CERC

Post Processor User Guide

Version 1.1

Contents

1 Introduction

The PostProcessor is a tool for processing the ADMS Comprehensive Output File, allowing for spatial and temporal processing of the netCDF file output by MAQS and the ADMS suite of models.

2 How to run

The PostProcessor utility is run from the command line or a batch file using the following syntax:

```
<Executable file path>    <Full path to main input file>
```

Where `<Executable file path>` is the full path to the utility executable (*.out* for Linux, *.exe* for Windows), followed by the full path to the main input text file that contains the parameters for the run.

For example, if the executable file is saved in the default MAQS installation directory on Linux, `/home/user/maqs/utils`, and the input text file is saved as `/home/user/input.txt`, the command line to run the utility on Linux would be:

```
/home/user/maqs/utils/PostProcessor.out /home/user/input.txt
```

3 Main input file format

The first line of the main input file should contain the following version string:

```
POSTPROCVERSION1
```

This should be followed by a set of sections, each starting with a particular keyword (see Table 1). It is advisable (though not necessary) to add spaces between each section for readability.

Some sections are required regardless of the run type, other sections are further required for a particular run type, and other sections are optional. The following table lists the available sections

and indicates which are required/optional for each run type. Each section is then described in detail in § 3.1 and § 3.2.

Section	Required (R) / Optional (O) / Not applicable (NA), for each run type			
	RAW	INTERP	GRIDAVE	POLYAVE
RUNTYPE	R	R	R	R
NCFILE	R	R	R	R
OUTPUTPOINTTYPES	R	R	R	R
HOURLY PERIODAVERAGE	R (one or the other)	R (one or the other)	R (one or the other)	R (one or the other)
NUMGRIDLINES	NA	R	NA	NA
GRIDCELLSIZE	NA	NA	R	NA
POLYGON	NA	NA	NA	R
OUTPUTDIRANDSTEM	O	O	O	O
PLOTFILETYPE	NA	O	O	NA
PLOTLIMITS	O	O	O	NA
ZLEVEL	NA	O	O	O
MINOVERLAP	NA	O	O	O
RECALCULATEWEIGHTS	NA	O	NA	NA
OUTPUTXYCOLS	NA	O	O	NA
OUTPUTNODATA	O	O	O	NA

Table 1 Summary of sections in the main input file

3.1 Required sections

- **RUNTYPE**

  This section is used to specify what type of processing is to be done. The format of this section is given below:

  ```
  RUNTYPE
  <Run type identifier>
  ```

  Where `<Run type identifier>` is one of the following:

  - `RAW` – Use this option to generate ADMS-style concentration output files (.gst/.glt/.pst/.plt/.levels.gst/.levels.glt) containing raw or processed concentrations/statistics at (all or a subset of) the output point locations in the input netCDF file.
  - `INTERP` – Use this option to further interpolate these concentrations/statistics onto a regular grid of output points (using the natural neighbour interpolation method) for subsequent contour plotting.
  - `GRIDAVE` – Use this option to calculate horizontal spatial averages of the raw concentrations/statistics within a set of regular grid cells, e.g., for subsequent raster plotting.
  - `POLYAVE` – Use this option to calculate horizontal spatial averages of the raw concentrations/statistics within a specified polygon, e.g. to calculate ward-level average concentrations.

- **NCFILE**

This section is used to specify the netCDF file from MAQS that contains the concentration data to be processed. The format of this section is given below:

```
NCFILE
PATH = "<Full path to input netCDF file>"
```

- **OUTPUTPOINTTYPES**
  This section is used to specify which output point types are to be included in the processing. The format of this section is given below:

```
OUTPUTPOINTTYPES
<Y/N flag for specified points>
<Y/N flag for regular grid points>
<Y/N flag for source-oriented grid points>
```

  Note that source-oriented grid points can only be selected if regular grid points are also selected.

One (and only one) of the following sections is also required:

- **HOURLY**
  This section is used to indicate that datasets containing hourly concentrations (named Dataset1, Dataset2, etc. in the input netCDF file) are to be processed, as well as what processing is to be done. The format of this section is given below:

```
HOURLY
<Y/N flag indicating whether to specify the period to be processed>
If above flag is set to Y:
StartYear, StartMonth, StartDay, StartHour
EndYear, EndMonth, EndDay, EndHour
Followed by
PATH = "<Full path to input .pop file>"
<Y/N flag indicating whether to use temporal subsets>
If above flag is set to Y:
PATH = "<Full path to input .sub file>"
```

  Where:

  - `StartYear`, `StartMonth`, `StartDay`, `StartHour` specifies the start time of the first hour and `EndYear`, `EndMonth`, `EndDay`, `EndHour` the end time of the last hour to be processed. So, for example, to process the first full day of 2022, the start time should be set to 2022, 1, 1, 0 and the end time should be set to 2022, 1, 1, 24. `StartHour` is therefore constrained to be between 0-23 and `EndHour` between 1-24. If the start and date times are not specified, i.e. the preceding flag is set to `N`, the whole modelling period will be processed.
  - The *.pop* file contains details of what pollutants are to be processed and what processing is to be done, e.g. the averaging time(s), whether to calculate maximum daily outputs, percentiles, exceedences, etc. The format of the *.pop* file is described further in §4.1.
  - The *.sub* file details which hours/days/months are to be included when using temporal subsets to process the concentrations. The format of the *.sub* file is described further in §4.2.

- **PERIODAVERAGE**
  This section is used to indicate that datasets containing period average concentrations (named Ave_Dataset1, Ave_Dataset2, etc. in the input netCDF file) are to be processed. The format of this section is given below:

```
PERIODAVERAGE
<Number of pollutants to process>
<Name of pollutant 1>
<Name of pollutant 2>
…
```

  The pollutant names need to match exactly with those in the input netCDF file.

The following section is required for an `INTERP` run:

- **NUMGRIDLINES**
  This section is used to specify the number of regular grid points in X and Y onto which the data is interpolated. The format of this section is given below:

```
NUMGRIDLINES
<Number of grid points in X>
<Number of grid points in Y>
```

  The number of grid points in X and Y must be between 2 at 20,000.

The following section is required for a `GRIDAVE` run:

- **GRIDCELLSIZE**
  This section is used to specify the size of the grid cells, in X and Y, within which the spatial averages are calculated. The format of this section is given below:

```
GRIDCELLSIZE
<Grid cell size in X>
<Grid cell size in Y>
```

  The grid cell size in X and Y must be between 10 and 100,000 m.

The following section is required for a `POLYAVE` run:

- **POLYGON**
  This section is used to specify the path to the file containing the definition of the polygon within which the spatial averages are calculated. The format of this section is given below:

```
POLYGON
PATH = "<Full path to polygon file>"
```

  The format of the polygon file is described further in §4.3. The supplied polygon must be simple, i.e. it must not intersect itself or contain any holes.

3.2 Optional sections

- **OUTPUTDIRANDSTEM**
  This section can be used to specify the directory and filename stem of the output files created by the PostProcessor. The format of this section is given below:

```
OUTPUTDIRANDSTEM
/An/Example/Linux/Directory/
MyStem
```

  The directory must already exist and must be specified in full (i.e. no relative paths). For `GRIDAVE` and `POLYAVE` runs, the filename stem will be followed by additional text that describes the contents of each output file, e.g. *MyStem_Conc-NOx.grd*.
  If this section is not present, any output files will be saved to the directory containing the main input file and the stem of the input netCDF file will be used as the filename stem.

- **PLOTFILETYPE**
  This section is only applicable to `INTERP` and `GRIDAVE` runs. It can be used to select whether to output the interpolated/grid-averaged data to *.grd* files (appropriate for plotting in the third-party mapping software package Surfer) or *.csv* files. The format of this section is given below:

```
PLOTFILETYPE
<File type identifier>
```

  Where `<File type identifier>` is one of:
  - `1` – Use this option to output to *.grd* files
  - `2` – Use this option to output to *.csv* files

  If this section is not present, *.grd* files will be created for `INTERP` runs and *.csv* files for `GRIDAVE` runs.
- **PLOTLIMITS**
  This section is only applicable to `RAW`, `INTERP` and `GRIDAVE` runs. It can be used to include only points located within a region to ADMS-style concentration output files in `RAW` mode; or specify the limits of the interpolation grid for `INTERP` runs; or the region to be grid-averaged for `GRIDAVE` runs. The format of this section is given below:

```
PLOTLIMITS
Xmin
Xmax
Ymin
Ymax
```

  For `GRIDAVE` runs, `Xmax` and `Ymax` will automatically be extended so that the region includes an integer number grid cells in X and Y, if necessary.
  If this section is not present, the X and Y limits of the set of points in the netCDF file being processed will be used (again, extending `Xmax` and `Ymax` for `GRIDAVE` runs if necessary).
- **ZLEVEL**
  This section is only applicable to `INTERP`, `GRIDAVE` and `POLYAVE` runs when regular grid points are to be processed. It can be used to specify the vertical index of the regular grid of output points (in the input netCDF file) that is to be extracted for processing. The format of this section is given below:

```
ZLEVEL
<Vertical index>
```

  If the vertical index is greater than 1, only regular grid points will be processed (i.e. specified points and source-oriented grid points are always treated as though they belong to the lowest grid level).
  If this section is not present, the lowest grid level will be processed.
- **MINOVERLAP**
  This section is only applicable to `INTERP`, `GRIDAVE` and `POLYAVE` runs when processing period average datasets. It can be used to specify the minimum fractional area overlap that the Voronoi cell associated with an interpolation point or grid cell or polygon must have with regions of valid data in order that the interpolated/averaged value is calculated rather than set to missing data in the output file. Period average datasets can contain -999 values, e.g., over the sea when using annual average (Defra background map) data instead of regional air quality model data. The format of this section is given below:

```
MINOVERLAP
<Minimum overlap fraction>
```

  The minimum overlap fraction must be between 0.0001 and 1. For `POLYAVE` runs, the model will stop with an error if the polygon does not overlap with any valid data but will

otherwise always output the average, issuing a warning if the fractional overlap falls below the minimum overlap fraction.
If this section is not present, <`Minimum overlap fraction`> will be taken as 0.1.

- **RECALCULATEWEIGHTS**
  This section is only applicable to `INTERP` runs. It can be used to reduce the memory requirements of the PostProcessor utility if running into memory issues, at the expense of increasing the runtime. Memory issues are most likely to occur when the number of interpolation grid points (specified via the `NUMGRIDLINES` section) is large. Memory issues can present themselves as allocation errors (most likely to occur on Windows) or as runs lagging or being killed due to the use of swap memory (most likely to occur on Linux). The format of this section is given below:
  ```
  RECALCULATEWEIGHTS
  <Y/N>
  ```
  Setting the flag to `Y` will mean that the interpolation weights are recalculated each time a new plot file is generated, which uses a lot less memory (since the weights do not have to be stored) but can increase runtimes significantly.
  If this section is not present, the interpolation weights will be calculated only once (memory intensive but faster runtimes).
- **OUTPUTXYCOLS**
  This section is only applicable to `INTERP` and `GRIDAVE` runs when outputting to *.csv* files. It can be used to specify whether every output *.csv* file should include X and Y columns, or just the first output file created. This option can be used to save on disk space, since all output files created by a single run of the PostProcessor in either of these modes will contain identical X and Y columns. The format of this section is given below:
  ```
  OUTPUTXYCOLS
  <Y/N flag indicating whether to output X,Y columns to all .csv files>
  ```
  If this section is not present, X and Y columns will be output to all *.csv* files.
- **OUTPUTNODATA**
  This section is only applicable to `RAW`, `INTERP` and `GRIDAVE` runs when processing period average datasets and not outputting to *.grd* files. It can be used to specify whether missing data values should be included in the output files or not. The format of this section is given below:
  ```
  OUTPUTNODATA
  <Y/N>
  ```
  If this section is not present, missing data will not be output for `RAW` runs but will be output for `INTERP` or `GRIDAVE` runs.

4 Secondary input file formats

4.1 *.pop* file

The *.pop* file follows the same format as files generated using the **Save**... button in the **Create Output Files** screen of the ADMS Comprehensive Output File Processor (COFP). The first four lines of the *.pop* file should contain the following:

```
POPFORMAT3
"Name"
"Description"
<Number of outputs>
```

The first line is the version string. The next two lines are ignored by the PostProcessor but must be present. The fourth line should contain the number of outputs, which can be between 1 and 30.

The next line must be blank.

A block for each output should then follow, with a single blank line separating each block. The format of each block is given below, with limits/valid identifiers given in brackets:

```
Pollutant name
Include flag (0/1)
Averaging time value (1-99,999)
Averaging time unit identifier (0-6)
Extra condition identifier (0-4)
Temporal subset type identifier (0-3)
Temporal subset number (1-9)
Averages flag (0/1)
Number of percentiles (0-20)
If number of percentiles > 0:
Percentile 1 (0-100 %)
Percentile 2 (0-100 %)
…
Followed by
Number of exceedences (0-20)
If number of exceedences > 0:
Exceedence 1 (0-10,000,000 ug/m3)
Exceedence 2 (0-10,000,000 ug/m3)
…
Followed by
Exceedence unit identifier (0-2)
Output type (0)
Units (ug/m3)
Validity threshold (0-100 %)
```

Notes:

- Pollutant name – This must exactly match one of the pollutant names in the netCDF file
- Include flag – Set to 0 to indicate that output for this block is not required, or 1 to indicate that it is
- Averaging time value – This can range between 1 and 99,999, depending on the specified averaging time unit
- Averaging time unit identifier – This should be set to one of the following:
    - 0 – Period
    - 1 – Year
    - 2 – Month
    - 3 – Day
    - 4 – Hour
    - 5 – Hour rolling
    - 6 – Peak season

  Refer to Section 2.4.3 of the COFP v4 User Guide (available here) for details about averaging time unit identifiers 0-5. Averaging time unit identifier 6 (Peak season) is specific to the MAQS PostProcessor. It can be used to compare modelled ozone concentrations against the World Health Organisation's "peak season" air quality guideline for ozone. This guideline states that the average of daily maximum 8-hour rolling mean ozone concentration in the six consecutive calendar months with the highest six-month running-average ozone concentration should not exceed 60 $\mu g/m^3$. For peak season outputs, the averaging time value must be set to 1, while the extra condition identifier, the number of percentiles and the number of exceedences must all be set to 0.
- Extra condition identifier – This should be set to one of the following:
    - 0 – None
    - 1 – Maximum daily
    - 2 – Subset average
    - 3 – Subset minimum
    - 4 – Subset maximum

  Refer to Section 2.4.3 of the COFP User Guide for details
- Temporal subset type identifier – This must always be present but will only be used if the extra condition identifier is set to 2, 3 or 4. It should be set to one of the following:
    - 0 – None
    - 1 – Daily subset type
    - 2 – Weekly subset type
    - 3 – Yearly subset type

  Refer to Section 2.4.3 of the COFP User Guide for details
- Temporal subset number – This must always be present but will only be used if the extra condition identifier is set to 2, 3 or 4. Nine different subsets of each subset type (Daily/Weekly/Yearly) can be defined in the *.sub* file (see §4.2 for format details of this file). Use digits 1-9 to select which subset to use in this case.
- Averages flag – Set to 1 to indicate that the results of processing individual averaging time blocks are to be output, or 0 to indicate that they are only to be used for the purposes of calculating percentiles/exceedences. This flag must be set to 1 if the averaging time unit is set to Period, since in this case there is only one averaging time block.
- Number of percentiles – Up to 20 percentiles can be specified
- Number of exceedences – Up to 20 exceedences can be specified

- Exceedence unit identifier – This must always be present but will only be used if the number of exceedences is greater than zero. It should be set to one of the following:
  - 0 – Annualised
  - 1 – Percent
  - 2 – Count

  Refer to Section 2.4.3 of the COFP User Guide for details. The default in the COFP is Annualised.
- Units – This must always be set to `ug/m3`
- Validity threshold – Use this to set the percentage of values that must be valid before any average is valid. Refer to Section 2.4.3 of the COFP User Guide for further details. The default in the COFP is 75 %.

4.2 *.sub* file

The *.sub* file follows the same format as files generated using the **Save**... button in the **Define Temporal Subsets** screen of the COFP. An example *.sub* file is shown below, followed by a description of its contents:

```
FileVersion1
DailySubset,Description,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,…
Daily1,Odds,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0
Daily2,Evens,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1
Daily3,Working day,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0
Daily4,,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
Daily5,,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
Daily6,,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
Daily7,,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
Daily8,,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
Daily9,,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
WeeklySubset,Description,DailySubsetUsed,Monday,Tuesday,Wednesday,Thursday,Friday,…
Weekly1,Working week,Daily3,1,1,1,1,1,0,0
Weekly2,Weekend,[All hours],0,0,0,0,0,1,1
Weekly3,,[All hours],1,1,1,1,1,1,1
Weekly4,,[All hours],1,1,1,1,1,1,1
Weekly5,,[All hours],1,1,1,1,1,1,1
Weekly6,,[All hours],1,1,1,1,1,1,1
Weekly7,,[All hours],1,1,1,1,1,1,1
Weekly8,,[All hours],1,1,1,1,1,1,1
Weekly9,,[All hours],1,1,1,1,1,1,1
YearlySubset,Description,WeeklySubsetUsed,January,February,March,April,May,June,…
Yearly1,Working year,Weekly1,1,1,1,1,1,1,0,1,1,1,1,0
Yearly2,Summer,[All days],0,0,0,0,0,1,1,1,0,0,0,0
Yearly3,,[All days],1,1,1,1,1,1,1,1,1,1,1,1
Yearly4,,[All days],1,1,1,1,1,1,1,1,1,1,1,1
Yearly5,,[All days],1,1,1,1,1,1,1,1,1,1,1,1
Yearly6,,[All days],1,1,1,1,1,1,1,1,1,1,1,1
Yearly7,,[All days],1,1,1,1,1,1,1,1,1,1,1,1
Yearly8,,[All days],1,1,1,1,1,1,1,1,1,1,1,1
Yearly9,,[All days],1,1,1,1,1,1,1,1,1,1,1,1
```

Notes:

- The first line is the version string
- The second line is the daily subset header line (comma separated, as with all subsequent lines), which should read exactly as shown above (replacing the '…' with the remaining hours of the day up to 24)
- The next nine lines contain the daily subsets. There must be nine daily subsets even if fewer than this number will actually be used. The first column contains the daily subset identifier,

which must be set to Daily1, Daily2, …, Daily9 (in ascending order). The next column can be used to give each subset a description. The next 24 columns are used to define which hours of the day are to be included in that daily subset, using 0 (excluded) or 1 (included). So, for example, the Daily3 subset above with the description "Working day" only includes hours 10 through to 17 (9:00am – 5:00pm). Note that an hour-ending convention is employed, so hour 10 refers to the hour 9:00-10:00 am.

- The next line (line 12) is the weekly subset header line, which should read exactly as shown above (replacing the '…' with the remaining days of the week up to Sunday)
- The next nine lines contain the weekly subsets. Again, there must always be nine of these. The first column contains the weekly subset identifier, which must be set to Weekly1, Weekly2, …, Weekly9 (in ascending order). The next column can be used to give each subset a description. The next column is used to reference a daily subset identifier, or can be set to [All hours] to include all hours of each included day in that weekly subset. The next seven columns are used to define which days of the week are to be included in that weekly subset. So, for example, the Weekly1 subset above with the description "Working week" only includes days Monday through to Friday, and since it references the Daily3 subset, only includes hours 10 through to 17 on those days. Conversely, the Weekly2 subset (description "Weekend") includes all hours on days Saturday and Sunday.
- The next line (line 22) is the yearly subset header line, which should read exactly as shown above (replacing the '…' with the remaining months of the year up to December)
- The next nine lines contain the yearly subsets. Again, there must always be nine of these. The first column contains the yearly subset identifier, which must be set to Yearly1, Yearly2, …, Yearly9 (in ascending order). The next column can be used to give each subset a description. The next column is used to reference a weekly subset identifier, or can be set to [All days] to include all days (and hours) of each included month in that yearly subset. The next 12 columns are used to define which months of the year are to be included in that yearly subset. So, for example, the Yearly1 subset above with the description "Working Year" includes all months except July and December, and since it references the Weekly1 subset, only includes days Monday through to Friday and only hours 10 through to 17 on those days. Conversely, the Yearly2 subset (description "Summer") includes all days (and all hours) of months June, July and August.

4.3 Polygon file

An example polygon file is shown below, followed by a description of its contents:

```
POLYVERSION1
6
-20,20
-20,-20
20,-20
20,0
0,0
0,20
```

Notes:

- The first line is the version string
- The second line specifies the number of (distinct) vertices that make up the polygon
- This should be followed by this number of lines, each containing the X and Y coordinates of that polygon vertex (separated by a comma). The vertices should be listed in (anticlockwise or clockwise) order around the polygon. The first vertex should not be repeated at the end

of the list, i.e. the utility will automatically close the polygon by assuming an edge exists between the first and last vertex in the list. As mentioned above, the defined polygon must be simple, i.e. it must not intersect itself or contain any holes.

5 Output

For `RAW` runs, the PostProcessor generates ADMS-style concentration output files (.gst/.glt/.pst/.plt/.levels.gst/.levels.glt) containing the raw or processed concentrations/statistics at the relevant output points in the input netCDF file. Refer to Section 6.1 of the ADMS-Urban v5 User Guide (available here) for details.

For `INTERP` runs, the PostProcessor generates either Surfer-format *.grd* files[1] or generic *.csv* files containing the concentrations/statistics interpolated onto a regular grid of output points. These can be used to create contour plots. By default, the *.csv* files contain three columns; X, Y and the interpolated value at that point. However, since all output files created by a single run of the PostProcessor contain identical X and Y columns, the main input file can be used to specify that only the first file created should include the X and Y columns, thus saving on disk space (see OUTPUTXYCOLS section in §3.2). When processing hourly datasets, if the Averages flag is set to 1 in the *.pop* file for a given output, the end hour of the averaging period associated with a given output file will be included in the file name. Whether processing hourly or period average data, the output file name will also include a description of what is in this file. Some example output filenames are given below:

- *MyStem_2020-180-06_Conc-NOx-2hrs.grd* – This *.grd* file contains NO_X concentrations with an associated averaging time of 2 hours, the last hour of which is hour 2 on Julian day 180 of the year 2020.
- *MyStem_2021-285-24_Conc-NO2-RM-8hrs.csv* – This *.csv* file contains maximum daily ('M') NO_2 concentrations with an associated averaging time of 8 hours rolling ('R'), for Julian day 285 of the year 2021.
- *MyStem_2022-002-24_Conc-PM10-D3Avg-1hr.grd* – This *.grd* file contains average PM_{10} concentrations over the hours included in the Daily3 temporal subset for Julian day 2 of the year 2022
- *MyStem_Conc-NOx.csv* – This *.csv* file contains period average NO_X concentrations.
- *MyStem_CP 99.79-NO2-1hr.grd* – This *.grd* file contains the 99.79th percentile of hourly NO_2 concentrations
- *MyStem_CEXpaM-NO2- 0.20000E+02 ug-m3-1hr.csv* – This *.csv* file contains the annualised number of exceedences of hourly NO_2 concentrations over 200 µg/m^3

For `GRIDAVE` runs, the PostProcessor generates either generic *.csv* files or Surfer-format *.grd* files containing the concentrations/statistics spatially averaged within a set of regular grid cells. These can be used to create raster plots. By default, the *.csv* files contain three columns; the X and Y coordinates of the centrepoint of each grid cell and the grid-averaged value at that point. Again, the OUTPUTXYCOLS section can be used to remove the X and Y columns from all but the first output file. The output file naming convention follows that described above for `INTERP` runs.

For `POLYAVE` runs, the PostProcessor generates a single *.csv* file containing two columns; the first indicates what field is being averaged and the second gives the average value within the specified

[1] https://grapherhelp.goldensoftware.com/subsys/ascii_grid_file_format.htm

polygon. The field name follows the same convention as that used for the output file name (following the stem) described above for `INTERP` runs.

Note that any spatial interpolation/grid averaging/polygon averaging is always done *after* any time averaging has been performed and/or statistics have been calculated.