# GIS Python Scripts

Several scripts have been produced to take raw GPS data stored in Excel, convert to ESRI shapefile and process in a number of ways. This includes producing edges, block movement poly-lines, calculating movement distance and direction and manipulating/exporting the attribute tables.

The scripts are as follows:

|  |  |  |
| --- | --- | --- |
| ID | Script Name | Purpose |
| 1 | CreateEdgePointSHPFromExcel.pyc | Main script used to convert Excel GPS data |
| 2 | CreateMultiPointPolyFrom1PointSHP.pyc | Convert point data from (1) to poly-line |
| 3 | CreatePolyFrom2PointSHPs.pyc | Convert point data from two poly-lines created using (1) to block movement |
| 4 | Arc\_BearingCalc\_Glamorgan\_v2.pyc | Add movement distance and bearing to output shapefile from (4) |
| 5 | Arc\_BearingCalc\_SingleSHP.py | As above but for a single polyline shapefile e.g. layer edge |
| 6 | ExportAttTableAndShapeAsCSV.py | Export point shapefile attribute data including X,Y(,Z,M?) to .CSV file |
| 7 | ExportAttTableAndShapeAsCSV\_v2.py | As above but can handle polylines … work-in-progress/experimental |
| 8 | CheckSHPDetails.py | Quickly check info on spatial reference in use |
| 9 | CreatePolygonFromPolylineSHP.py | Create a block island polygon etc |

OLD Scripts:

CompilePythonScript.py

Arc\_BearingCalc\_Glamorgan.py

Arc\_BearingCalc\_Glamorgan\_v2.py

## Bugs/Gotchas:

* Only written and tested for Excel 2003 … does not accept Office 2007 .xslx file format.
* Any script reading data from Excel spreadsheets … e.g. create from Excel or reading block size file. Due to the way the script uses Excel it minimises it and when finished processing will close without asking. If you have any other file open in Excel it is also closed. Save all work in Excel and close files before using scripts*. It might be possible (in future) for the scripts to read data from Excel in a different way without this problem, but this workaround should be used for foreseeable future.*
* Input columns cannot used calculated data … there should not be any formulas in the source Excel file.
* CreateEdgePointSHPFromExcel - Limitations converting Excel to shapefile
  + Shapefiles do not allow field names longer than 10 characters.
  + Use underscores in column headers rather than spaces.
  + Do not use ‘Date’ as a column heading … most scripts now will replace with ‘Date\_’
  + Source data should be in individual worksheets by date as script cannot work with single date in worksheet containing measurements made on multiple dates … *future update to script could prescan file for all dates and prompt user to select required.*
  + Shapefiles cannot store date time … Excel ‘Date\_’ columns will be stored as string.
* CreatePolyFrom2PointSHPs … the block names in both Excel input workbooks/sheets must match exactly and exist in both files. There is an additional problem when blocks have split … earlier files do not contain the subsequent numbering scheme. Where a block does break and you want to show the positions of the block whilst intact and the distance/direction travelled for the broken pieces, then the following ‘fudge factor’ can be applied:
  + For the intact block, leave input data as normal. The last date recorded = last intact date surveyed. Notes need to be added to the input files showing the last intact date.
  + For the broken pieces, duplicate the intact block data for dates it was intact (there will be one ‘location’ for all broken pieces, hence a bit of a fudge), numbering them as per the broken pieces (e.g. 5.1, 5.2, 5.3) for all dates. This will allow the scripts to be run for each of the broken pieces, starting from the last intact measurement.
  + *Longer-term, the script could be improved by use of ‘ancestry’ linkages between intact full blocks and then subsequent bits it breaks into. This can also work for intact blocks, and their individual detachment location (or centroid point).*
* Arc\_BearingCalc … the block names in the converted shape files must exactly match an entry in the block size Excel worksheet otherwise script cannot work out whether has moved relative to A axis.