M}",W)
039: 1SERIAL("{END^M}",W)
040: 'schedule definition
041:
042: T(W,=8CV)
043: Function1="Start"{G}
Function2="Stop"{H}
Function3="Del Data"{deldata*}
Function4="Restart"{SINGLEPUSH}
Function5=""{}
Function6=""{}
Function7=""{}
Function8=""{}
Function9=""{}
DO{1RELAY=1}
044: 'schedule definition
045: RA"GPSCAN"("B:",ALARMS:OV:100KB,DATA:OV:16H)2S:4~W LOGONA GA
046: 1SERIAL("\e{RP 1^M}%d",1,"Fix Quality~State",FF0,=4CV)
047: ALARM(4CV<0.5000)1WARN
048: 1SERIAL("\e{RP 2^M}%f",1,"Lat D~Deg",FF0,=10CV)
```

```
049: 1SERIAL("\e{RP 3^M}%f",1,"Lat M~Min",FF6,=11CV)
050: 1SERIAL("\e{RP 4^M}%f",1,"Lon D~Deg",FF0,=12CV)
051: 1SERIAL("\e{RP 5^M}%f",1,"Lon M~Min",FF6,=13CV)
052: 1SERIAL("\e{RP 6^M}%f",1,"True Course~Deg",W,=1CV)
053: 1SERIAL("\e{RP 7^M}%f",1,"Ground speed~kph",W,=2CV)
054: 1SERIAL("\e{RP 8^M}%f",1,"Altitude~m",W,=3CV)
055: 5CV("True Course~Deg",FF1)=(2CV>1)*1CV+(2CV<=1)*5CV
056: 6CV("Ground Speed~kph",FF1)=(2CV>1)*2CV
057: 7CV("Altitude~m",FF1)=(2CV>1)*3CV+(2CV<=1)*7CV
058: 1L(S2,"Pressure F-LOOP",10)
059: 2L(S2,"Pressure R-LOOP",10)
060: 3TK("DPF Inlet Temp (Deg C)")
061: 4TK("Ambient Temp (Deg C)")
062: REFT("INTERNAL DT85 TEMP",NL)
063: ALARMR12(VBAT("INTERNAL BATTERY VOLTAGE")<5.8000/5S)12CV"Low Battery"
064: 5HV(S1,"Delta-P - DPF (mBar)")
065: 1DS("Unit Mode W-T")
066: 2DS("Nozzle Position")
067: 3DS("Fan State")
068: 4DS("Ignition STATE")
069: 1SERIAL("\e{RP 11^M}%f",2,"AVE Engine Speed (rpm)",AV)
070: 1SERIAL("\e{RP 13^M}%f",2,"MAX Engine Speed (rpm)",MX)
071: 1SERIAL("\e{RP 14^M}%f",2,"EEC1 Actual Torque ")
072: 1SERIAL("\e{RQSTJ 1 65255 1 4 256 6 0 FORMAT un 0.05 0 \034%%.2f\\n\034 ;RP^M}%f",2,"VHTotalVehHours (hrs)")
073: 1SERIAL("\e{RP 12^M}%f",2,"Engine Fuel Rate")
074: 1SERIAL("\e{RP 9^M}%f",2,"ET1EngineCoolT (deg C)")
075: 25SV("Modem Machine Satus",NL)
076: 'schedule definition
077: RB"GPS_CVS"("B:",ALARMS:OV:100KB,DATA:OV:5MB)6+CV LOGONB GB
078: T(W,=9CV)
079: 9CV("Wait Time")=9CV-8CV
080: 4CV("Fix Quality",FF0,ND)
081: 10CV("Lat D~Deg",FF0,ND)
082: 11CV("Lat M~Min",FF6,ND)
083: 12CV("Lon D~Deg",FF0,ND)
084: 13CV("Lon M~Min",FF6,ND)
085: 5CV("True Course~Deg",FF1,ND)
086: 'schedule definition
087: RC("B:",ALARMS:OV:100KB,DATA:OV:1MB)6-CV LOGOFFC GC
088: T(W,=8CV)
089: 'schedule definition
090: RD"VAR"("B:",ALARMS:OV:100KB,DATA:NOV:100KB)1D LOGOFFD HD
091: ALARM(3ST>3.0000)1RELAY"TURN OFF MODEM"{[1RELAY=0]}
092: ALARM(3ST>3.0000/1M)"TURN MODEM ON"{[1RELAY=1]}
093: 'schedule definition
094: RK("B:",ALARMS:OV:100KB,DATA:OV:1MB)8E LOGONK GK
095: ALARMR1(8DS>0.5000)"Started"{[delay(W)=4000 GA T(=8CV,W)]}
096: ALARMR2(8DS<0.5000)"Stopped"{[HA]}
097: END
     'end of program file
```

Sample Parameter Table for GPS parameters in use with Garmin GPS18 LVC Receiver in DeLogger 4

Signal	Type	Wiring	Label	Factor	Scaling	Destination	Alarms	CV
CANgate	GPS	1SERIAL("\e{RP 1^M}%d",1,"&label;")	Fix Quality~State					=4CV
Channel Variable	CV	4CV					ALARM(n$<0.5000)1WARN	
CANgate	GPS	1SERIAL("\e{RP 2^M}%f",1,"&label;")	Lat D~Deg					=10CV
CANgate	GPS	1SERIAL("\e{RP 3^M}%f",1,"&label;")	Lat M~Min					=11CV
CANgate	GPS	1SERIAL("\e{RP 4^M}%f",1,"&label;")	Lon D~Deg					=12CV
CANgate	GPS	1SERIAL("\e{RP 5^M}%f",1,"&label;")	Lon M~Min					=13CV
CANgate	GPS	1SERIAL("\e{RP 6^M}%f",1,"&label;")	True Course~Deg			W		=1CV
CANgate	GPS	1SERIAL("\e{RP 7^M}%f",1,"&label;")	Ground speed~kph			W		=2CV
CANgate	GPS	1SERIAL("\e{RP 8^M}%f",1,"&label;")	Altitude~m			W		=3CV
Channel Variable	CV	5CV=(2CV>1)*1CV+(2CV<=1)*5CV	True Course~Deg					
Channel Variable	CV	6CV=(2CV>1)*2CV	Ground Speed~kph					
Channel Variable	CV	7CV=(2CV>1)*3CV+(2CV<=1)*7CV	Altitude~m					