Ground Radar plugin for EuroScope

- version 1.6 -

Developer Guide

EuroScope | power of control

Table of Contents

1 Installation

The plugin requires the MSVC++ Redistributable package to be installed. Without it, the plugin will either fail to load at all, or crash while trying to load.

- In EuroScope, open the “OTHER SET” menu and click on the “Plug-ins…” item
- Check if the “Ground Radar plugin” is already loaded
    - If not, click on “Load” and select the plugin file (GRplugin.dll)
- Close the Plug-ins Dialog with “Close”
- If you are using other plugins and they need to draw on the screen when a Ground Radar plugin ASR file is open, they need permission to do so. For example, if you are using the TopSky plugin and the aircraft lists use menus by that plugin, the TopSky plugin needs to be given permission to draw on the Ground Radar display:
    - On the plugins list area click on “TopSky plugin”
    - Move “Ground Radar display” from the “Forbidden to draw on types” box to the “Allowed to draw on types” box
- Open the “OTHER SET” menu and click on “Save profile”
    - This will automatically load the plugin with the profile when it’s used the next time
    - You can also select “Save profile as” if you want to create a different profile for this plugin for some reason
- At this point you should see no difference on your screen. To use the plugin, a specific ASR file needs to be created. This is explained in the next chapter.

2 ASR file setup

Regardless of which mode you plan to create, the first step is to create an ASR file. Select the desired sector file, set a good zoom level, use the Display settings dialog to select what you want to be shown and then save the ASR file ("OPEN SCT" menu, "Save as…"). Then exit EuroScope and open the ASR file in a text editor such as Notepad. How to edit the file depends on which of the two modes you wish to see, but in both cases the file should contain the following lines:

```
DisplayTypeName:Ground Radar display
DisplayTypeNeedRadarContent:0
DisplayTypeGeoReferenced:1
```

If some of these items are missing (or contain different values), insert the lines and/or correct the values.

If you are using the TopSky plugin, insert the following line to hide some unnecessary TopSky graphics:

```
PLUGIN:TopSky plugin:NoDraw:1
```

The next step depends on the desired mode:

2.1 Ground mode

The ground mode needs at least one additional line added to the ASR file:

```
PLUGIN:Ground Radar plugin:GroundMode:EFHK
```

It tells the plugin to use the ground movement radar mode and the airport ICAO code. The code is used to set up the Approach and Inbound Windows, as well as coloring the arriving and departing tracks differently.

A bit more information about the airport is required to filter out high flying aircraft from the ground radar display and on-ground traffic from the Approach Window. The information can be in the plugin's settings file, or in the ASR file. If it is entered in the ASR file, use the following two lines:

```
PLUGIN:Ground Radar plugin:AirportElevation:180
PLUGIN:Ground Radar plugin:AirportRadius:1.9
```

The first line is the airport elevation in feet. The second line contains the airport radius in nautical miles. It is the distance from the airport reference point to the furthest corner of the airport. Aircraft within that circle and below elevation+50ft will be filtered out from the Approach Window.

2.2 Tower mode

The tower mode only needs one additional line added to the ASR file:

```
PLUGIN:Ground Radar plugin:TowerMode:ESSA
```

It tells the plugin to use the tower mode and the airport ICAO code to color the arriving and departing tracks differently.

3 Plugin data files

This chapter gives guidance on developing the data files used by the plugin for various features. Even though the plugin does its best to check the data for errors, some errors may get through and cause all kinds of issues, possibly leading to ES crashing, so it's important to be careful to provide correctly formed data when creating the files. Errors in these files discovered by the plugin are reported in a "GRplugin" chat tab when the files are loaded (for some of the files only the first error in the file is displayed).

Coordinate formats

The following coordinate formats can be used to define a <position>, <latitude> or <longitude>:

<position>

Position defined in the active sector file

- Id — Fix, VOR, NDB or airport identifier
- AirportId/rwyId — Runway threshold

Position defined according to ARINC 424 paragraph 7.2.5 rules ([L] is N, E, S or W)

- [L]dddd , d[L]ddd , dd[L]dd or dddd[L] — Full or half degrees of latitude, full degrees of longitude
- Hdddd — Same as Ndddd (i.e. H6030 = N6030 = 60°30'N 030°00'W)

Latitude/longitude coordinates (first [L] is N or S, second [L] is E or W)

- dd[L]ddd[L] — Full degrees of latitude and longitude
- [L]dd[L]ddd — Full degrees of latitude and longitude
- ddmm[L]dddmm[L] — Full degrees and minutes of latitude and longitude
- [L]ddmm[L]dddmm — Full degrees and minutes of latitude and longitude
- ddmmss[L]dddmmss[L] — Full degrees, minutes and seconds of latitude and longitude
- [L]ddmmss[L]dddmmss — Full degrees, minutes and seconds of latitude and longitude

Pad all formats with zeroes as necessary to get correct number of digits (two for latitude degrees, three for longitude degrees and two for all minutes and seconds, with the smallest unit optionally followed by decimal point and one or more digits)

<latitude> or <longitude> ([L] is N, E, S or W)

- d.m.s[L] or [L]d.m.s — Degrees, minutes and seconds (sector file format)
- +d , -d or d — Degrees
- d[L] , [L]d , d°[L] or [L]d° — Degrees
- d°m'[L] or [L]d°m' — Degrees and minutes
- d°m's"[L] or [L]d°m's" — Degrees, minutes and seconds
- dddmm[L] or [L]dddmm — Degrees and minutes (1)
- dddmmss[L] or [L]dddmmss — Degrees, minutes and seconds (1)

1) *Pad these formats with zeroes as necessary to get correct number of digits (two for latitude degrees, three for longitude degrees and two for all minutes and seconds, with the smallest unit optionally followed by decimal point and one or more digits)*

3.1 GRpluginEventStands.txt

This file contains pre-defined stand assignments (for example for event use). The information in the file is only used for stand assignments made inside the defined validity period. The data is automatically reloaded at specified intervals (by default 10 minutes). The following example is used to show the syntax:

START:1608011600	Validity start time
END:1608012100	Validity end time
// EFHK arrivals	Comment
FIN123:EFHK:22	Stand assignment

The file can also optionally point to another local file or a URL to download the data from. In that case, the location must be defined on the first line of the file with format “FILE=” or “URL=” followed by the location. The file path can be absolute or relative to the folder containing the plugin dll. Any following lines in the file are then not used in any way.

Validity start time

START:StartTime

A mandatory line that defines the start time for the file validity period. Any stand assignments made before this time will use the default assignment criteria.

- StartTime UTC time (format YYMMDDHHMM)

Validity end time

END:EndTime

A mandatory line that defines the end time for the file validity period. Any stand assignments made after this time will use the default assignment criteria.

- EndTime UTC time (format YYMMDDHHMM)

Stand assignment

Callsign:Ades:Stand

Pre-defines a stand assignment for a flight matching both the callsign and the destination.

- Callsign Flight callsign (text string)
- Ades Destination airport ICAO code (text string)
- Stand Stand designator (text string, must be found in the GRpluginStands.txt file)

3.2 GRpluginMaps.txt

This file contains the definitions for the plugin drawn maps. The following is an example of the syntax:

COLORDEF:Runway:0:0:0	Color definition
// runway 01L/19R	Comment
MAP:01L/19R	Name
FOLDER:ESSA	Folder
ACTIVE:RWY:ARR:ESSA01L:DEP:*	Active
ACTIVE:RWY:ARR:ESSA19R:DEP:*	Active
ACTIVE:RWY:ARR:*:DEP:ESSA01L	Active
ACTIVE:RWY:ARR:*:DEP:ESSA19R	Active
COLOR:Runway	Color
COORDTYPE:OTHER:POLYGON	CoordType
COORD:N059.38.14.252:E017.54.49.244	Coordinate
COORD:N059.39.58.802:E017.55.26.928	Coordinate
COORD:N059.39.59.055:E017.55.24.239	Coordinate
COORD:N059.38.14.503:E017.54.46.513	Coordinate

The mandatory items for each map are a name, a folder and some type of contents (drawing, runway or stand closure or stand limitation). Whitespace and tab characters will be automatically stripped from the beginnings of lines, and any lines with a first non-whitespace/tab character being "{", "}", ";" or "/" are disregarded.

File version

//VERSION:VersionString

When an URL has been defined to download maps data, the *file version* line is used to check if the downloaded data is newer than the existing data, and should replace it. To display the version string in a map, use "<VersionString>" without the quotes anywhere where a text string is to be drawn, the code will replace it with the active version string.

- If not present in either file, the downloaded data is used
- If present in the existing data only, the existing data is used.
- If present in the downloaded data only, the downloaded data is used
- If present in both files, the VersionStrings are compared, and the downloaded data is used if its VersionString is greater than the existing data's. Note that the comparison is done one character at a time, so "9" is greater than "10", but "09" is less.

Color definition

COLORDEF:ColorName:R:G:B

Every color used in the maps must be defined using one of these lines.

- ColorName — Color name to be used in the Color lines (text string)
- R, G and B — Color's red, green and blue component values (0-255)

Symbol definition

SYMBOLDEF:SymbolName

The first line for each symbol must be a *symbol definition* line.

- SymbolName Symbol name to use for this symbol in the Symbol lines (text string)

The symbol itself can consist of various elements, drawn by the following lines. The X and Y coordinates are relative to the symbol centerpoint, with the X axis having increasing values to the right and the Y axis having increasing values to the down direction. The commands are the same as in the EuroScope Symbology dialog with the exception of the possibility to draw elliptical arcs and the ":" separating the values here so the ES dialog can be used in most cases to test the results.

MOVETO:X:Y

Sets the starting point for the next LINETO command

- X Number of pixels from the symbol centerpoint in the left(-)-right(+) direction
- Y Number of pixels from the symbol centerpoint in the up(-)-down(+) direction

LINETO:X:Y

Draws a straight line from the previous position

- X Number of pixels from the symbol centerpoint in the left(-)-right(+) direction
- Y Number of pixels from the symbol centerpoint in the up(-)-down(+) direction

SETPIXEL:X:Y

Paints the selected pixel

- X Number of pixels from the symbol centerpoint in the left(-)-right(+) direction
- Y Number of pixels from the symbol centerpoint in the up(-)-down(+) direction

ARC:X:Y:Radius:StartAngle:EndAngle
ARC:X:Y:$Radius_X$:$Radius_Y$:StartAngle:EndAngle

Draws a part of a circle

- X Centerpoint offset from the symbol centerpoint in the left(-)-right(+) direction
- Y Centerpoint offset from the symbol centerpoint in the up(-)-down(+) direction
- Radius Arc radius in pixels (to make a circular arc)
- $Radius_X$ Arc radius in relation to the X axis in pixels (to make an elliptical arc)
- $Radius_Y$ Arc radius in relation to the Y axis in pixels (to make an elliptical arc)
- StartAngle Arc starting angle (integer degrees, 0 degrees is at positive X-axis, increasing counterclockwise)
- EndAngle Arc ending angle (integer degrees, 0 degrees is at positive X-axis, increasing counterclockwise)

FILLARC:X:Y:Radius:StartAngle:EndAngle
FILLARC:X:Y:$Radius_X$:$Radius_Y$:StartAngle:EndAngle

Otherwise the same as ARC above but the result is filled

POLYGON:X_1:Y_1:X_2:Y_2:…:X_n:Y_n

Draws a filled polygon with n vertices

Name

MAP:MapName
MAP:MapName:VisibilityOptions

The first line for each map definition must be a *name* line.

- MapName — Map name (text string)
- VisibilityOptions — Optional flag to set the map visibility (one or more of the following):
  - A — Visible on APP Window
  - G — Visible on Ground mode main radar screen
  - T — Visible on Tower mode
  - 2 — Visible on Traffic Situation Windows

If the visibility options flag is not specified, a default value of "GT2" is used.

Folder

FOLDER:FolderName

Every map must have a *Folder* line. There is practically no limit to how many maps a single folder can contain.

- FolderName — Folder name for the map (text string)

Note: the folder name may not start with an empty space character, and may not contain the backslash ("\") character. The vertical bar ("|") character is reserved as well, it is used to create subfolders. For example, "FOLDER:Folder|Subfolder|Subsubfolder" puts the map into a folder called "Subsubfolder" which is a subfolder of "Subfolder", which in turn is a subfolder of the root level folder "Folder". To place the map directly into the root level, use "FOLDER:|".

Airport

AIRPORT:IcaoList

- IcaoList — Comma-separated list of airport ICAO codes (text string)

If this line is used, the map will be displayed in the map list and screens only for the defined airports.

Layer

LAYER:LayerNumber

To adjust the drawing order of the maps, a layer number can be specified. Layers are drawn in increasing order, maps within the same layer in the order they are defined in the data file. Layer 0 is reserved. If a layer is not specified, the map will be placed in layer 1.

- LayerNumber Drawing layer number (valid layers are -999…-1 and 1…999)

Default_Screen-specific

DEFAULT_SCREEN-SPECIFIC

Sets the default visibility state of all following maps to screen-specific unless an *active* line is present. This is the default value when starting to read a map data file.

Default_Global

DEFAULT_GLOBAL

Sets the default visibility state of all following maps to global (synchronized across all radar screens)

Screen-specific

SCREEN-SPECIFIC

Sets this map's visibility state to be screen-specific. Only available for maps with no *active* lines.

Global

GLOBAL

Sets this map's visibility state to be global (synchronized across all radar screens).

Hidden

HIDDEN

Sets the map to be hidden from the maps list. Hidden maps must have at least one *active* line.

Zoom

ZOOM:ZoomLevel

A whole map or parts of it can be hidden based on the current zoom level. With a positive value set, when the radar screen is zoomed out so that there are less than the specified number of pixels per nautical mile, the lines of the map definition after this line are not read. There can be more than one *zoom* line in one map to hide parts of the map at different zoom levels. When the set value is negative, the following lines are not read when the radar screen is zoomed in more than the set value.

- ZoomLevel Radar screen zoom level (pixels per nautical mile, decimal value)

Note : when there is more than one zoom line in a map, their order is important (for example "ZOOM:5" has to be before "ZOOM:10" to have any effect as with zoom below 10 pix/nm the "ZOOM:5" line will never be read if it's after the "ZOOM:10" line...)

ASRdata

ASRDATA:ItemList

Display of certain parts of the map can be decided based on information entered in the currently active ASR file. This line defines the “type” of the following lines in the map until the next *ASRdata* line. It is then checked against any definitions in the ASR file. “ASRDATA:*” reads the following lines regardless of ASR definitions.

- ItemList Comma-separated list of items

The formats to define map data in the ASR file are as follows:

PLUGIN:Ground Radar plugin:ShowMapData:<ItemList>
PLUGIN:Ground Radar plugin:HideMapData:<ItemList>

<ItemList> is a comma-separated list of items. ShowMapData draws only those parts of plugin maps affected by *ASRdata* lines that contain at least one of the defined items, HideMapData hides such parts.

SctData

SCTDATA:Type
SCTDATA:Type\Name
SCTDATA:FREETEXT\Group
SCTDATA:FREETEXT\Group\Name

The *SctData* line is used to draw items from the active sector file data. Specifying only the Type or Group will draw all items. Due to limitations in the data available from EuroScope, the drawing will only look correct when an item contains a single polygon or polyline with the points in consecutive order. The items are drawn using the current map line style and color regardless of what is specified in the sector file.

- Type — Type of item(s) to draw
  - "LOW AIRWAY", "HIGH AIRWAY", "ARTCC LOW", "ARTCC", "ARTCC HIGH", "SID", "STAR" or "GEO"
- Group — Group name of free text item(s) to draw
- Name — Name of item to draw

SctFileData

SCTFILEDATA:Type
SCTFILEDATA:Type\Name
SCTFILEDATA:FREETEXT\Group
SCTFILEDATA:FREETEXT\Group\Name

The *SctFileData* line is used to draw items from a sector file specified with the *SctFilePath* line. The file is parsed directly by the plugin so it is not subject to the same quality of data limitations as the *SctData* line. Specifying only the Type or Group will draw all items. The items are drawn using the current map line style and color unless a specific color is set in the sector file.

- Type — Type of item(s) to draw
  - "LOW AIRWAY", "HIGH AIRWAY", "ARTCC LOW", "ARTCC", "ARTCC HIGH", "SID", "STAR", "GEO", "REGIONS", "SECTOR" or "SECTORLINE"
- Group — Group name of free text item(s) to draw
- Name — Name of item to draw

Note : To force all items to be drawn using the current map color regardless of possible specific colors set in the sector file, start the line with "SCTFILEDATA/MONO" instead of "SCTFILEDATA".

SctFilePath

SCTFILEPATH:FileLocation

The *SctFilePath* line sets the location for the sector file to use as the data source for the *SctFileData* lines. The line can be anywhere in the file, but only once.

- FileLocation Location of the sector file including the file name

The path can be either absolute or relative to the folder where the plugin dll is located. If the path ends with the "*" character, all files matching up to that and having the ".sct" extension will be considered and the one that's alphabetically last will be chosen. (e.g. "..\ABCD-*" will search the parent folder of the plugin dll folder for any files starting with "ABCD-" and with ".sct" extension. If "ABCD-1234.sct" and "ABCD-1235.sct" are found, "ABCD-1235.sct" will be used).

Blocks

BLOCKS:AirportCode:StandList

A map can be set to block specified stands when it is active using this line. A map can contain more than one *blocks* line if necessary. This only blocks the stands from automatic assignment.

- AirportCode Airport ICAO code
- StandList Comma-separated list of stand designators to be blocked

Limits

LIMITS:AirportCode:StandList:WingspanOrCode
LIMITS:AirportCode:StandList:WingspanOrCode:Length

A map can be set to limit aircraft dimensions on specified stands when it is active using this line. A map can contain more than one *limits* line if necessary. The length parameter is optional.

- AirportCode Airport ICAO code
- StandList Comma-separated list of stand designators to be limited
- WingspanOrCode Wingspan in meters (0.1-999) or element 2 of the aerodrome reference code (A-F)
- Length Length in meters (0.1-999)

RwyClosedArr

RWYCLOSEDARR:AirportCode:RwyList

The *RwyClosedArr* line is used to set runways closed for arrivals. When one of these runways is assigned as the arrival runway for an aircraft, an alert will be shown. If the runway surface is closed completely or partially, it needs to be specified separately as a polygon (see the *CoordType* and *coordinate* lines). Only one *RwyClosedArr* line may be used in a map.

- AirportCode Airport ICAO code
- RwyList Comma-separated list of runways

RwyClosedDep

RWYCLOSEDDEP:AirportCode:RwyList

The *RwyClosedDep* line is used to set runways closed for departures. When one of these runways is assigned as the departure runway for an aircraft, an alert will be shown. If the runway surface is closed completely or partially, it needs to be specified separately as a polygon (see the *CoordType* and *coordinate* lines). Only one *RwyClosedDep* line may be used in a map.

- AirportCode — Airport ICAO code
- RwyList — Comma-separated list of runways

Color

COLOR:ColorName
COLOR:ColorName:FillColorName
COLOR:ColorName:FillColorName:FillBgColorName

Every map drawing something on the screen must have at least one *color* line. It sets the color to be used to draw the subsequent drawings. Each line, symbol, etc. within a map can be drawn with a different color by including a new *color* line when a color change is required. If the FillColorName is not specified, it is set to the same color as ColorName. The color names must be defined in the file with a *color definition* line.

- ColorName — Color to be used for drawing lines and texts
- FillColorName — Color to be used for filling FILLARC, POLYGON and region type objects
- FillBgColorName — If specified, used to color the background of the filled area of region type items with a hatch fill. If not specified, a transparent background is set.

Most drawing types support transparency. It can be set by suffixing the desired color name(s) with a forward slash ("/") followed by the transparency value from 0 (fully transparent) to 255 (fully opaque).

Note: Transparency is not supported for SYMBOL or TEXT items, or for the HatchType property of the COORDTYPE lines. They are always drawn fully opaque.

Style

STYLE:StyleName
STYLE:StyleName:Width
STYLE:StyleName/EndCap/JoinType:Width

The *style* line defines the line type for any subsequent *line* items within this map. It is not mandatory; a Solid type line with width 1 pixel will be drawn by default. As with the *color* line, a single map may contain any required number of *style* lines to draw different line styles within the same map.

- StyleName — Style to be used
  - Default: Solid, Alternate, Dash, Dot, DashDot, DashDotDot or Null
  - Custom: a style name defined earlier in the file using a *LineStyleDef* line
- EndCap — Line end cap type (Flat, Round or Square)
- JoinType — Line join type (Bevel, Miter or Round)
- Width — Width of line if greater than 1 (pixels)

Line style definition

LINESTYLEDEF:StyleName:BrushStyle:HatchStyle
LINESTYLEDEF:StyleName:BrushStyle:HatchStyle:StyleArray

Custom line styles can be defined using the *LineStyleDef* line.

- StyleName — Style name to be used in the *Style* lines (text string)
- BrushStyle — Brush style (Solid, Hatched or Null)
- HatchStyle — Hatch style (set any value when brush style is not “Hatched”)
  - / 45-degree upward hatch
  - + Horizontal and vertical crosshatch
  - X 45-degree crosshatch
  - \ 45-degree downward hatch
  - - Horizontal hatch
  - | Vertical hatch
- StyleArray — Comma-separated array of values to define the line style if not a solid line. First value defines the length of the first dash, second the first space, etc. The pattern repeats as necessary when drawing the line – reversing if the number of values is odd (dashes become spaces and vice versa). A maximum of 16 values may be entered to define the style.

Line

LINE:Lat$_1$:Lon$_1$:Lat$_2$:Lon$_2$
LINE:StartPos:EndPos

Draws a line from one point to another using the previously defined line style.

- Lat_1 <latitude> of starting point
- Lon_1 <longitude> of starting point
- Lat_2 <latitude> of end point
- Lon_2 <longitude> of end point
- StartPos Line start <position>
- EndPos Line end <position>

CoordType

COORDTYPE:AreaType:DrawType
COORDTYPE:AreaType:DrawType:HatchType
COORDTYPE:AreaType:DrawType:HatchType:FillType

The *CoordType* line is used to define the type of area drawn with the following *coordinate* lines. The region or polygon is drawn with the color(s) defined in a preceding *color* line.

- AreaType Type of area to be drawn
  - APW Area for the APW safety net
  - RWYCLOSED Area defined as a closed part of a runway
  - TWYCLOSED Area defined as a closed part of a taxiway
  - AREATYPE Area defined as a restricted area for the RVM safety net (1)
  - TWYTYPE Area defined as a restricted taxiway for the RVM safety net (1)
  - OTHER Other type (no safety nets involved)
- DrawType Drawing type for the area
  - POLYGON Area outline drawn
  - POLYLINE Otherwise same as above but does not draw a line from the last coordinate to the first one to close the shape
  - REGION Filled area and outline drawn
  - REGION_FILLONLY Filled area without outline drawn
  - NONE Area not drawn
- HatchType Opaque hatch type (only applicable when DrawType is REGION or REGION_FILLONLY). If not specified, no hatch is drawn.
  - / 45-degree upward left-to-right hatch
  - + Horizontal and vertical crosshatch
  - X 45-degree crosshatch
  - \ 45-degree downward left-to-right hatch

- `-` Horizontal hatch
- `|` Vertical hatch
- 0 No hatch

- FillType — Transparent fill type (only applicable when DrawType is REGION or REGION_FILLONLY). If not specified, a solid opaque fill color is used.
  - 0 — Solid opaque fill
  - 5, 10, 20, 25, 30, 40, 50, 60, 70, 75, 80, 90 — Percentage fill
  - 100 — Solid fill
  - E0 – E52 — Hatch fill (2)

Note 1: When AreaType is set to either AREATYPE or TWYTYPE, the restriction conditions need to be specified using restriction lines (see below) before the CoordType line.

Note 2: The hatch fill values correspond to the GDI+ HatchStyle enumeration values. For example, "E0" sets "HatchStyleHorizontal" and "E6" sets "HatchStyle05Percent" which can also be achieved by "5" (the numeric values for the FillPattern are just shortcuts to the percentage hatch styles).

Restriction

RESTRICTION:Type:Value

The *restriction* line is used to set restrictions to following AREATYPE and TWYTYPE area definitions. More than one *restriction* line can be used, but successive lines with the same Type will override the earlier ones. If a map contains more than one AREATYPE and/or TWYTYPE polygon, any previously set restrictions in the same map will remain valid for all polygons until overridden by new *restriction* lines with the same Types.

- Type — Restriction type
  - MAXWTC — Maximum wake turbulence category (L,M,H or J)
  - MAXWEIGHT — Maximum weight (in kilograms, 0-999999.9)
  - MAXWINGSPAN — Maximum wingspan (in meters, 0-999.9)
  - MAXCODE — Maximum aerodrome reference code (A-F, but only the wingspan is checked)
  - MAXENGINECOUNT — Maximum number of engines (1-99)
  - NOTENGINETYPES — Prohibited engine types (any combination of P, T, J and E)
  - NOTATYP — Prohibited aircraft types (comma-separated list of aircraft type codes)
- Value — Restriction value or list of values where allowed

The NOTATYP type can contain "Groups" in the Value item (which must have been defined earlier in the file using a *group* line). The syntax to use a group is to use "GROUP_<groupname>" in the value list. It is then automatically expanded to the list of items in the group definition.

Group

GROUP:Name:Contents

Defines a text alias to represent a group of values.

- Name — Name for the group
- Contents — Text strings, separated with ":"

Coordinate

COORD:Lat:Lon
COORD:Pos

Defines a vertex point for a filled region or polygon.

- Lat — <latitude>
- Lon — <longitude>
- Pos — <position>

Coordinate_PBD

COORD_PBD:Lat:Lon:Bearing:Distance
COORD_PBD:Pos:Bearing:Distance

Defines line or polygon vertex coordinates to be used later with *CoordLine* or *CoordPoly* lines as a bearing and distance from a specified point.

- Lat — <latitude>
- Lon — <longitude>
- Pos — <position>
- Bearing — Bearing from the point (decimal degrees, 0.0-360.0, or 0.0M-360.0M)
- Distance — Distance from the point (in nautical miles, 0.1-9999.9)

Coordinate_PBX

COORD_PBX:Lat_1:Lon_1:$Bearing_1$:Lat_2:Lon_2:$Bearing_2$
COORD_PBX:Pos_1:$Bearing_1$:Pos_2:$Bearing_2$

Defines line or polygon vertex coordinates to be used later with *CoordLine* or *CoordPoly* lines as an intersection of bearings from two specified points.

- $Lat_{1/2}$ — <latitude>
- $Lon_{1/2}$ — <longitude>
- $Pos_{1/2}$ — <position>
- $Bearing_{1/2}$ — Bearing from the point (decimal degrees, 0.0-360.0, or 0.0M-360.0M)

Coordinate_AF

COORD_AF:Lat:Lon:Radius:Spacing:StartAngle:Direction:EndAngle
COORD_AF:Pos:Radius:Spacing:StartAngle:Direction:EndAngle

Defines a set of vertex points making up an arc to be used later with *CoordLine* or *CoordPoly* lines. The line is expanded to a number of *Coordinate* lines when the maps data file is read. This means small Spacing values may affect performance when the map is active due to the number of line segments being drawn.

- Lat <latitude>
- Lon <longitude>
- Pos <position>
- Radius Arc radius (in nautical miles, 0.1-9999.9)
- Spacing Vertex radial spacing (in degrees, 0.1-120.0)
- StartAngle Arc start angle (decimal degrees, 0.0-360.0, or 0.0M-360.0M for magnetic)
- Direction ">" for clockwise, "<" for counterclockwise
- EndAngle Arc end angle (decimal degrees, 0.0-360.0, or 0.0M-360.0M for magnetic)

Coordinate_CR

COORD_CR:Course:Lat:Lon:Radial
COORD_CR:Course:Pos:Radial

Defines a vertex point as a specified course from the previously defined point to intercept a radial from a specified point be used later with *CoordLine* or *CoordPoly* lines.

- Course Intercept course (decimal degrees, 0.0-360.0, or 0.0M-360.0M for magnetic)
- Lat <latitude>
- Lon <longitude>
- Pos <position>
- Radial Bearing from the point (decimal degrees, 0.0-360.0, or 0.0M-360.0M)

Coordinate_DF

COORD_DF:InitCrs:TurnDir:Radius:Spacing:Lat:Lon
COORD_DF:InitCrs:TurnDir:Radius:Spacing:Pos

Defines a set of vertex points making up a turn direct to a specified point be used later with *CoordLine* or *CoordPoly* lines.

- InitCrs — Initial course (decimal degrees, 0.0-360.0, or 0.0M-360.0M for magnetic)
- TurnDir — Turn direction ("L" or "R" forcing a turn direction, anything else meaning shortest turn)
- Radius (1) — Turn radius (in nautical miles, 0.1-999.9)
- Spacing — Vertex radial spacing (in degrees, 0.1-120.0)
- Lat — <latitude>
- Lon — <longitude>
- Pos — <position>

Note 1: the radius can optionally be defined using a groundspeed value in knots. The value needs to be suffixed with "KTS" in that case, and the radius is automatically calculated assuming a rate of turn 3°/sec, limited to a maximum bank angle of 25°.

Coordinate_FC

COORD_FC:Bearing:Distance

Defines line or polygon vertex coordinates to be used later with *CoordLine* or *CoordPoly* lines as a bearing and distance from the previously defined point.

- Bearing — Bearing from the point (decimal degrees, 0.0-360.0, or 0.0M-360.0M)
- Distance — Distance from the point (in nautical miles, 0.1-9999.9)

Coordinate_FD

COORD_FD:Bearing:Lat:Lon:Radius
COORD_FD:Bearing:Pos:Radius

Defines line or polygon vertex coordinates to be used later with *CoordLine* or *CoordPoly* lines as a bearing from the previously defined point crossing a specified circle.

- Bearing — Bearing from the point (decimal degrees, 0.0-360.0, or 0.0M-360.0M)
- Lat — <latitude>
- Lon — <longitude>
- Pos — <position>
- Radius — Radius of the circle (in nautical miles, 0.1-9999.9)

Coordinate_HM

COORD_HM:Lat:Lon:InbdCrs:TurnDir:Length:Radius:Spacing
COORD_HM:Pos:InbdCrs:TurnDir:Length:Radius:Spacing

Defines a set of vertex points making up a holding pattern to be used later with *CoordLine* or *CoordPoly* lines. The line is expanded to the necessary number of *Coordinate* lines when the maps data file is read. This means small Spacing values may affect performance when the map is active due to the number of line segments being drawn.

- Lat <latitude>
- Lon <longitude>
- Pos <position>
- InbdCrs Holding inbound course (decimal degrees, 0.0-360.0, or 0.0M-360.0M)
- TurnDir Turn direction ("L" or "R")
- Length (1) Length of straight segments (in nautical miles, 0.1-999.9)
- Radius (2) Radius of the turns (in nautical miles, 0.1-999.9)
- Spacing Vertex radial spacing in turns (in degrees, 0.1-120.0)

Note 1: the length can optionally be defined using a time value in minutes. The value needs to be suffixed with "MIN" in that case, and the defined groundspeed (calculated from the radius value) will be used to calculate the length.

Note 2: the radius can optionally be defined using a groundspeed value in knots. The value needs to be suffixed with "KTS" in that case, and the radius is automatically calculated assuming a rate of turn 3°/sec, limited to a maximum bank angle of 25°.

Coordinate_Circle

COORD_CIRCLE:Lat:Lon:Radius:Spacing
COORD_CIRCLE:Pos:Radius:Spacing

Defines a set of vertex points making up a circle to be used later with *CoordLine* or *CoordPoly* lines. The line is expanded to a number of *Coordinate* lines when the maps data file is read. This means small Spacing values may affect performance when the map is active due to the number of line segments being drawn.

- Lat <latitude>
- Lon <longitude>
- Pos <position>
- Radius Radius (in nautical miles, 0.1-9999.9)
- Spacing Vertex radial spacing (in degrees, 0.1-120.0)

Coordinate_Turn

COORD_TURN:Radius:Spacing:InitialTrack:TrackChange

Defines a set of vertex points starting from the previously defined point, making up a circle or a part of it (a circular arc) to be used later with *CoordLine* or *CoordPoly* lines. The line is expanded to the necessary number of *Coordinate* lines when the maps data file is read. This means small Spacing values may affect performance when the map is active due to the number of line segments being drawn.

- Radius (1) — Radius (in nautical miles, 0.1-9999.9)
- Spacing — Vertex radial spacing (in degrees, 0.1-120.0)
- InitialTrack — Track to start the turn from (decimal degrees, 0.0-360.0, or 0.0M-360.0M)
- TrackChange — Amount of turn (decimal degrees, -360.0 to 360.0, negative meaning left turn)

Note 1: the radius can optionally be defined using a groundspeed value in knots. The value needs to be suffixed with "KTS" in that case, and the radius is automatically calculated assuming a rate of turn 3°/sec, limited to a maximum bank angle of 25°.

Coordinate_List

COORD_LIST:ListSep:LatLonSep:CoordList

Defines a set of coordinates. The list is first split to text strings using ListSep as the separator, and then each of them is treated like a *Coord* line – a <position> if not containing LatLonSep, and a <latitude>,<longitude> pair if a LatLonSep character is found.

- ListSep — Separator character between two coordinates
- LatLonSep — Separator character between a latitude,longitude pair
- CoordList — Coordinates (either <latitude> and <longitude> separated by LatLonSep, or <position>) separated by ListSep

Rwy_Area

RWY_AREA:StartLat:StartLon:EndLat:EndLon:Width
RWY_AREA:StartLat:StartLon:EndLat:EndLon:Width:StartOffset:EndOffset

Draws a rectangular area such as a runway with two end coordinates and a width. Optionally, the ends can be offset by a specified distance – positive values are in the direction from the start point to the end point.

- StartLat <latitude>
- StartLon <longitude>
- EndLat <latitude>
- EndLon <longitude>
- Width Width (in meters, 1.0-999.9)
- StartOffset Offset to apply to the start position (in meters, -9999.9 to 9999.9) (default 0)
- EndOffset Offset to apply to the start position (in meters, -9999.9 to 9999.9) (default 0)

The area is drawn using the currently selected style using a solid fill. To specify a different fill type and/or hatch pattern, use "RWY_AREA/<HatchType>/<FillType>" instead of "RWY_AREA" to start the line. For the possible <HatchType> and <FillType> options, see the *CoordType* line.

Rwy_Arrows

RWY_ARROWS:IcaoRwy:StartLat:StartLon:EndLat:EndLon:Width
RWY_ARROWS:IcaoRwy:StartLat:StartLon:EndLat:EndLon:Width:StartOffset:EndOffset:Spacing:Thickness

Draws a series of arrows along a line such as a runway centerline with two end coordinates and a width. Optionally, the ends can be offset by a specified distance – positive values are in the direction from the start point to the end point, and the pattern spacing and thickness of lines can be adjusted.

- IcaoRwy Airport ICAO code and runway identifier written together
- StartLat <latitude>
- StartLon <longitude>
- EndLat <latitude>
- EndLon <longitude>
- Width Width (in meters, 1.0-999.9)
- StartOffset Offset to apply to the start position (in meters, -9999.9 to 9999.9) (default 0)
- EndOffset Offset to apply to the start position (in meters, -9999.9 to 9999.9) (default 0)
- Spacing Pattern spacing (in meters, 1.0-9999.9) (default 270.0)
- Thickness Line thickness (fraction of Width, 0.1-999.9) (default 1.0)

The arrows are by default drawn in Arrival, Departure or Mixed color, depending on the state of the runway. If the runway is not active for arrivals or departures, nothing will be drawn. To set the transparency for the color, use "RWY_ARROWS/<alpha>" instead of "RWY_ARROWS" to start the line (where <alpha> is a number from 0 to 255 specifying the transparency with 0 being full transparent and 255 fully opaque).

To use a fixed color (the currently selected style), use "RWY_AREA/<HatchType>/<FillType>" instead of "RWY_AREA" to start the line. For the possible <HatchType> and <FillType> options, see the *CoordType* line.

Rwy_Xs

RWY_XS:StartLat:StartLon:EndLat:EndLon:Width
RWY_XS:StartLat:StartLon:EndLat:EndLon:Width:StartOffset:EndOffset:Spacing:Thickness

Draws a series of X's along a line such as a runway centerline with two end coordinates and a width. Optionally, the ends can be offset by a specified distance – positive values are in the direction from the start point to the end point, and the pattern spacing and thickness of lines can be adjusted.

- StartLat <latitude>
- StartLon <longitude>
- EndLat <latitude>
- EndLon <longitude>
- Width Width (in meters, 1.0-999.9)
- StartOffset Offset to apply to the start position (in meters, -9999.9 to 9999.9) (default 0)
- EndOffset Offset to apply to the start position (in meters, -9999.9 to 9999.9) (default 0)
- Spacing Pattern spacing (in meters, 1.0-9999.9) (default 270.0)
- Thickness Line thickness (fraction of Width, 0.01-1.0) (default 0.2)

The area is drawn using the currently selected style using a solid fill. To specify a different fill type and/or hatch pattern, use "RWY_XS/<HatchType>/<FillType>" instead of "RWY_XS" to start the line. For the possible <HatchType> and <FillType> options, see the *CoordType* line.

Active

The *active* line is optional. If there is no *active* line, the map will not be automatically activated. A map can contain more than one *active* line; the plugin will check all of them to set the activation status. The map will be activated if at least one of the *active* lines is a match.

ACTIVE:1

Activates the map automatically when the plugin is loaded. Note that this option cannot be used together with other *active* lines.

ACTIVE:SchedStartDate:SchedEndDate:SchedWeekdays:StartTime:EndTime

Used to set activation schedules.

- SchedStartDate First day to activate the map
  - month and day in the format MMDD (for recurring periods every year)
  - year, month and day in the format YYMMDD (for a single period)
- SchedEndDate Last day to activate the map, formats as above
- SchedWeekdays Days of the week to activate the map
  - list of numbers representing the days to activate the map, for example "145" means the map will activate on Mondays, Thursdays and Fridays
  - "0" (zero) to activate the map continuously from StartTime on SchedStartDate to EndTime on SchedEndDate

- StartTime Time to activate the map (UTC time in the format HHMM)
- EndTime Time to deactivate the map (UTC time in the format HHMM)

Note: SchedEndDate and SchedWeekdays only limit the activation of the map. If the activation time extends past midnight, the map stays active until EndTime on the following day.

ACTIVE:RWY:ARR:ArrRwyList:DEP:DepRwyList
ACTIVE:RWY:ARR:ArrRwyList:NotArrRwyList:DEP:DepRwyList:NotDepRwyList

Activates the map based on active runways. If all the specified runway states match, the map is activated. The runway identifiers must be in the format "<4-letter ICAO code><runwayID>", for example "EFHK15".

- ArrRwyList Comma-separated list of runways. Enter "*" to disregard.
- NotArrRwyList Comma-separated list of runways. Enter "*" to disregard.
- DepRwyList Comma-separated list of runways. Enter "*" to disregard.
- NotDepRwyList Comma-separated list of runways. Enter "*" to disregard.

ACTIVE:ID:YourIdList:NotYourIdList:OnlineIdList:NotOnlineIdList

Activates the map based on the current controller position ID, and the IDs of other online controllers. The map is activated if the current controller position ID is found in YourIdList, not found in NotYourIdList, all controllers specified in OnlineIdList and none of the controllers specified in NotOnlineIdList are online.

- YourIdList Comma-separated list of controller IDs (enter "*" to disregard)
- NotYourIdList Comma-separated list of controller IDs (enter "*" to disregard)
- OnlineIdList Comma-separated list of controller IDs (enter "*" to disregard)
- NotOnlineIdList Comma-separated list of controller IDs (enter "*" to disregard)

ACTIVE:CALLSIGN:YourCallsignList:NotYourCallsignList:OnlineCallsignList:NotOnlineCallsignList

As "ACTIVE:ID" above but activates the map based on the controller callsigns instead of position IDs. Partial matches and wildcards are not supported, but consecutive underscore ("_") characters in callsigns are treated as if there was only one.

- YourCallsignList Comma-separated list of controller callsigns (enter "*" to disregard)
- NotYourCallsignList Comma-separated list of controller callsigns (enter "*" to disregard)
- OnlineCallsignList Comma-separated list of controller callsigns (enter "*" to disregard)
- NotOnlineCallsignList Comma-separated list of controller callsigns (enter "*" to disregard)

ACTIVE:LVP:IcaoList:State

Activates the map based on the LVP state. The map is activated when the current airport is found in the defined list and the LVP state matches.

- IcaoList Comma-separated list of airport ICAO codes
- State LVP state that activates the map
  - 1 LVP active
  - 0 LVP not active

ACTIVE:MAP:Type:MapFolder\MapName

Activates the map based on the state of another map. The other map must be defined earlier in the file than this map, and must not be defined as screen-specific (note that maps without an *active* line are screen-specific by default).

- Type — How to activate the map (one of the following):
  - = — Same state as the specified other map
  - ! — Opposite state as the specified other map
- MapFolder\MapName — Folder and name of the other map

ACTIVE:STAND:Type:FoundList
ACTIVE:STAND:Type:FoundList:NotFoundList

Activates the map based on stand occupancy, assignment status, blocked status or limited status. The format to specify a stand is "airportIcao/standName", case sensitive. Stand status is checked in 5 second intervals so the map activation/deactivation has that maximum delay.

- Type — What to check
  - OCCUPIED, ASSIGNED, BLOCKED or LIMITED
- FoundList — Comma-separated list of stands that must all meet the criteria (or "*" to disregard)
- NotFoundList — Comma-separated list of stands none of which can meet the criteria

And_Active

To combine two or more conditions, the first condition must be defined using an *active* line (see above), and the other conditions using *and_active* lines. The syntax for *and_active* is the same as for *active*, the only difference is that instead of starting with "ACTIVE", the *and_active* line definitions start with "**AND_ACTIVE**". More than one set of conditions can be defined just by starting the next set with an *active* line.

The following setup would create two activation rule sets, and the map would activate when either both of the first two conditions are met, or the third one.

ACTIVE:something
AND_ACTIVE:something
ACTIVE:something

Combining multiple lines with time-based schedules within a rule set will not work, the plugin will not attempt to combine the schedules.

Other_Map_Act

OTHER_MAP_ACT:Type:MapFolder\MapName
OTHER_MAP_ACT:Type:MapFolder

Activates another map or all maps in a folder based on the state of this map. The other map(s) must not have any *active* lines, and this and the other map(s) need to have the same screen-specific/global state. This only affects the state of the other map(s) when the state of this map is changed, the other map(s) can later be manually toggled on/off independently.

- Type — How to activate the other map(s) (one of the following):
  - = — Same state as this map
  - ! — Opposite state as this map
- MapFolder\MapName — Folder and name of the other map
- MapFolder — Folder name

FontSize

FONTSIZE:Type:Size
FONTSIZE:0

Each new map starts out with the default font size. It can be modified using the *FontSize* line. All texts after the line in that map use the new size. “FONTSIZE:0” sets the size back to the default value.

- Type — Type of change (the resulting font size is limited to values between 1 and 99)
  - = — sets a new size
  - - — reduces the size from the default by the given amount
  - + — increases the size from the default by the given amount
  - * — multiplies the size of the default by the given amount
- Size — New font size (1-99)

FontStyle

FONTSTYLE:Weight:Italic:Underline:Strikethrough
FONTSTYLE:0

Each new map starts out with the default font style. It can be modified using the *FontStyle* line. All texts after the line in that map use the new style. “FONTSTYLE:0” sets the style back to the default settings.

- Weight — Font weight (0-1000)
  - some example values are 0=default weight, 400=normal, 700=bold
- Italic — Italic (1=yes, 0=no)
- Underline — Underline (1=yes, 0=no)
- Strikethrough — Strikethrough (1=yes, 0=no)

TextAlign

TEXTALIGN:Flags

Sets the default text alignment used in the *Text* and *Symbol* lines. If defined before the first map, becomes the default alignment for all maps. If defined within a map, becomes the default alignment for all following lines of that map.

- Flags — Combination of the following:
  - "L", "C" or "R" — for left, center or right-aligned horizontally
  - "T", "C" or "B" — for top, center or bottom-aligned vertically

By default, the alignment is centered both horizontally and vertically, i.e. the text label is centered on the defined position. Entering for example "LT" puts the label's top left corner in the defined position instead.

Text

TEXT:Lat:Lon:Label
TEXT:Lat:Lon:Label:OffsetX:OffsetY
TEXT:Pos:Label
TEXT:Pos:Label:OffsetX:OffsetY

Draws a text label on the screen. Optionally, the label can be offset a given number of pixels from the given position. The text alignment can be set using the *TextAlign* line (see also note below).

- Lat — <latitude>
- Lon — <longitude>
- PointName — <position>
- Label — Text label (text string)
- OffsetX — Number of pixels to offset the label in the left(-)-right(+) direction
- OffsetY — Number of pixels to offset the label in the up(-)-down(+) direction

Note : to set the text alignment for just this label, it is possible to suffix TEXT with a forward slash followed by the required alignment flags, e.g. TEXT/LT to align the label top left corner on the anchor point.

Symbol

SYMBOL:SymbolName:Lat:Lon
SYMBOL:SymbolName:Lat:Lon:Label:OffsetX:OffsetY
SYMBOL:SymbolName:Pos
SYMBOL:SymbolName:Pos:Label:OffsetX:OffsetY

Draws a predefined symbol on the screen. Optionally, can also display a text label. The text alignment can be set using the *TextAlign* line (see also note below). If a text label is defined, the offset values need to be entered, even if 0.

- SymbolName — Name of symbol
- Lat — <latitude>
- Lon — <longitude>
- Pos — <position>
- Label — Text label (text string)
- OffsetX — Number of pixels to offset the label in the left(-)-right(+) direction
- OffsetY — Number of pixels to offset the label in the up(-)-down(+) direction

Note : to set the text alignment for just this label, it is possible to suffix SYMBOL with a forward slash followed by the required alignment flags, e.g. SYMBOL/LT to align the label top left corner on the symbol centerpoint.

3.3 GRpluginSymbols.txt

This file makes it possible to change the default symbols drawn by the plugin. Any symbols defined here will not be scaled with the GUI scale settings. The following example symbol shows the syntax:

// history symbol	Comment
SYMBOL:GND_HISTORY	Type
FILLRECT:-1:-1:1:1	Definition

Symbol definition

SYMBOL:SymbolType

The first line for each symbol definition must be a type line.

- SymbolType Symbol type (one of the following):
  - Ground mode symbols
    - GND_PRIMARY PSR tracks
    - GND_SSR SSR tracks
    - GND_SSR_SPI SSR tracks with SPI (ident)
    - GND_HISTORY Position history symbols
    - APP_PREDMIN APP Window prediction line minute symbols
  - Tower mode symbols
    - TWR_PRIMARY PSR tracks
    - TWR_SSR SSR only tracks
    - TWR_SSR_SPI SSR only tracks with SPI (ident)
    - TWR_PSR+SSR PSR+SSR tracks
    - TWR_PSR+SSR_SPI PSR+SSR tracks with SPI (ident)
    - TWR_ADSB ADS-B only tracks
    - TWR_ADSB_SPI ADS-B only tracks with SPI (ident)
    - TWR_HISTORY Position history symbols
    - TWR_UNCORR Uncorrelated SSR tracks
    - TWR_UNCORR_SPI Uncorrelated SSR tracks with SPI (ident)

The symbol itself can consist of various elements, drawn by the following lines. The X and Y coordinates are relative to the symbol centerpoint, with the X axis having increasing values to the right and the Y axis having increasing values to the down direction. The commands are the same as in the EuroScope Symbology dialog with the exception of the possibility to draw elliptical arcs and the ":" separating the values here so the ES dialog can be used in most cases to test the results.

MOVETO:X:Y

Sets the starting point for the next LINETO command

- X Number of pixels from the symbol centerpoint in the left(-)-right(+) direction
- Y Number of pixels from the symbol centerpoint in the up(-)-down(+) direction

LINETO:X:Y

Draws a straight line from the previous position

- X Number of pixels from the symbol centerpoint in the left(-)-right(+) direction
- Y Number of pixels from the symbol centerpoint in the up(-)-down(+) direction

SETPIXEL:X:Y

Paints the selected pixel

- X Number of pixels from the symbol centerpoint in the left(-)-right(+) direction
- Y Number of pixels from the symbol centerpoint in the up(-)-down(+) direction

ARC:X:Y:Radius:StartAngle:EndAngle
ARC:X:Y:$Radius_X$:$Radius_Y$:StartAngle:EndAngle

Draws a part of a circle or ellipse

- X Centerpoint offset from the symbol centerpoint in the left(-)-right(+) direction
- Y Centerpoint offset from the symbol centerpoint in the up(-)-down(+) direction
- Radius Arc radius in pixels (to make a circular arc)
- $Radius_X$ Arc radius in relation to the X axis in pixels (to make an elliptical arc)
- $Radius_Y$ Arc radius in relation to the Y axis in pixels (to make an elliptical arc)
- StartAngle Arc starting angle (integer degrees, 0 degrees is at positive X-axis, increasing counterclockwise)
- EndAngle Arc ending angle (integer degrees, 0 degrees is at positive X-axis, increasing counterclockwise)

FILLARC:X:Y:Radius:StartAngle:EndAngle
FILLARC:X:Y:$Radius_X$:$Radius_Y$:StartAngle:EndAngle

Otherwise the same as ARC, but the result is filled

ELLIPSE:X:Y:Radius
ELLIPSE:X:Y:$Radius_X$:$Radius_Y$

Otherwise the same as FILLARC, but always draws a complete circle or ellipse

FILLRECT:Left:Top:Right:Bottom

Draws a filled rectangle

- Left — Left edge offset from the symbol centerpoint in the left(-)-right(+) direction
- Top — Top edge offset from the symbol centerpoint in the up(-)-down(+) direction
- Right — Right edge offset from the symbol centerpoint in the left(-)-right(+) direction
- Bottom — Bottom edge offset from the symbol centerpoint in the up(-)-down(+) direction

POLYGON:X_1:Y_1: X_2:Y_2:…: X_n:Y_n

Draws a filled polygon with n vertices

3.4 GRpluginOperatorInfo.txt

This file lists typical use categories based on callsigns to improve the quality of the stand assignment feature. The file has one callsign per line and the format is *Callsign<TAB>Use*, for example:

```
BOX		C
```

When a callsign is found in this data file, the defined use category will override the aircraft type based one.

Valid entries for the Use field are the use categories – see ICAO_Aircraft.json.

More than one character may be entered. If the aircraft type based use category is found in the list, it is used. If not, all the listed use categories are used when assigning a stand.

Two special items may be used as well:

- Mil Defines the operator as military. This causes the following use categories to be assigned, in the following priority order:
  - "CARGO" found in FPL remarks T
  - aircraft type based use category is "C" T
  - aircraft type based use category is "H" I
  - WTC L or M M
  - WTC H or J T

- Civ Defines the operator as civilian. This causes the following use categories to be assigned, in the following priority order:
  - aircraft type based use category is "T" C
  - aircraft type based use category is "I" H
  - WTC L, M, H or J default use category based on the WTC

3.5 GRpluginSettings.txt & GRpluginSettingsLocal.txt

These files allow adjusting the plugin settings. Each setting must be on its own line, and the syntax is *SettingName=Value*, for example *Equip_ProMode=1* to set the pro mode on by default. It is possible to set most settings to be airport-specific by using an [icao] line. For example, if there is a line *[EFHK]* anywhere in the file, any settings after that are only applied for EFHK until another [icao] line is found.

The settings are read in the following order:

1. System-wide settings from GRpluginSettings.txt
2. System-wide settings from GRpluginSettingsLocal.txt
3. Airport-specific settings from GRpluginSettings.txt
4. Airport-specific settings from GRpluginSettingsLocal.txt

If a given setting is found more than once, the last read value will take effect.

When creating a setup for others to use, it is recommended to only provide GRpluginSettings.txt in the package, leaving GRpluginSettingsLocal.txt for personal preferences such as graphics scaling and window positions.

The available settings, their default values, short descriptions and acceptable values are listed below.

3.5.1 Settings that must be system-wide

Setting name	Default value	Description
Aircraft_DefaultUse_?	A	Default use for unknown WTC category aircraft when unknown (one character, see AircraftInfo data file, or ? to leave unknown)
Aircraft_DefaultUse_L	P	Default use for WTC category L aircraft when unknown (one character, see AircraftInfo data file, or ? to leave unknown)
Aircraft_DefaultUse_M	A	Default use for WTC category M aircraft when unknown (one character, see AircraftInfo data file, or ? to leave unknown)
Aircraft_DefaultUse_H	A	Default use for WTC category H aircraft when unknown (one character, see AircraftInfo data file, or ? to leave unknown)
Aircraft_DefaultUse_J	A	Default use for WTC category J aircraft when unknown (one character, see AircraftInfo data file, or ? to leave unknown)
Aircraft_DefaultWTC	?	Default wake turbulence category when unknown (L, M, H or J to force a category or ? to leave as unknown)
Equip_ModeS	WZLCISG	List of “mode S equipped” equipment code letters
Equip_NonAlt	XTDBMNYCVS	List of “non-altitude-reporting” equipment code letters
Equip_ProMode	0	Pro mode on(1)/off(0)
List_Inbound_DefPos	100,100	Default position (x,y) of the Time To Threshold Lists
List_Outbound_DefPos	100,100	Default position (x,y) of the Dep Timer Lists
Maps_URL		Use this URL to download the maps data file

If the Maps_URL string starts with either ".\" or "..\" it is seen as a local path relative to the plugin folder instead of an URL. The GRpluginMaps.txt file in the plugin folder will be loaded first in any case, and this

location is then checked for data. If newer data is found, it will replace the GRpluginMaps.txt data (in memory only, the file itself will not be updated).

System_APM_MissedApp_ROC	500	Climb rate (in ft/min) above this considered as missed approach (0-9999)
System_Alert_InbdEst	EST	Text to display for the alert (if empty, the alert won't be shown)
System_Alert_Irregular	IRREG.	Text to display for the alert (if empty, the alert won't be shown)
System_Alert_MissedApp	M/A	Text to display for the alert (if empty, the alert won't be shown)
System_Alert_Weather	WEATHER	Text to display for the alert (if empty, the alert won't be shown)
System_AutoAssignDist	100	Default distance from destination [nm] to auto-assign a stand (0-999) Setting the value to zero disables the automatic assignment. Airport-specific distances can be set using the format "ICAO,distance" - each airport in its own settings line - the default value will be used for airports without a specific value
System_EventStands_ReloadInterval	600	Reload interval for event stands datafile in seconds (10-99999)
System_GS_Samples	5	Number of positions used for calculated ground speed (1-19)
System_GS_DeleteMinMax	0	Disregard highest and lowest from above(1)/Use all values(0)
System_GUI_Scale_GlobalMenu	1.0	Graphics scale factor for the top bar and its menus (0.2-10.0)
System_GUI_Scale_Lists	1.0	Graphics scale factor for the plugin lists (0.2-10.0)
System_GUI_Scale_Screen	1.0	Graphics scale factor for the screen drawings (0.2-10.0)
System_GUI_Scale_Tracks	1.0	Graphics scale factor for track related items (0.2-10.0)
System_GUI_Scale_Windows	1.0	Graphics scale factor for the plugin windows (0.2-10.0)
System_NoDupeCodes	1000,2000,2200, 7000,7500,7600, 7700	Comma-separated list of transponder codes for which no DUPE warning will be displayed
System_TagColor	0	The colors used for the plugin's tag items: 0 EuroScope default 1 TopSky plugin (124,1,124) 2 TopSky plugin (COOPANS) (110,153,110) R,G,B TopSky plugin custom (entered RGB value) (the color value in parentheses is the "Concerned" color to use, the other colors are taken from EuroScope settings)
System_Use_Code_7601	1	Enable(1)/disable(0) alerts for transponder code 7601
System_RwyArea	45.0	Default "on runway" distance from centerline in meters (0.1-999.9)
System_RwyArea_LVP	45.0	Default "on runway" distance from centerline in meters in LVP (0.1-999.9)
System_RwyBufferArea	90.0	Default "on runway" buffer area from centerline in meters (0.1-999.9)
System_RwyBufferArea_LVP	90.0	Default "on runway" buffer area from centerline in meters in LVP (0.1-999.9)
System_SchengenArea	BG,BI,EB,ED,EE,EF, EH,EK,EL,EN,EP,ES, ET,EV,EY,GC,LB,LD, LE,LF,LG,LH,LI,LJ,LK, LM,LO,LP,LR,LS,LZ	Comma-separated list of Schengen area ICAO codes
System_SchengenArea_Exclude	ENAS,ENSA,ENSB	Comma-separated list of ICAO codes to exclude from the above list
System_State_OnFreq	ON FREQ	Text for "ON FREQ" ground state (plugin specific state)
System_State_DeIce	DEICE	Text for "DEICE" ground state (plugin specific state)
System_State_StartUp	START UP	Text for "START UP" ground state (ES default state)
System_State_Push	PUSH	Text for "PUSH" ground state (ES default state)
System_State_Taxi_Out	TAXI	Text for "TAXI OUT" ground state (ES default state)
System_State_LineUp	LINE UP	Text for "LINE UP" ground state (plugin specific state)

System_State_Depa	TAKE OFF	Text for “DEPA” ground state (ES default state)
System_State_Taxi_In	TAXI	Text for “TAXI IN” ground state (ES default state)
System_State_Parked	PARKED	Text for “PARKED” ground state (ES default state)
System_Unblock_GS	5	Groundspeed to unblock a manually blocked stand (1-999)
System_UseMouseWheel	1	Enable(1)/disable(0) mouse wheel scrolling for plugin windows & lists
System_UseReportedGS	1	Use reported(1)/calculated(0) ground speed
Window_APP_DefPos	100,100	Default position (x,y) of the Approach Windows
Window_APP_DefSize	200,200	Default size (width,height) of the Approach Windows
Window_APP_Sync_Display	1	Sync display settings of APP Windows yes(1)/no(0)
Window_APP_Sync_Maps	1	Sync map activity status of APP Windows yes(1)/no(0)
Window_APW_DefPos	100,100	Default position (x,y) of the Approach Path Windows
Window_APW_DefSize	370,200	Default size (width,height) of the Approach Path Windows
Window_APW_Sync_Display	1	Sync display settings of Approach Path Windows yes(1)/no(0)
Window_TSW_DefPos	100,100	Default position (x,y) of the Traffic Situation Windows
Window_TSW_DefSize	200,200	Default size (width,height) of the Traffic Situation Windows
Window_TSW_Sync_Display	1	Sync display settings of Traffic Situation Windows yes(1)/no(0)
Window_TSW_Sync_Maps	1	Sync map activity status of Traffic Situation Windows yes(1)/no(0)
Window_TSW_Sync_RawVideo	1	Sync raw video settings of Traffic Situation Windows yes(1)/no(0)

3.5.2 Settings that must be airport-specific

Note: For settings where the default value is either “from ASR file” or “from sct file”, the default data is only read when opening a new ASR. Therefore, when a custom value has previously been set, removing the setting from the settings file will have no effect when reloading the settings file for any ASRs opened before the settings file reload. This limitation only applies to cases where a previously existing setting has been removed, added or changed settings are applied to previously opened ASRs as well. Reloading the settings files also wipes out any default Approach Paths for airports with no paths defined in the settings files. They are recreated when opening a new ASR for that airport.

Setting name	Default value	Description
List_Inbound		Default state and position of the Time To Threshold Lists Format either: “id,state” or “id,state,x-pos,y-pos” - id: runway identifier (or “*” for all runways) - state: 0 (not displayed) 1 (displayed) 2 (automatic) - x-pos, y-pos: list top left corner coordinates
List_Outbound		Default state and position of the Dep Timer Lists Format either: “id,state” or “id,state,x-pos,y-pos” - id: runway identifier (or “*” for all runways) - state: 0 (not displayed) 1 (displayed) 2 (automatic) - x-pos, y-pos: list top left corner coordinates

Setting the state to automatic displays the list when the runway is active. Closing an automatically opened list or manually opening a list from the top menu bar cancels its automatic state.

Window_APP		Default state and position of the APP Windows Format either: “index,state”, “index,state,x-pos,y-pos” or “index,state,x-pos,y-pos,width,height” - index: number of window (starting from 1) - state: 0 (not displayed) or 1 (displayed) - x-pos, y-pos: window top left corner coordinates - width, height: width and height of window
Window_APP_AltFilter		Default APP Window altitude filter Format: “index,altitude” - index: number of window (starting from 1) - altitude: filter altitude (feet, 0-99900)
Window_APP_CenterPos		Default APP Window centerpoint position Format: “index,lat,lon” - index: number of window (starting from 1) - lat: <latitude>

	- lon: <longitude>
Window_APP_Extensions	Default APP Window runway extensions length Format: “index,length” - index: number of window (starting from 1) - length: extensions length (nm, 0-999)
Window_APP_Extensions_TickWidth	Default APP Window runway extensions tick width Format: “index,width” - index: number of window (starting from 1) - width: extensions width (nm, 0.0-99.0)
Window_APP_Rotation	Default APP Window view rotation Format: “index,value” - index: number of window (starting from 1) - value: view rotation (degrees, -360.0-360.0) (or “Auto” to match main display rotation value)
Window_APP_Scale	Default APP Window scale Format: “index,scale” - index: number of window (starting from 1) - scale: scale (pixels/nm, 1.0-100.0)
Window_APW	Default state and position of the Approach Path Windows Format either: “id,state” or “id,state,x-pos,y-pos” - id: approach name - state: 0 (not displayed) 1 (displayed) 2/rwyId (automatic) - x-pos, y-pos: window top left corner coordinates

Setting the state to automatic displays the window when the specified runway (“rwyId”) is active. Closing an automatically opened window or manually opening a window from the top menu bar cancels its automatic state.

Window_WindRose	Default state and position of the Wind Rose Windows Format either: “id,state” or “id,state,x-pos,y-pos” - id: runway identifier (or “*” for all runways) - state: 0 (not displayed) 1 (displayed) 2 (automatic, dep or arr) 3 (automatic, dep) 4 (automatic, arr) - x-pos, y-pos: window top left corner coordinates

Setting the state to automatic displays the window when the runway is active. Closing an automatically opened window or manually opening a window from the top menu bar cancels its automatic state.

Window_TSW	Default state and position of the Traffic Situation Windows Format either: “index,state”, “index,state,x-pos,y-pos” or “index,state,x-pos,y-pos,width,height”

		- index: number of window (starting from 1) - state: 0 (not displayed) or 1 (displayed) - x-pos, y-pos: window top left corner coordinates - width, height: width and height of window
Window_TSW_CenterPos		Default Traffic Situation Window centerpoint position Format: “index,lat,lon” - index: number of window (starting from 1) - lat: <latitude> - lon: <longitude>
Window_TSW_Rotation		Default Traffic Situation Window view rotation Format: “index,value” - index: number of window (starting from 1) - value: view rotation (degrees, -360.0-360.0) (or “Auto” to match main display rotation value)
Window_TSW_Scale		Default Traffic Situation Window scale Format: “index,scale” - index: number of window (starting from 1) - scale: scale (pixels/nm, 1.0-100000.0)
Airport_SMR_Raw	1	Primary radar raw video availability: 0(no), 1(ESE), 2(yes)
Airport_SMR_Track	1	Primary radar tracks availability: 0(no), 1(ESE), 2(yes)
Airport_ModeA	1	Mode A SSR tracks availability: 0(no), 1(ESE), 2(yes)
Airport_ModeS	2	Mode S SSR tracks availability: 0(no), 1(ESE), 2(yes)

These settings define what data will be shown at this airport when using the ‘Pro mode’ of the plugin. Setting “0” will not display that data, “1” will use the radar station definitions in the ESE file and the EuroScope setting that decides if the radar coverage simulation is used (and for the SSR options, also the transponder mode), “2” will display the data regardless of EuroScope and ESE file radar station settings.

Airport_TransAlt	EuroScope	Transition altitude in feet (0-99999)
Airport_MagVar	0.0	Magnetic variation in degrees (-180.0-180), east positive
Airport_Elevation	from ASR file	Airport elevation in feet (-1000-30000)
Airport_Radius	from ASR file	Radius of the airport area from the reference point in nm (0.1-100.0)
Airport_Refpoint	from sct file	Airport reference point <latitude> and <longitude> separated by “:”
Airport_LVP	1	LVP available at the airport, 0(no) or 1(yes)

If the transition altitude is not specified, the EuroScope value is used. The elevation and radius can be defined in the ASR file, but any values in the settings file override those (if defined nowhere, a default value of 0 is used for both). The reference point can be overridden here to optimize the airport area when the reference point is not near the geographical center of the airport area, but this is normally not necessary as the airport area is only used for filtering primary radar raw video and Approach Window traffic.

Airport_Dep_Transfer_Height	99999	Height AAL in feet (50-99999) to give the “Transfer?” info
Airport_Dep_Transfer_Dist	999.9	Distance in nm from ref point (0.1-999.9) to give the “Transfer?” info
Airport_Runway_End	from sct file	Runway end coordinates if different from sct file data. Runway ID, <latitude> and <longitude>, all separated by “:” This setting defines where the runway starts.

Airport_Runway_Thr	from sct file	Runway threshold coordinates if different from sct file data. Format as above.

The runway ends and thresholds can be defined here if the values from the sector file need to be adjusted. One case would be a runway with a displaced threshold: To give correct RIM alerts, the plugin must know the locations of both the physical ends of the runway as well as the thresholds. The default values for both the thresholds and runway ends are the coordinates from the sector file, so depending on which ones are defined in the sector file, the other ones must be specified here.

Airport_Runway_Thr_Elev		Runway threshold elevation, used to calculate QFE. Format “Runway ID,elevation” (elevation in feet, -9999…99999)

Airport_Runway_Approach	from sct file	Approach path to the runway if different from straight-in Runway ID, <latitude>, <longitude>, course and delta distance, separated by “:”

By default, the code for the Time To Threshold Lists as well as the safety nets determines aircraft on approach to a runway assuming the approach is flown on the runway extended centerline. If this is not the case, the approach can be defined here. The latitude and longitude define the “end point” of the approach, course the approach course (degrees true), and delta distance the distance (nm) that the code adds to the aircraft’s distance from the “end point” for display in the TTT List. This setting must be located after the “Airport_Runway_Thr” setting for the same runway if it exists.

Airport_Runway_FinalLength	4.0	Distance in nm from threshold to consider flight on final Runway ID and distance, separated by “:”
Airport_Runway_FinalXTE	0.3:0.3	Cross-track distance in nm to consider flight on final Runway ID, cross-track error left, cross-track error right, all separated by “:”
Airport_Runway_LongFinalLength	30.0	Distance in nm from threshold to consider flight on final Runway ID and distance, separated by “:”
Airport_Runway_LongFinalXTE	0.6:0.6	Cross-track distance in nm to consider flight on final to assigned runway Runway ID, cross-track error left, cross-track error right, all separated by “:”

The first two values are used to determine if an aircraft is on final to a runway regardless of whether it is the assigned runway or not. For closely separated runways, it may be necessary to reduce the XTE value to avoid overlap (the aircraft will be considered to be on final to the first runway meeting the conditions). The third and fourth values are used to determine if the aircraft is on final to the assigned runway if it has not been determined to be on final to any runway using the criteria set by the first two settings.

Airport_Runway_Area	from sct file	Runway area polygon Runway ID and list of polygon vertex coordinates, all separated by “:” or Runway ID and width of area from centerline in meters, separated by “:”
Airport_Runway_Area_LVP	from sct file	Runway area polygon in LVP
Airport_Runway_Buffer	from sct file	Runway buffer area polygon Runway ID and list of polygon vertex coordinates, all separated by “:” or Runway ID and width of buffer from centerline in meters, separated by “:”

Airport_Runway_Buffer_LVP	from sct file	Runway buffer area polygon

The runway areas and buffers are normally constructed automatically from sector file data using default values (System_RwyArea(_LVP) and System_RwyBufferArea(_LVP)), but if that does not provide satisfactory results, the area widths can be defined per runway, or the entire areas can be defined as polygons. The runway id specified may be for either end, the area or buffer is constructed for the whole runway.

Airport_Runway_MaxCode	F	Maximum aerodrome reference code letter for the runway Runway ID and code letter (A-F), separated by ":"
Airport_Runway_MaxWeight	999999.9	Maximum weight for the runway Runway ID and weight in kg (0.1-999999.9), separated by ":"
Airport_Runway_MaxWingspan	999.9	Maximum wingspan for the runway Runway ID and wingspan in meters (0.1-999.9), separated by ":"
Airport_Runway_MaxWTC	J	Maximum wake turbulence category for the runway Runway ID and WTC (L, M, H or J), separated by ":"
Airport_Runway_NotATYP		Prohibited aircraft types for the runway Runway ID followed by ":" and a comma-separated list of types
Airport_Runway_NotEngineTypes		Prohibited engine types for the runway Runway ID followed by ":" and a list of types (can contain P, T, J and E)

The above settings limit the runway area as well as departures and arrivals to both ends, regardless of which end's runway ID is specified. To limit just approaches or departures to a specific runway end, the following settings can be used instead (formats and accepted values as above):

Airport_Runway_MaxCode_Arr
Airport_Runway_MaxCode_Dep
Airport_Runway_MaxWeight_Arr
Airport_Runway_MaxWeight_Dep
Airport_Runway_MaxWingspan_Arr
Airport_Runway_MaxWingspan_Dep
Airport_Runway_MaxWTC_Arr
Airport_Runway_MaxWTC_Dep
Airport_Runway_NotATYP_Arr
Airport_Runway_NotATYP_Dep
Airport_Runway_NotEngineTypes_Arr
Airport_Runway_NotEngineTypes_Dep

Airport_Runway_Closed_Arr	Specified runway ID is closed for all arrivals to close more than one runway, use one of these settings for each runway ID
Airport_Runway_Closed_Dep	Specified runway ID is closed for all departures

3.5.3 Settings that can be either system-wide or airport-specific

Setting name	Default value	Description
AppPath		See separate section Approach path definitions for details
System_GUI_Version	0	GUI version, original(0) or new(1) This setting cannot be entered into the local settings file Highlighted default values are different with this setting at 1, see Different default values with System_GUI_Version=1.
Setup_LoadMap		Displays the defined map(s)
Setup_HideMap		Hides the defined map(s)

The syntax for Setup_LoadMap and Setup_HideMap is either "foldername\mapname" for a single map or "foldername" for all maps in the folder.

When setting up shortcut keys, care should be taken to avoid problems, as the selected key combinations (or single keys) will also be forwarded to EuroScope. If for example <SHIFT>+<A> is defined for a shortcut, using it will also enter the capital letter "A" in EuroScope chat. The shortcuts must also not conflict with other plugins' (such as TopSky) or EuroScope's shortcut keys. If that happens, the results are unpredictable.

Shortcut_Maps	0	Main key to open the Maps Window
Shortcut_Maps_Combo	0x12	Combination key to open the Maps Window

The syntax for these is a MSDN Virtual-Key Code (hex number from 0x00 to 0xFE), or "0" to disable the key. When a combination key is specified, it needs to be kept down while the main key is pressed to activate the shortcut.

Shortcut_Map_Xfr_Combo
Shortcut_Map_Xfr_Key

The syntax for these is almost as above, but an index number from 1 to 10 must be defined first, then a comma, and then the key code (for example "1,0x41" sets the 'A' key to index 1). The default values for the main keys 1-10 correspond to the keys '1'-'0' on top of the keyboard (0x31 to 0x39 and 0x30), and the ALT key (0x12) for all the combination keys.

Shortcut_Map_Target	Map to toggle with the corresponding key

The syntax for these is "index,mapfolder" or "index,mapfolder\mapname", without the quotes. Index from 1 to 10. For example "8,FIX\VFR" assigns index number 8 (which happens to be the '8' key by default) to toggle a map named "VFR" in folder "FIX". The default value for all targets is empty, i.e. no function defined. Note that the same index number can't be set for both map toggling and transfer functions, only one of them. When no mapname is specified, all the maps in the folder are affected. The maps data file is searched in order and the state of the first map found belonging to that folder is toggled, and the rest of the maps in that folder are set to the same state with the first one.

Shortcut_Xfr_Target	Controller ID to initiate transfer to with the corresponding key

The syntax for these is "index,controllerIDlist", without the quotes. Index from 1 to 10, controllerIDlist a comma-separated list of controller IDs to transfer to in order of priority. For example "8,A,B,C" assigns index number 8 (which happens to be the '8' key by default) to initiate transfer to controller ID "A", "B" or "C", selecting the first found online in this priority order. If none of them are online, do nothing. The default

value for all targets is empty, i.e. no function defined. Note that the same index number can't be set for both map toggling and transfer functions, only one of them.

System_GroundMode_AltFilter_AAL	5000	Hide tracks above this height in feet above airport elevation (100-99900)
System_TowerMode_AltFilter_Low	-1000	Hide labels for tracks below this altitude in feet (-1000-99900)
System_TowerMode_AltFilter_High	99900	Hide labels for tracks above this altitude in feet (-1000-99900)
System_GlobalMenu_Date	1	Show(1)/hide(0) date item Global Menu
System_GlobalMenu_Pin	1	Pin(1)/unpin(0) Global Menu
System_GlobalMenu_QNH	1	Show(1)/hide(0) QNH item in Global Menu
System_GlobalMenu_Time	1	Show(1)/hide(0) time in Global Menu with Clock Window open
System_State_CAPS_Lists	0	Capitalize ground state text in plugin lists yes(1)/no(0)
System_State_CAPS_Menus	0	Capitalize ground state text in plugin menus yes(1)/no(0)
System_Units_Altitude	1	Altitudes displayed in meters(0)/feet(1)
System_Units_Distance	1	Distances displayed in kilometers(0)/nautical miles(1)
System_Units_FL	1	Flight levels displayed in meters(0)/feet(1)
System_Units_Speed	1	Speeds displayed in km/h(0)/knots(1)
System_Altitude_Metric_Divisor	1	Metric altitudes/FLs divided by this number for display (1 or 10)
System_Altitude_Metric_Rounding	50	Metric altitudes/FLs rounded to nearest x meters (1-1000)
System_Altitude_QFE	0	Reference to use for displaying altitudes: QNH (0) or QFE (1)
System_APM	1	Approach Path Monitoring on(1)/off(0)/disabled(-1)
System_APW	1	Area Proximity Monitoring on(1)/off(0)/disabled(-1)
System_CBM	1	Clearance Behavior Monitoring on(1)/off(0)/disabled(-1)
System_ECM	1	Emergency Code Monitoring on(1)/off(0)/disabled(-1)
System_ICM	1	Identification Conflict Monitoring on(1)/off(0)/disabled(-1)
System_OSM	1	Occupied Stand Monitoring on(1)/off(0)/disabled(-1)
System_RIM	1	Runway Incursion Monitoring on(1)/off(0)/disabled(-1)
System_RUM	1	Runway Usage Monitoring on(1)/off(0)/disabled(-1)
System_RVM	1	Restriction Violation Monitoring on(1)/off(0)/disabled(-1)
System_RMCA_RWY_Closure	50	Closure speed to trigger the alert in knots (0-1000)
System_RMCA_RWY_Closure_Info	20	Closure speed to trigger the info in knots (0-1000)
System_RMCA_RWY_Closure_LVP	40	Closure speed to trigger the alert in LVP in knots (0-1000)
System_RMCA_RWY_Closure_LVP_Info	20	Closure speed to trigger the info in LVP in knots (0-1000)
System_RMCA_APP_Distance	0.7	Distance to threshold to trigger the alert in nm (0.0-100.0)
System_RMCA_APP_Distance_Info	1.2	Distance to threshold to trigger the info in nm (0.0-100.0)
System_RMCA_APP_Distance_LVP	1.2	Distance to threshold to trigger the alert in LVP in nm (0.0-100.0)
System_RMCA_APP_Distance_LVP_Info	1.7	Distance to threshold to trigger the info in LVP in nm (0.0-100.0)
System_RMCA_APP_Time	20	Time to threshold to trigger the alert in sec (0-999)
System_RMCA_APP_Time_Info	35	Time to threshold to trigger the info in sec (0-999)
System_RMCA_APP_Time_LVP	35	Time to threshold to trigger the alert in LVP in sec (0-999)
System_RMCA_APP_Time_LVP_Info	50	Time to threshold to trigger the info in LVP in sec (0-999)
System_RMCA_rwy_div	5	Divergence rate from runway centerline to inhibit RIM in knots (0-1000)
System_RMCA_rwy_div_LVP	10	Divergence rate from runway centerline to inhibit RIM in LVP in knots (0-1000)
System_RMCA_X_extension	2.0	Runway extensions considered for crossing runway checks in nm (0.0-100.0)
System_RMCA_NoLineUpState	0	Line Up state in use(0)/not in use(1) for RIM alerts
System_CMAC_HighSpeed	55	Speed to trigger the alert in knots (0-1000)
System_CMAC_HighSpeed_Info	40	Speed to trigger the info in knots (0-1000)
System_CMAC_NoTakeOffClr_Speed	20	Speed to trigger the alert in knots (0-1000)
System_CMAC_NoTaxiClr_Speed_Info	10	Speed to trigger the info in knots (0-1000)
System_CMAC_NoContact_Distance_Info	4.0	Distance to threshold to trigger the info in nm (0.0-100.0)
System_CMAC_NoContact_Time_Info	120	Time to threshold to trigger the info in sec (0-999)

Color_APP_Background	0,0,0	R,G,B value of the APP window background color
Color_APP_Runway	160,160,160	R,G,B value of the APP window runway color
Color_APP_RunwayExtension	80,80,80	R,G,B value of the APP window runway extension color
Color_APP_Scale	96,96,96	R,G,B value of the APP window scale and north reference
Color_AppPath_Background	90,90,90	R,G,B value of the AppPath Window background color
Color_AppPath_Grid	105,105,105	R,G,B value of the AppPath Window distance lines
Color_AppPath_Grid5	130,105,80	R,G,B value of the AppPath Window distance lines (every 5th)
Color_AppPath_Path	140,140,140	R,G,B value of the AppPath Window approach vertical path
Color_AppPath_Fill_H	140,125,110	R,G,B value of the AppPath Window horizontal path fill area
Color_AppPath_Fill_V	100,115,130	R,G,B value of the AppPath Window vertical path fill area
Color_AppPath_Track	0,0,130	R,G,B value of the AppPath Window horizontal part track/history
Color_Arrival	255,255,64	R,G,B value of the arrival flight color
Color_Caution	224,224,0	R,G,B value of the caution color
Color_CautionText	0,0,0	R,G,B value of the caution text color
Color_CW_DimText	128,128,128	R,G,B value of the dimmed texts in the Clock Window
Color_Departure	64,255,255	R,G,B value of the departure flight color
Color_Information	0,255,0	R,G,B value of the information color
Color_InformationText	0,0,0	R,G,B value of the information text color
Color_Mixed	128,128,128	R,G,B value of the mixed (arr+dep runway) color
Color_New_Stand	255,127,0	R,G,B value of the changed stand assignment color
Color_RawVideo	224,224,0	R,G,B value of the raw video data (latest positions)
Color_RawVideoHistory	224,224,0	R,G,B value of the raw video data (older positions)
Color_Selected	255,255,255	R,G,B value of color for mouse-over flights
Color_SelectedLabelBorder_App	0,0,0	R,G,B value of edge color for mouse-over APP window label
Color_SelectedLabelBorder_APW	0,0,0	R,G,B value of edge color for mouse-over AppPath window label
Color_SelectedLabelBorder_Gnd	0,0,0	R,G,B value of edge color for mouse-over ground mode label
Color_SelectedLabelBorder_Twr	0,0,0	R,G,B value of edge color for mouse-over tower mode label
Color_SelectedLabelFill_App	255,255,255	R,G,B value of fill color for mouse-over APP window label
Color_SelectedLabelFill_APW	255,255,255	R,G,B value of fill color for mouse-over AppPath window label
Color_SelectedLabelFill_Gnd	255,255,255	R,G,B value of fill color for mouse-over ground mode label
Color_SelectedLabelFill_Twr	255,255,255	R,G,B value of fill color for mouse-over tower mode label
Color_SelectedFieldFill_App	0,0,0	R,G,B value of fill color for APP window mouse-over label field
Color_SelectedFieldFill_APW	0,0,0	R,G,B value of fill color for AppPath window mouse-over label field
Color_SelectedFieldFill_Gnd	0,0,0	R,G,B value of fill color for ground mode mouse-over label field
Color_SelectedFieldFill_Twr	0,0,0	R,G,B value of fill color for tower mode mouse-over label field
Color_SelectedItemFill	64,64,128	R,G,B value of fill color for mouse-over menu items
Color_Text_Notes	255,255,255	R,G,B value of the text notes color
Color_TSW_Background	none	R,G,B value of the TSW window background color
Color_Unknown	224,224,224	R,G,B value of the unknown flight color
Color_Warning	224,0,0	R,G,B value of the warning color
Color_WarningText	255,255,255	R,G,B value of the warning text color
Color_WBackground	165,165,165	R,G,B value of the background color of menus
Color_WBackground2	111,111,111	R,G,B value of the background color of lists and extended Wind Rose Window
Color_WBackground3	199,199,199	R,G,B value of the background color of QNH and LVP boxes

Color_WBackground4	224,224,224	R,G,B value of the background color of windows
Color_WBorder	160,160,160	R,G,B value of the window border color
Color_WForeground	0,0,0	R,G,B value of the texts in menus, lists and extended Wind Rose Window
Color_WForeground2	96,96,96	R,G,B value of the disabled menu items
Color_WForeground3	0,0,0	R,G,B value of the texts in windows
Color_WRW_Background	192,192,192	R,G,B value of the Wind Rose Window center circle and right edge
Color_WRW_Runway	192,192,192	R,G,B value of the Wind Rose Window runway
Color_WRW_WindDir	255,0,0	R,G,B value of the Wind Rose Window wind direction pointer
Color_WRW_WindVar	255,192,0	R,G,B value of the Wind Rose Window wind variation
Color_WSlider_Bg	143,143,143	R,G,B value of the window scrollbar background color
Color_WTitleBar	143,143,143	R,G,B value of the window title bar color
Color_WTitleButton	160,160,160	R,G,B value of the window title bar buttons background color
Color_WTitleText	228,228,228	R,G,B value of the window title bar text color

System_GlobalMenu_Font	EuroScope	Font used for the Global Menu and its submenus
System_GlobalMenu_FontSize	13	Font size for the Global Menu and its submenus (1-99)
System_GlobalMenu_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)

System_Alert_RMCA_Window	Conflict	Alert text to display in the Alerts Window
System_Alert_RMCA_Info_Window	Conflict	Alert text to display in the Alerts Window
System_Alert_No_Takeoff_Clr_Window	No take-off clearance	Alert text to display in the Alerts Window
System_Alert_RIM_Window	Runway incursion	Alert text to display in the Alerts Window
System_Alert_ECM_7500_Window	Squawking 7500	Alert text to display in the Alerts Window
System_Alert_ECM_7600_Window	Squawking 7600	Alert text to display in the Alerts Window
System_Alert_ECM_7601_Window	Squawking 7601	Alert text to display in the Alerts Window
System_Alert_ECM_7700_Window	Squawking 7700	Alert text to display in the Alerts Window
System_Alert_Rwy_Closed_Window	Runway closed	Alert text to display in the Alerts Window
System_Alert_Rwy_Type_Window	Runway type	Alert text to display in the Alerts Window
System_Alert_Twy_Type_Window	Taxiway type	Alert text to display in the Alerts Window
System_Alert_APM_Window	Approach path alert	Alert text to display in the Alerts Window
System_Alert_APW_Window	Area penetration	Alert text to display in the Alerts Window
System_Alert_RVM_Window	Restriction	Alert text to display in the Alerts Window
System_Alert_Twy_Closed_Window	Taxiway closed	Alert text to display in the Alerts Window
System_Alert_High_Speed_Window	High speed	Alert text to display in the Alerts Window
System_Alert_Rwy_Closed_Info_Window	Runway closed	Alert text to display in the Alerts Window
System_Alert_Rwy_Type_Info_Window	Runway type	Alert text to display in the Alerts Window
System_Alert_No_Contact_Info_Window	No contact	Alert text to display in the Alerts Window
System_Alert_High_Speed_Info_Window	High speed	Alert text to display in the Alerts Window
System_Alert_No_Taxi_Clr_Info_Window	No taxi clearance	Alert text to display in the Alerts Window
System_Alert_No_Push_Clr_Info_Window	No pushback clearance	Alert text to display in the Alerts Window
System_Alert_Transfer_Info_Window	Transfer?	Alert text to display in the Alerts Window
System_Alert_Stationary_RPA_Window	Stationary in RPA	Alert text to display in the Alerts Window
System_Alert_ICM_Info_Window	Duplicate SSR code	Alert text to display in the Alerts Window

System_Alert_RMCA_Label	CONFLICT	Alert text to display in the Track label
System_Alert_RMCA_Info_Label	CONFLICT	Alert text to display in the Track label
System_Alert_No_Takeoff_Clr_Label	NO TOF CLR	Alert text to display in the Track label
System_Alert_RIM_Label	RWY INCURSION	Alert text to display in the Track label
System_Alert_ECM_7500_Label	HIJACK	Alert text to display in the Track label
System_Alert_ECM_7600_Label	COMFAIL	Alert text to display in the Track label
System_Alert_ECM_7601_Label	COMFAIL VMC	Alert text to display in the Track label
System_Alert_ECM_7700_Label	EMERG	Alert text to display in the Track label

System_Alert_Rwy_Closed_Label	RWY CLOSED	Alert text to display in the Track label
System_Alert_Rwy_Type_Label	RWY TYPE	Alert text to display in the Track label
System_Alert_Twy_Type_Label	TWY TYPE	Alert text to display in the Track label
System_Alert_APM_Label	APM	Alert text to display in the Track label
System_Alert_APW_Label	APW	Alert text to display in the Track label
System_Alert_RVM_Label	RESTRICTION	Alert text to display in the Track label
System_Alert_Twy_Closed_Label	TWY CLOSED	Alert text to display in the Track label
System_Alert_High_Speed_Label	HIGH SPEED	Alert text to display in the Track label
System_Alert_Rwy_Closed_Info_Label	RWY CLOSED	Alert text to display in the Track label
System_Alert_Rwy_Type_Info_Label	RWY TYPE	Alert text to display in the Track label
System_Alert_No_Contact_Info_Label	NO CONTACT	Alert text to display in the Track label
System_Alert_High_Speed_Info_Label	HIGH SPEED	Alert text to display in the Track label
System_Alert_No_Taxi_Clr_Info_Label	NO TAXI CLR	Alert text to display in the Track label
System_Alert_No_Push_Clr_Info_Label	NO PUSH CLR	Alert text to display in the Track label
System_Alert_Transfer_Info_Label	TRANSFER?	Alert text to display in the Track label
System_Alert_Stationary_RPA_Label	STATIONARY RPA	Alert text to display in the Track label
System_Alert_ICM_Info_Label	DUPE	Alert text to display in the Track label
System_Alert_OSM_Label	STAND	Alert text to display in the Track label

System_Lists_Font	EuroScope	Font used for the plugin lists
System_Lists_FontSize	13	Font size for the plugin lists (1-99)
System_Lists_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)
System_Windows_Font	EuroScope	Font used for the plugin windows
System_Windows_FontSize	13	Font size for the plugin windows (1-99)
System_Windows_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)
Maps_Font	EuroScope	Font used for map texts
Maps_FontSize	11	Font size for map texts (1-99)
Maps_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)
TextNotes_Font	Lucida Sans Unicode	Font used for text notes
TextNotes_FontSize	10	Font size for text notes (1-99)
TextNotes_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)
GroundLabel_Font	Lucida Sans Unicode	Font used for ground mode track labels
GroundLabel_FontSize	13	Font size for ground mode track labels (1-99)
GroundLabel_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)
AppLabel_Font	Lucida Sans Unicode	Font used for APP window track labels
AppLabel_FontSize	13	Font size for APP window track labels (1-99)
AppLabel_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)
ApwLabel_Font	Lucida Sans Unicode	Font used for Approach Path Window track labels
ApwLabel_FontSize	13	Font size for Approach Path Window track labels (1-99)
ApwLabel_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)
TowerLabel_Font	EuroScope	Font used for tower mode track labels
TowerLabel_FontSize	13	Font size for tower mode track labels (1-99)
TowerLabel_FontStyle	0,0,0,0	Font style: weight (0-1000), italic (0/1), underline (0/1), strikeout (0/1)

Label		See separate section Track label definitions for details
Label_Use_Assumed_Other_Colors	0	If set to "1", the departure and arrival colors in track labels are only applied to assumed tracks.
Label_ADEP	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_ADES	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_AFL	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_AFL+VS	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_AHDG	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_ARWY	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_ASSR	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_ATYP	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_CFL	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_DEP	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_DRWY	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_GS	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_MALRT	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_REF	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_REG	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_RMK	1,1,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_SI	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_SID	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_STAND	1,1,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_STAR	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_STS	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_TSSR	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_TYP	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_VS	0,0,1,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_WTC	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_WTG	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
Label_WTG_EU	0,0,0,0	Show(1)/hide(0) field in [APP window, ground, tower, AppPath] labels
AppLabel_UseSelColor	0	Use(1)/Don't use(0) fixed color for mouse-over flights
AppLabel_Transparency_Bd	0	Transparency value, 0(fully transparent) to 255(fully opaque) for edge around mouse-over label
AppLabel_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label background
AppLabel_Field_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label field background
ApwLabel_UseSelColor	0	Use(1)/Don't use(0) fixed color for mouse-over flights
ApwLabel_Transparency_Bd	0	Transparency value, 0(fully transparent) to 255(fully opaque) for edge around mouse-over label
ApwLabel_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label background
ApwLabel_Field_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label field background
GroundLabel	1	Show(1)/hide(0) the label, track symbol, history and heading line in ground mode
GroundLabel_UseSelColor	0	Use(1)/Don't use(0) fixed color for mouse-over flights
GroundLabel_Transparency_Bd	0	Transparency value, 0(fully transparent) to 255(fully opaque) for edge around mouse-over label

GroundLabel_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label background
GroundLabel_Field_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label field background
GroundLabel_Filter_Stby	0	Hide(1)/Show(0) labels for tracks with transponder in standby
GroundLabel_Filter_Squawk_Error	0	Hide(1)/Show(0) labels for tracks not squawking assigned code
GroundLabel_Filter_NoState	0	Hide(1)/Show(0) labels for departures with no ground state set
GroundLabel_Filter_OnFreq	0	Hide(1)/Show(0) labels for tracks with ground state ON FREQ
GroundLabel_Filter_Parked	0	Hide(1)/Show(0) labels for tracks with ground state PARKED
TowerLabel_UseSelColor	0	Use(1)/Don't use(0) fixed color for mouse-over flights
TowerLabel_Transparency_Bd	0	Transparency value, 0(fully transparent) to 255(fully opaque) for edge around mouse-over label
TowerLabel_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label background
TowerLabel_Field_Transparency_Bg	0	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label field background
Track_PredictionLine_APP	0	APP window prediction line length in minutes (0-99)
Track_PredictionLine_APP_Type	0	Prediction line has dots(0), small gaps(1) or nothing(2) at one minute intervals
Track_PredictionLine_APW_sec	0	Approach Path Window prediction line length in seconds (0-999)
Track_PredictionLine_TWR	0	Tower mode prediction line length in minutes (0-99)
Track_HistoryDots_APP	5	Number of history dots in APP window (0-19)
Track_HistoryDots_APW	5	Number of history dots in Approach Path Window (0-19)
Track_HistoryDots_GND	5	Number of history dots in ground mode (0-19)
Track_HistoryDots_TWR	5	Number of history dots in tower mode (0-19)
Track_HistoryDots_TWR_Type	0	History dot a small "+"(0) or a 2x2 pixels box(1)
Track_Heading_Line	0	Show(1)/hide(0) the heading line in ground mode
List_Alerts	0,100,100	Default state and position of the Alerts List Format either: "state" or "state,x-pos,y-pos" - state: 0(not displayed) or 1(displayed) - x-pos and y-pos: list top left corner coordinates
List_Alerts_Items	10	Number of flights to display in Alerts list (5-999)
List_Arrival	0,100,100	Default state and position of the Arrival List (see List_Alerts above for format)
List_Arrival_RWY	0	Show(1)/hide(0) RWY in Arrival list
List_Arrival_TYPE	0	Show(1)/hide(0) TYPE in Arrival list
List_Arrival_WTC	0	Show(1)/hide(0) WTC in Arrival list
List_Arrival_SSR	0	Show(1)/hide(0) SSR in Arrival list
List_Arrival_RULE	0	Show(1)/hide(0) RULE in Arrival list
List_Arrival_ADEP	0	Show(1)/hide(0) ADEP in Arrival list
List_Arrival_ETA	1	Show(1)/hide(0) ETA in Arrival list
List_Arrival_STAND	1	Show(1)/hide(0) STAND in Arrival list
List_Arrival_CLEARANCE	1	Show(1)/hide(0) CLEARANCE in Arrival list
List_Arrival_Items	10	Number of flights to display in Arrival list (5-999)
List_Arrival_MaxTime	60	Display flights in Arrival list this time (min) before arrival (0-999)
List_Arrival_WTC_Type	0	Type of WTC to display: WTC(0), RECAT-EU(1), ICAO WTG(2)
List_Arrival_Sort	1	Sorting criteria for the Arrival list 1(ETA) or 2(callsign), negative value sets descending order

List_Departure	0,100,100	Default state and position of the Departure List (see List_Alerts above for format)
List_Departure_RWY	0	Show(1)/hide(0) RWY in Departure list
List_Departure_SID	0	Show(1)/hide(0) SID in Departure list
List_Departure_TYPE	0	Show(1)/hide(0) TYPE in Departure list
List_Departure_WTC	0	Show(1)/hide(0) WTC in Departure list
List_Departure_SSR	0	Show(1)/hide(0) SSR in Departure list
List_Departure_RULE	0	Show(1)/hide(0) RULE in Departure list
List_Departure_ADES	0	Show(1)/hide(0) ADES in Departure list
List_Departure_EOBT	1	Show(1)/hide(0) EOBT in Departure list
List_Departure_STAND	1	Show(1)/hide(0) STAND in Departure list
List_Departure_CLEARANCE	1	Show(1)/hide(0) CLEARANCE in Departure list
List_Departure_Items	10	Number of flights to display in Departure list (5-999)
List_Departure_WTC_Type	0	Type of WTC to display: WTC(0), RECAT-EU(1), ICAO WTG(2)
List_Departure_SID_MaxChars	7	Maximum characters to display for the SID field
List_Departure_Sort	1	Sorting criteria for the Departure list 1(callsign), 2(ground state) or 3(EOBT), negative for descending
List_Inbound_DTT	0	Show(1)/hide(0) DTT in Time To Threshold lists
List_Inbound_TYPE	0	Show(1)/hide(0) TYPE in Time To Threshold lists
List_Inbound_WTC	0	Show(1)/hide(0) WTC in Time To Threshold lists
List_Inbound_ETA	0	Show(1)/hide(0) ETA in Time To Threshold lists
List_Inbound_STAND	1	Show(1)/hide(0) STAND in Time To Threshold lists
List_Inbound_WTC_Type	0	Type of WTC to display: WTC(0), RECAT-EU(1), ICAO WTG(2)
List_Outbound_TYPE	0	Show(1)/hide(0) TYPE in Dep Timer lists
List_Outbound_WTC	0	Show(1)/hide(0) WTC in Dep Timer lists
List_Outbound_SID	0	Show(1)/hide(0) SID in Dep Timer lists
List_Outbound_ADES	0	Show(1)/hide(0) ADES in Dep Timer lists
List_Outbound_ExpiryTime	600	Automatically remove flights after this time (1-5940sec)
List_Outbound_WTC_Type	0	Type of WTC to display: WTC(0), RECAT-EU(1), ICAO WTG(2)
List_Outbound_SID_MaxChars	7	Maximum characters to display for the SID field
List_Stand	0,100,100	Default state and position of the Stands List (see List_Alerts above for format)
List_Stand_Items	10	Number of stands to display in Stands list (5-999)
Window_APP_AltFilter	99900	Default APP Window altitude filter in feet (0 to 99900)
Window_APP_Extensions	0.0	Default APP Window runway extensions length in nm (0-999) (with metric distance units selected, this value will be doubled, i.e. setting 10 will display 10nm or 20km extensions)
Window_APP_Extensions_TickWidth	0.2	Default APP Window runway extensions tick width in nm (0.0-99.0) (same behavior as Window_APP_Extensions regarding display units)
Window_APP_Number	1	Number of APP Windows (1-10)
Window_APP_Rotation	0.0	Default APP Window view rotation in degrees (-360.0 to 360.0 or “Auto” to match main display rotation value)
Window_APP_Scale	10.0	Default APP Window scale in pixels/nm (1.0-100.0)

Window_APP_Draw_NorthRef	1	Hide(0)/Show(1) north reference if display is not north up
Window_APP_Draw_Runways	1	Hide(0)/Show(1) default runway and extended centerline maps
Window_APP_Draw_Scale	1	Hide(0)/Show(1) scale marker
Window_APW_MaxDist	8	Approach Path Window displayed distance (1-99nm)
Window_Clock	0,100,100	Default state and position of the Clock Window "state" or "state,x-pos,y-pos" - state: 0 (not displayed) or 1 (displayed) - x-pos, y-pos: window top left corner coordinates
Window_Maps_Items	12	Number of lines to display in Map Selection Window (5-999)
Window_Maps_Sorting	0	Map Selection Window sorting criteria: (0) Alphabetic order (1) Data file order
Window_METAR	0,100,100,200,100	Default state and position of the METAR Window "state" or "state,x-pos,y-pos" or "state,x-pos,y-pos,width,height" - state: 0 (not displayed) or 1 (displayed) - x-pos, y-pos: window top left corner coordinates - width, height: window width and height
Window_QNH	0,100,100	Default state and position of the QNH Window "state" or "state,x-pos,y-pos" - state: 0 (not displayed) or 1 (displayed) - x-pos, y-pos: window top left corner coordinates
Window_QNH_DisplayUnits	0	Units displayed by default in the QNH Window (0) Reported units (1) The other of hPa or inHg (2) mmHg (3) All available units
Window_QNH_ShowQFE	0	Hide(0)/show(1) QFE values by default in QNH Window
Window_QNH_SwitchUnits	0	Toggle showing one or all(0)/switch between(1) different units in QNH Window
Window_QNH_Units	3	Units available for display in the QNH Window (0) Reported units only (1) Reported units and the other of hPa or inHg (2) Reported units and mmHg (3) All three units
Window_RwyConf	0,100,100	Default state and position of the Runway Configuration Window "state" or "state,x-pos,y-pos" - state: 0 (not displayed) or 1 (displayed) - x-pos, y-pos: window top left corner coordinates
Window_RVR	0,100,100	Default state and position of the RVR Window "state" or "state,x-pos,y-pos" - state: 0 (not displayed) or 1 (displayed) - x-pos, y-pos: window top left corner coordinates
Window_TSW_Number	1	Number of Traffic Situation Windows (1-10)
Window_TSW_Rotation	Auto	Default Traffic Situation Window view rotation in degrees (-360.0 to 360.0 or "Auto" to match main display rotation value)
Window_TSW_Scale	10.0	Default Traffic Situation Window scale in pixels/nm (1.0-100000.0)

Window_WindRose_Units_Clouds	0	Display cloud heights in (0)feet or (1)meters
Window_WindRose_Units_QFE	0	QFE display units: (0)METAR, (1)hPa, (2)inHg or (3)mmHg
Window_WindRose_Units_Wind	0	Wind display units: (0)METAR, (1)knots, (2)m/s or (3)km/h
RawVideo	1	Show(1)/hide(0) raw video data
RawVideo_Afterglow	30	Raw video data afterglow (0-100)
RawVideo_Brightness	100	Raw video data brightness (0-100)
RawVideo_Gradient	0	(1) Fade all raw video from *Color_RawVideo* to *Color_RawVideoHistory* based on plot age (0) Use *Color_Rawvideo* for latest plot and *Color_RawVideoHistory* for all history plots
RawVideo_MaxHistory	10	Raw video data maximum history positions displayed (0-10)

3.5.3.1 Different default values with System_GUI_Version=1

Setting name	Default value	Description
System_GlobalMenu_Date	0	Show(1)/hide(0) date item Global Menu
Color_Information	64,128,64	R,G,B value of the information color
Color_InformationText	232,232,232	R,G,B value of the information text color
Color_WBackground	96,96,96	R,G,B value of the background color of menus
Color_WBackground2	64,64,64	R,G,B value of the background color of lists and extended Wind Rose Window
Color_WBackground3	64,64,64	R,G,B value of the background color of QNH and LVP boxes
Color_WBackground4	48,48,48	R,G,B value of the background color of windows
Color_WBorder	232,232,232	R,G,B value of the window border color
Color_WForeground	232,232,232	R,G,B value of the texts in menus, lists and extended Wind Rose Window
Color_WForeground2	128,128,128	R,G,B value of the disabled menu items
Color_WForeground3	232,232,232	R,G,B value of the texts in windows
Color_WRW_Background	32,32,32	R,G,B value of the Wind Rose Window center circle and right edge
Color_WRW_Runway	32,32,32	R,G,B value of the Wind Rose Window runway
Color_WSlider_Bg	128,128,128	R,G,B value of the window scrollbar background color
Color_WTitleBar	232,232,232	R,G,B value of the window title bar color
Color_WTitleButton	192,192,192	R,G,B value of the window title bar buttons background color
Color_WTitleText	48,48,48	R,G,B value of the window title bar text color
AppLabel_UseSelColor	1	Use(1)/Don't use(0) fixed color for mouse-over flights
AppLabel_Transparency_Bd	64	Transparency value, 0(fully transparent) to 255(fully opaque) for edge around mouse-over label
AppLabel_Transparency_Bg	64	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label background
AppLabel_Field_Transparency_Bg	128	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label field background
ApwLabel_UseSelColor	1	Use(1)/Don't use(0) fixed color for mouse-over flights
ApwLabel_Transparency_Bd	64	Transparency value, 0(fully transparent) to 255(fully opaque) for edge around mouse-over label
ApwLabel_Transparency_Bg	64	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label background
ApwLabel_Field_Transparency_Bg	128	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label field background

GroundLabel_UseSelColor	1	Use(1)/Don’t use(0) fixed color for mouse-over flights
GroundLabel_Transparency_Bd	64	Transparency value, 0(fully transparent) to 255(fully opaque) for edge around mouse-over label
GroundLabel_Transparency_Bg	64	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label background
GroundLabel_Field_Transparency_Bg	128	Transparency value, 0(fully transparent) to 255(fully opaque) for mouse-over label field background

3.5.4 Approach path definitions

The approaches for the Approach Path Windows are defined using “AppPath” settings lines. The lines may be anywhere within the file. The approach is defined as a straight line with a constant vertical angle leading to a defined point in space. If the cross-track error limits are not defined, a default value of 1 nm on either side will be used.

3.5.4.1 Approach definition

AppPath=Icao:Name:EndLat:EndLon:InbdCrsT:Direction:EndAlt:VertAngle
AppPath=Icao:Name:EndLat:EndLon:InbdCrsT:Direction:EndAlt:VertAngle:XTElimitL:XTElimitR

- Icao — Airport ICAO code
- Name — Approach name to display in list and window title
- EndLat — Approach end point <latitude>
- EndLon — Approach end point <longitude>
- InbdCrsT — Approach inbound course in degrees true (0.0-360.0)
- Direction — Display direction of approach in the window (“L” or “R”)
- EndAlt — Approach end point altitude in feet AMSL (-999 to 99999)
- VertAngle — Approach vertical angle in degrees (0.1-89.9)
- XTElimitL — Filter out tracks more than this distance left of approach course (0.1-99.9nm)
- XTElimitR — Filter out tracks more than this distance right of approach course (0.1-99.9nm)

3.5.4.2 APM definition

To activate APM alerts for the approach, the parameters must be specified after the AppPath line. The only mandatory line is the AppPath_APM line which activates the desired alerts within the specified range.

AppPath_APM=Alerts:Rmin:Rmax

- Alerts — Types of alerts to show (one or more of the following):
  - A — Above path alerts
  - B — Below path alerts
  - L — Lateral deviation alerts
- Rmin — Minimum range from end point to trigger alerts (0.1-99.9nm)
- Rmax — Maximum range from end point to trigger alerts (0.1-99.9nm)

By default, the lateral area for the approach is a rectangle bounded by Rmin, Rmax, XTElimitL and XTElimitR. It can be further reduced (but not expanded) using either an AppPath_APM_Area or an AppPath_APM_Coords line:

AppPath_APM_Area=XTE_init:XTE_slope
AppPath_APM_Area=XTE_init:XTE_slope:XTE_limit

- XTE_init — Half-width of the approach area at Rmin (0.0-99.9nm)
- XTE_slope — Angle by which the area opens outwards (0.0-45.0deg)
- XTE_limit — Maximum half-width of the approach area (0.1-999.9nm)

AppPath_APM_Coords=Lat1:Lon1:Lat2:Lon2:…

- LatX Vertex point <latitude>
- LonX Vertex point <longitude>

By default, the vertical area for the approach starts as a +-20ft window around EndAlt at the approach end point, opening outwards at VertAngle +-1deg. This can be adjusted using a AppPath_APM_Vert line:

AppPath_APM_Vert=VTE_init:AngleLow:AngleHigh

- VTE_init Half-height of approach vertical area at the approach end point (0-9999ft)
- AngleLow Low limit angle in degrees (0.0-VertAngle)
- AngleHigh High limit angle in degrees (VertAngle-90.0)

To specify the lowest altitude to join the final approach path and the altitude above which aircraft are not considered to be on the approach, the AppPath_APM_Vert_Lim line can be used:

AppPath_APM_Vert_Lim=JoinAlt:OverflyAlt

- JoinAlt No below path alerts above this altitude (-2000-99999ft)
- OverflyAlt No above path alerts above this altitude (-2000-99999ft)

By default, the system starts to monitor the approach when the aircraft is within the approach lateral area and its track error is less than 20 degrees (Tolerance), and issues a lateral alert even within the area if the track error thereafter becomes greater than 30 degrees (Tolerance+Deviation). To adjust these values, the AppPath_APM_TKE line can be used.

AppPath_APM_TKE=Tolerance:Deviation

- Tolerance Track error tolerance (0-180deg)
- Deviation Track error deviation (0-180deg)

3.5.5 Track label definitions

Using the "Label" setting it is possible to customize the track labels for the ground and tower modes, as well as the APP window. For example, the default ground mode track label for an arrival track would be defined as follows:

```
Label=GND:ARR:0:ALRT,0,0:ASSR_E,0,1:COMM,0,1
Label=GND:ARR:1:CALLSIGN,0,0:STAND,0,1
Label=GND:ARR:2:ATYP,0,0:WTC,0,1:RMK,0,1
Label=GND:ARR:3:
```

3.5.5.1 Label setting syntax

Label=Mode:Type:Line:Field$_1$:Field$_2$:…

- Mode — Label mode: "APP", "APW", "GND", "TWR" or "ALL" (APP window, App Path Window, ground mode, tower mode or all of them)
- Type — Track type: "ARR", "DEP", "OTH", "UNC" or "ALL" (arrival, departure, other/unknown, uncorrelated or all of them)
- Line — Label line number: "0", "1", "2" or "3" (line 1 is the main line to which the leader line connects to)
- Field$_X$ — Label field to display, see below for details (to leave a line empty, do not define any fields as in the example's line 3)

3.5.5.2 Field item syntax

Name,MinPos,SpacesBefore
Name,MinPos,SpacesBefore,ExtraData

- Name — Field name
- MinPos — Minimum position (number of characters) from left edge of label
- SpacesBefore — If not the first displayed item in the line, needs at least this many whitespace characters between this and the previous item
- ExtraData — Field-specific settings (numeric value, defaults to 0 if not entered – see AFL field for guidance and example)

3.5.5.3 Available field types

Name	Description	ExtraData
A_FLAG	"A" if any manual alerts active	
ADEP	Departure aerodrome ICAO code	
ADES	Destination aerodrome ICAO code	

Name	Description	ExtraData
AFL	Actual flight level (feet/100, meters/10 or meters)	Group 1 (altitude number format): - "0" (default) see below - "2" displays minimum 2 digits - "3" displays minimum 3 digits - "4" displays minimum 4 digits - "5" displays minimum 5 digits Group 2 (FL number format): - "0" (default) see below - "20" displays minimum 2 digits - "30" displays minimum 3 digits - "40" displays minimum 4 digits - "50" displays minimum 5 digits Group 3 (altitude prefix): - "0" (default) prefix altitudes with "A" ("M" for meters, "E" for QFE with both units) - "100" prefix with empty space - "200" no prefix Group 4 (FL prefix): - "0" (default) no prefix - "1000" prefix with empty space - "2000" prefix with "F" ("S" for meters) The default value for group 1 displays 3 digits for feet/100, 4 digits for meters/10 and meters. The default value for group 2 displays 3 digits for feet/100, 4 digits for meters and 5 digits for meters. No more than 3 digits will be displayed for feet/100, the higher settings apply to metric units only. More than one option can be selected by adding their values together ("23" sets "3" and "20"), but only one option from each Group can be selected.
AFL+VS	AFL and VS fields combined	Groups 1 - 4: - same as AFL field Group 5 (VS format): - "0" (default) displays arrows (must use ES font) - "10000" displays "v" for descent, "^" for climb
AHDG	Assigned heading	- "0" (default) does not show DCT point - "1" shows DCT point if no AHDG value is set
ALRT	Alert indicator (displays various alerts and infos)	
ARWY	Arrival runway (automatically displayed with Warning background if not an active arrival runway)	- "0" (default) no additional behavior - "1" also automatically displayed when there is more than one active arrival runway
ASSR	Assigned SSR code	- "0" (default) no additional behavior - "1" prefixes the code with "A"

Name	Description	ExtraData
ASSR_E	SSR code error indicator (displays ASSR code in Warning color if different from transponded code)	- "0" (default) no additional behavior - "1" prefixes the code with "A"
ATYP	Aircraft type (ICAO)	
CALLSIGN	"Pro mode": - Flight plan callsign for correlated tracks - TSSR code for uncorrelated secondary tracks - "----" for uncorrelated primary tracks Default: - Radar target callsign	
CFL	Cleared flight level - "CA" for cleared for approach, - "VA" for cleared for visual app	Groups 1 - 4: - same as AFL field Group 5 (auto-hide when within 300ft of AFL): - "0" (default) hides field - "10000" does not hide field
COMM	Communication type indicator - "r" for voice receive only - "t" for text only (displayed in Warning color)	- "0" (default) no additional behavior - "1" prefixes the indicator letter with "/"
DEP	Departure (displays AHDG if set, SID if not)	
DRWY	Departure runway (if RUM is selected on, automatically displayed with Warning background if not an active departure runway)	- "0" (default) no additional behavior - "1" automatically displayed when there is more than one active departure runway
FLTID	Mode S downlinked callsign	- "0" (default) no additional behavior - "1" only display with plugin in "pro mode"
FLTID_E	Mode S correlation error (if mode S downlinked callsign is different from correlated flightplan callsign, displays in Warning color: - on unselected label: "S" - on selected label: mode S downlinked callsign	
GS	Ground speed (knots or km/h)	Group 1 (number format): - "0" (default) displays exact value - "1" displays value divided by 10 - "2" displays value rounded to nearest 10 - "3" displays value with one decimal Group 2 (prefix): - "0" (default) no prefix - "10" prefix number with "N" character ("K" with units set to km/h) - "20" prefix number with "G" character
MALRT	Manual alerts	- "0" (default) displayed always - "1" displayed only on selected label

Name	Description	ExtraData
REF	Aerodrome reference code letter (“A”-“F”, based on wingspan only)	
REG	Aircraft registration code (searched for in the FPL remarks)	
RMK	Remarks (scratchpad text)	
SI	Sector Indicator (transfer in: tracking controlled id + “>” transfer out: “>” + target controller id otherwise: tracking controller id)	Group 1 (display in unselected label when not in transfer in or out state): - “0” (default) displays any id - “1” displays id if tracked by someone else - “2” displays id if tracked by you - “3” displays nothing Group 2 (blink during transfer): - “0” (default) no blinking - “10” field blinks during transfer in - “20” field blinks during transfer out - “30” field blinks during transfer in & out
SID	SID designator	Group 1 (number of first letters to display): - “0” (default) no limit - “1”-“9” maximum number of letters Group 2 (display procedure number and letter part): - “0” (default) yes - “10” no
STAND	Assigned arrival stand name (in New Stand color if assignment has been changed and not acknowledged)	
STAR	STAR designator	Group 1 (number of first letters to display): - “0” (default) no limit - “1”-“9” maximum number of letters Group 2 (display procedure number and letter part): - “0” (default) yes - “10” no
STS	Ground state	- “0” (default) does not alter the text - “1” capitalizes the text
TSSR	Transponded SSR code (meant to be used in correlated labels)	
TYP	Aircraft type (IATA) (may not be the correct one in cases where more than one choice exists for a given aircraft ICAO code)	
VS	Vertical speed indicator	- “0” (default) displays arrows (must use ES font) - “1” displays “v” for descent, “^” for climb (blank space displayed for level flight or unknown)
WTC	Wake turbulence category	- “0” (default) displays only the WTC letter - “1” prefixes the WTC letter with “/”
WTG	Wake turbulence group (ICAO)	- “0” (default) displays only the WTG letter - “1” prefixes the WTG letter with “/”
WTG_EU	Wake turbulence group (RECAT-EU)	- “0” (default) displays only the WTG letter - “1” prefixes the WTG letter with “/”

3.5.5.4 Field mouse functions

Name	Left-click	Right-click
AHDG	Open assigned heading popup list	
ARWY	Open RWY setup popup list	
CALLSIGN	Open Callsign menu	
CFL	Open temporary altitude popup list	
DEP	Open SID setup popup list	Open assigned heading popup list
RMK	Edit scratch pad string	
SID	Open SID setup popup list	
STAND	New assignment: Acknowledge it Otherwise: Open Stand assignment menu	
STAR	Open STAR setup popup list	
STS	Open ground state popup list	

3.6 GRpluginStands.txt

The file contains the stand definitions to be used when assigning arrival stands via the plugin. The following example shows the syntax (optional lines in grey color):

// stand 221	Comment
STAND:EFHK:221:N060.18.39.640:E024.58.42.050:20	Stand
BLOCKS:222	Blocks
WTC:LM	WTC
PRIORITY:-1	Priority
USE:C	Use

The plugin's stand assignment system checks the flightplan, finds out which code stands are available for that flightplan, checks for stands already in use, and then assigns one of the available stands.

The example stand definition above defines a stand "221" at EFHK, at the given coordinates and with a 20m radius. Whenever it's assigned or occupied, it blocks stand "222". It's restricted to cargo flights with wake turbulence categories light or medium. It's also classified as a lesser priority stand, to be used only when higher priority stands are not available for assignment.

If automatic stand assignment fails to find a suitable stand, the plugin tries again with some restrictions removed (first the CALLSIGN, then also the NOTCALLSIGN, USE*, ADEP, NOTADEP, SCHENGEN, NONSCHENGEN, VIA, NOTVIA, REMARKS and NOTREMARKS restrictions are lifted).

*) USE restriction is not lifted if the stand is for helicopter use only ("H", "I" or "HI")

Group

GROUP:GroupName:Item$_1$:Item$_2$:Item$_3$:…

The *group* line can be used as a shortcut to writing a large number of text entries. It can be used in line types where lists of text strings are used. To use a group in a line, enter "GROUP_<groupname>" like any other text string. It will be automatically expanded to the list of text strings in the group definition.

- GroupName Group name
- Item$_x$ Text string

Note: the item separator to be used here is the colon (:), regardless of what's used in the target line type.

Standlist

STANDLIST:Icao:CallsignList:Points:StandList

The *standlist* line can be used together with or as an alternative to the *callsign* and *notcallsign* lines to assign stands based on callsign data. The plugin attempts to assign one of the specified stands for matching callsigns if possible (priority numbers and any stand-specific restrictions except *callsign* lines are taken into account). More than one *standlist* line can be used and the relative priorities of the lines can be adjusted as needed to make certain stands more likely to be assigned than others. For example, with two *standlist* lines, to make the first set 10 times more likely to be chosen, the points of the lines can be set to 10 and 1 (100 and 10 will achieve the same result, it's the ratio of the numbers that matters). To have a set of stands

to be assigned only when none of the stands in these two lines are available, define one or more *standlist* lines with points set to 0 (zero chance of being assigned compared to the higher points lines)

- Icao Airport ICAO code
- CallsignList Comma-separated list of callsign beginnings
- Points Priority points for the stands in this line (0-10000)
- StandList Comma-separated list of stands

Stand

STAND:AirportCode:StandName:Lat:Lon:Radius
STAND:AirportCode:StandName

This line is the only mandatory line for a stand definition and must always be the first line in a definition. The latitude, longitude and radius define a circle which is used to check if the stand is occupied. In the second version, the stand area is defined using *coord* lines.

- AirportCode Airport ICAO code
- StandName Designator for the stand
- Lat <latitude> of the stand area
- Lon <longitude> of the stand area
- Radius Radius of the stand area (meters, decimal value)

Coord

COORD:Lat:Lon

The *coord* line is used to define a polygon vertex when defining a stand area as a polygon.

- Lat Vertex <latitude>
- Lon Vertex <longitude>

Heading

HEADING:HdgT
HEADING:HdgT:Tolerance

An optional heading value for the stand. If specified, the stand is considered occupied only when the aircraft heading in the stand area is within the given tolerance from the specified heading. If the tolerance value is not specified, it defaults to 20 degrees. This can be used to mark the correct stand as occupied when two or more stands overlap but are differently orientated.

- HdgT Heading in degrees true (0.0 to 360.0)
- Tolerance Allowed error in the heading in degrees (0.0 to 180.0)

Manual

MANUAL

Excludes this stand from automatic assignment.

Area

AREA

Defines the stand as an area capable of parking (and being assigned to) multiple aircraft at the same time. These stands can be blocked but not manually assigned using the Stand Window.

Copy

COPY

Ends the current stand definition and immediately starts a new one as a copy of it. Can be used for example to create a lesser priority copy for another use. Note that the following definitions are not carried over and need to be stated again if relevant: *ADEP/NotADEP, Callsign/NotCallsign, Remarks/NotRemarks, Via/NotVia* and *Schengen/Non-Schengen*. If necessary, the other definitions can be adjusted by stating them again.

Blocks

BLOCKS:StandList
BLOCKS:StandList:WingspanOrCode
BLOCKS:StandList:WingspanOrCode:Length

Blocks the specified stands from assignment when this stand is assigned or occupied. The first type blocks the specified stand(s) regardless of the dimensions of the aircraft on this stand, the second one only when the wingspan on this stand exceeds the given value, and the third only when either the wingspan or the length on this stand exceeds the given values. The wingspan value can be set also based on the aerodrome reference code letter.

- StandList — Comma-separated list of stands to be blocked
- WingspanOrCode — Wingspan in meters (0.0-999.9) or code letter (A-F)
- Length — Length in meters (0.0-999.9)

Limits

LIMITS:StandList:WingspanOrCode:LimitedWOC
LIMITS:StandList:WingspanOrCode:LimitedWOC:Length:LimitedLength

Limits the maximum aircraft dimensions of other stands when this stand is assigned or occupied. The first type sets a wingspan limit to the specified stand(s) when the wingspan on this stand exceeds the WingspanOrCode value, the second sets both wingspan and length limits when either the wingspan or the length on this stand exceed the WingspanOrCode and Length values. The wingspan values can be set also based on the aerodrome reference code letter.

- StandList — Comma-separated list of stands to be limited
- WingspanOrCode — Wingspan in meters (0.0-999.9) or code letter (A-F)
- LimitedWOC — Limited wingspan value (0.1-999.9) or code letter (A-F) on the specified stand(s)
- Length — Length in meters (0.0-999.9)
- LimitedLength — Limited length (0.1-999) on the specified stand(s)

Priority

PRIORITY:PriorityNumber

Assigns a priority group number to the stand. When assigning stands, available ones are checked according to priority groups, with the higher priority stands first. Stands in lower priority groups will be assigned only when matching higher priority ones are not available. A neutral (zero) priority is assigned by default.

- PriorityNumber Priority group number (from -3 to +3)

Use

USE:Users

Limits the stand assignment only to specified categories of aircraft. By default, all categories are allowed.

- Users Allowed categories (any combination of the following)
  - A (airliners/commuter aircraft)
  - B (business/corporate aircraft)
  - C (cargo aircraft)
  - H (helicopters)
  - I (military helicopters)
  - M (military aircraft)
  - P (private aircraft)
  - T (military tanker/transport aircraft)

Note: The categories are checked from the ICAO_Aircraft.json data file. For the cargo category also the flightplan remarks field is checked for "CARGO" and the callsign is compared against the list in the OperatorInfo data file.

Schengen

SCHENGEN

Causes the stand to be assigned only to flights arriving from the Schengen area.

Non-Schengen

NON-SCHENGEN

Causes the stand to be assigned only to flights arriving from outside the Schengen area.

ATYP

ATYP:TypeList

Limits the stand assignment to the specified aircraft types. To allow partial matches, enter "*" after the type string. For example, "A3" will only look for A3, whereas "A3*" will match anything beginning with A3 and having at least one character after it, so A3 will not be a match but A320 will be.

- TypeList Comma-separated list of aircraft types (complete or partial)

NotATYP

NOTATYP:TypeList

Blocks the stand assignment from the specified aircraft types. Otherwise, the format and limitations are the same as in the *ATYP* line.

- TypeList Comma-separated list of aircraft types (complete or partial)

WTC

WTC:CategoryList

Limits the stand assignment to the specified wake turbulence categories.

- CategoryList Allowed category letters (any combination of the following)
  - L (light)
  - M (medium)
  - H (heavy)
  - J (super)

NotWTC

NOTWTC:CategoryList

Blocks the stand assignment from the specified wake turbulence categories.

- CategoryList Blocked category letters (see *WTC* line above for available options)

EngineType

ENGINETYPE:TypeList

Limits the stand assignment to the specified engine types.

- TypeList Allowed type letters (any combination of the following)
  - E (electric)
  - J (jet)
  - P (piston)
  - R (rocket)
  - T (turboprop/turboshaft)

NotEngineType

NOTENGINETYPE:TypeList

Blocks the stand assignment from the specified engine types.

- TypeList Blocked type letters (see *EngineType* line above for available options)

Wingspan

WINGSPAN:MinSpan:MaxSpan
WINGSPAN:MaxSpan

Limits the stand assignment based on aircraft wingspan. The first option sets both the minimum and maximum values while the second only limits the maximum value.

- MinSpan Minimum allowed wingspan (meters, decimal value)
- MaxSpan Maximum allowed wingspan (meters, decimal value)

Note: The wingspans are checked from the ICAO_Aircraft.json data file. In case the information for a specific type is not found, a fixed value based on the wake turbulence category and group is used.

Length

LENGTH:MinLength:MaxLength
LENGTH:MaxLength

Limits the stand assignment based on aircraft fuselage length. The first option sets both the minimum and maximum values while the second only limits the maximum value.

- MinLength Minimum allowed fuselage length (meters, decimal value)
- MaxLength Maximum allowed fuselage length (meters, decimal value)

Note: The fuselage lengths are checked from the ICAO_Aircraft.json data file. In case the information for a specific type is not found, a value of 0 meters is used.

Height

HEIGHT:MinHeight:MaxHeight
HEIGHT:MaxHeight

Limits the stand assignment based on aircraft height. The first option sets both the minimum and maximum values while the second only limits the maximum value.

- MinHeight Minimum allowed height (meters, decimal value)
- MaxHeight Maximum allowed height (meters, decimal value)

Note: The heights are checked from the ICAO_Aircraft.json data file. In case the information for a specific type is not found, a value of 0 meters is used.

MTOW

MTOW:MinMTOW:MaxMTOW
MTOW:MaxMTOW

Limits the stand assignment based on the aircraft's maximum takeoff weight. The first option sets both the minimum and maximum values while the second only limits the maximum value.

- MinMTOW Minimum allowed MTOW (kilograms, decimal value)
- MaxMTOW Maximum allowed MTOW (kilograms, decimal value)

Note: The MTOWs are checked from the ICAO_Aircraft.json data file. In case the information for a specific type is not found, a fixed value based on the wake turbulence category and group is used.

Code

CODE:MinCode:MaxCode
CODE:MaxCode

Limits the stand assignment based on element 2 of the aerodrome reference code. The first option sets both the minimum and maximum values while the second only limits the maximum value.

- MinCode Minimum allowed code letter (A-F)
- MaxCode Maximum allowed code letter (A-F)

Note: This line type currently enforces only the wingspan limits (i.e. CODE:C has the same effect as WINGSPAN:35.999).

Callsign

CALLSIGN:CallsignList

Causes the stand to be assigned only to matching callsigns. Stands with a *callsign* line are always assigned first even when higher priority stands but without *callsign* lines would be available.

- CallsignList — Comma-separated list of partial or full callsigns ("SAS" matches with all callsigns beginning with "SAS")

NotCallsign

NOTCALLSIGN:CallsignList

Causes the stand to be blocked from the matching callsigns.

- CallsignList — Comma-separated list of partial or full callsigns ("SAS" matches with all callsigns beginning with "SAS")

ADEP

ADEP:ICAOcodeList

Limits the stand assignment to flights departing from one of the defined airports. The whole ICAO airport code is not needed; the match can also be done on the first one or more letters, e.g. entering "EF" will match all airports with ICAO designators beginning with "EF". The *ADEP* line can contain one or more airport codes and one stand definition can also have more than one *ADEP* line if necessary.

- ICAOcodeList — Comma-separated list of airport ICAO codes (complete or partial)

NotADEP

NOTADEP:ICAOcodeList

Limits the stand assignment to flights not departing from any of the defined airports. Otherwise, the format and limitations are the same as in the *ADEP* line.

- ICAOcodeList — Comma-separated list of airport ICAO codes (complete or partial)

Via

VIA:PointList

Limits the stand assignment to flights routing via at least one of the defined points. The point can be anywhere along the flightplan. One or more points can be defined in one *via* line and one stand definition can contain more than one *via* line if necessary.

- PointList Comma-separated list of point names (Fix, VOR, NDB or airport)

NotVia

NOTVIA:PointList

Limits the stand assignment to flights not routing via any of the defined points. Otherwise, the format and limitations are the same as in the *via* line.

- PointList Comma-separated list of point names (Fix, VOR, NDB or airport)

Remarks

REMARKS:Text$_1$:Text$_2$:Text$_3$:…

This line limits the stand assignment based on the flightplan remarks. When all the specified text strings are found in the remarks section, the line is a match. The *remarks* line can contain one or more text strings and one stand definition can also have more than one *remarks* line if necessary (in this case it is enough that one of the lines is a match for the stand to be used).

- Text$_x$ Text string to look for in the flightplan remarks

NotRemarks

NOTREMARKS:Text$_1$:Text$_2$:Text$_3$:…

Same as above but limits the stand assignment to flightplans whose remarks section contains none of the specified text strings. When more than one *NotRemarks* line is used in a stand definition, the stand is used when even one of the lines is a match.

- Text$_x$ Text string to look for in the flightplan remarks

3.7 ICAO_Aircraft.json

This file contains more detailed information on the aircraft types. It is a JSON file containing an array of objects with the following keys:

	Key	Data type	Description
-	ICAO	string	Type designator (mandatory item)
-	Description	string	Three-character description ▪ First character – description: A (Amphibian), G (Gyrocopter), H (Helicopter), L (Landplane), S (Seaplane) or T (Tiltrotor) ▪ Second character – engine count: 1-8 or C (Two engines coupled to drive a single propeller system) ▪ Third character – engine type: E (Electric), J (Jet), P (Piston), R (Rocket) or T (Turboprop/turboshaft)
-	WTC	string	Wake turbulence category ▪ L, M, H or J
-	WTG	string	ICAO wake turbulence group ▪ A, B, C, D, E, F or G
-	RECAT-EU	string	RECAT-EU wake turbulence group ▪ A, B, C, D, E or F
-	Wingspan	number	Wingspan in meters
-	Length	number	Length in meters
-	Height	number	Height in meters
-	MTOW	number	Maximum take-off weight in kilograms
-	Use	string	Typical use(s) for the aircraft ▪ One or more of the following: A (Airliner/commuter), B (Business/corporate), C (Cargo), H (Helicopter, other than military), I (Military helicopter), M (Military, other than helicopter), P (Private), T (Military tanker/transport)
-	IATA	string	IATA designator
-	IATA_cargo	string	IATA designator when used as cargo aircraft
-	Manufacturer	string	Manufacturer name
-	Model	string	Aircraft model name(s) for this type designator

The “ICAO” key is the only mandatory one. Keys that are irrelevant or whose values are not known can be left out.

3.8 ICAO_Airlines.txt & ICAO_Airlines_Virtual.txt

These files contain radiotelephony callsigns. The latter is used for virtual airlines, and only designators not conflicting with ones found in “ICAO_Airlines.txt” are used from it. The following example line shows the syntax:

AAB	Abelag Aviation	ABG	Callsign definition

The format of the files is the same as in the “ICAO_Airlines.txt” file provided with EuroScope. Only one callsign must be defined per line.

Callsign definition

ThreeLetterID<tab>**OperatorName**<tab>**Callsign**

- ThreeLetterID — Three-letter designator in the flightplan
- OperatorName — Operator name – Notifying state
- Callsign — Radiotelephony callsign

As an alternative to having the callsign data, the file can contain the location of another file that has the data. In that case, this file should have only one line, containing the data file location. The location can be defined as absolute or relative. Relative locations starting with “.\” or “..\” are relative to the plugin folder. The data in the pointed file must be in the above format, and the pointed file must contain the callsign data, not a path to another file.

4 Data communication

The plugin uses scratchpad broadcasts (setting a specific text and then immediately after the original text) to communicate certain events. These are explained in this chapter. Note that if it becomes necessary, the messages or their formats may be changed without prior notice. The plugin documentation will then be updated to reflect the current message set.

4.1 Ground states

Like the default states, the custom states are communicated using scratchpad broadcasts. The scratchpad texts sent are as follows:

- No state (departure) NSTS (*
- On Freq ONFREQ
- De-Ice DE-ICE
- Start-Up STUP (*
- Line Up LINEUP
- Taxi In TXIN (*
- No state (arrival) NOSTATE
- Parked PARK (*

*) These states are default states in the latest EuroScope betas, and are implemented as custom states to work also with earlier EuroScope versions.

4.2 Stand assignments

Stand assignments are broadcast using scratchpad messages automatically when the assignment is made and later at specific points, or manually by controller action. The broadcasts are as follows:

- “GRP/S/<stand>”
  - Stand <stand> assigned for that aircraft at its destination airport
- “GRP/S/”
  - Previous stand assignment has been cancelled
- “GRP/M/<icao>/<stand>”
  - Stand <stand> at airport <icao> manually marked as occupied by that aircraft
- “GRP/M/”
  - Previous manual occupancy of a stand cancelled