Ground Radar plugin for EuroScope

- version 1.2.2 -

Developer Guide

EuroScope | power of control

Table of Contents

1 Installation

- In EuroScope, open the “OTHER SET” menu, then 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). Then 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”, then 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, select the "GRplugin" tag family provided in the package which hides the EuroScope tags, 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
SHOWLEADER:0
HISTORY_DOTS:0
TAGFAMILY:GRplugin
```

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 to tell the plugin information about the airport:

```
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:179
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:

```
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 External data files

This chapter gives guidance on developing the external 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).

3.1 GRpluginAircraftInfo.txt

This file contains information about aircraft types to improve the quality of the stand assignment feature. An example data line is shown below:

```
A225	88.4	84	18.2	600000	C
```

Each line must contain the aircraft type designator, wingspan (meters), fuselage length (meters), height (meters), maximum takeoff weight (kg) and a description (see below) separated by TAB characters. Enter 0 (zero) for unknown numeric values.

The description defines the typical use for the aircraft type to help with stand assignment. Enter one or more of the following characters:

- A (airliner/commuter)
- B (business/corporate aircraft)
- C (cargo aircraft)
- H (helicopter)
- I (military helicopter)
- M (military aircraft)
- P (private aircraft)
- T (military tanker/transport)

3.2 GRpluginCallsigns.txt

This file contains radiotelephony callsigns. 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. If the file is not found, the plugin checks if either “TopSkyCallsigns.txt” or “ICAO_Airlines.txt” are found in the plugin folder, and if so, uses the information in that file.

Callsign definition

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

- ThreeLetterID — Three-letter designator in the flightplan
- OperatorName — Name of the operator (not used, but must contain something)
- Callsign — Radiotelephony callsign

As an alternative to having the callsign data, this 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.

3.3 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 following example is used to show the syntax (optional lines in grey color):

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

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)

3.4 GRpluginMaps.txt

This file contains the definitions for the plugin drawn maps. The following example area is used to show the syntax (optional lines in grey color):

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
N059.38.14.252 E017.54.49.244	Coordinate
N059.39.58.802 E017.55.26.928	Coordinate
N059.39.59.055 E017.55.24.239	Coordinate
N059.38.14.503 E017.54.46.513	Coordinate

The mandatory items for each map are a name, a folder and a color with which to draw the items.

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 — Color's red component value (0-255)
- G — Color's green component value (0-255)
- B — Color's blue component value (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

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

- MapName Map name (text string)

Folder

FOLDER:FolderName

Every map must belong to a folder. 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.

Screen-specific

SCREEN-SPECIFIC

When a map contains this line, each radar screen will toggle the map's visibility independently instead of affecting all other radar screens. Setting this is only possible for maps with no *active* lines.

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.

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

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 single 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 single map.

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

Color

COLOR:ColorName
COLOR:ColorName:FillColorName

Every map 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 simply 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. All used 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

Style

STYLE:StyleName
STYLE:StyleName: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. The width is only necessary for Solid type lines (it has no effect on the other types). If a width is not defined, a 1-pixel width is drawn by default.

- StyleName Style to be used (Solid, Dash, Dot, DashDot or DashDotDot)
- Width Width of line (pixels)

Line

LINE:Lat_1:Lon_1:Lat_2:Lon_2
LINE:StartPointName:EndPointName

Draws a line from one point to another. Uses the previously defined line style (or solid line with 1-pixel width if no style defined).

- Lat_1 Latitude of starting point (decimal degrees or sector file format)
- Lon_1 Longitude of starting point (decimal degrees or sector file format)
- Lat_2 Latitude of end point (decimal degrees or sector file format)
- Lon_2 Longitude of end point (decimal degrees or sector file format)
- StartPointName Fix, VOR, NDB, airport code or runway (must be found in the active sector file)
- EndPointName Fix, VOR, NDB, airport code or runway (must be found in the active sector file)

Note: the syntax to define a runway threshold as a StartPointName or an EndPointName is the 4-letter ICAO airport designator followed by a forward slash and the runway identifier.

CoordType

COORDTYPE:AreaType:DrawType

The *CoordType* line is used to define the type of area drawn by the following *coordinate* lines.

- AreaType Type of area to be drawn
  - APM Area for the APM 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
  - TWYTYPE Area defined as a restricted taxiway for the RVM safety net
  - OTHER Other type (no safety nets involved)
- DrawType Drawing type for the area
  - POLYGON Area outline drawn
  - REGION Filled area drawn
  - NONE Area not drawn

The AREATYPE and TWYTYPE options require the restriction conditions to be specified using *restriction* lines (see below).

Restriction

RESTRICTION:Type:Value

The *restriction* line is used to set restrictions to AREATYPE and TWYTYPE areas whose coordinates are specified after the *restriction* line. 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, the previously set restrictions in the same map will be used for the polygons unless 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)
  - 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

Lat Lon
Lat:Lon
COORD:Lat:Lon

Defines a vertex point for a filled region or polygon. The region or polygon is drawn with the color(s) defined in a preceding *color* line. To draw more than one region or polygon in a map, another line type must appear between them (for example a *color* line).

- Lat — Latitude (decimal degrees or sector file format)
- Lon — Longitude (decimal degrees or sector file format)

Active

ACTIVE:1
ACTIVE:SchedStartDate:SchedEndDate:SchedWeekdays:StartTime:EndTime
ACTIVE:RWY:ARR:ArrRwyList:DEP:DepRwyList

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 first option will activate the map automatically without any time limits when the plugin is loaded. Note that this option cannot be used together with other *active* lines as it would override any other schedule.

The second option can be 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.

The third option activates the map based on active runways. If all the specified runways are active, the map is activated. If even one of them is not, the map will be deactivated. The runway identifiers must be in the format "<4-letter ICAO code><runwayID>", for example "EFHK15".

- ArrRwyList — Comma-separated list of runways. Enter "*" to allow any runway.
- DepRwyList — Comma-separated list of runways. Enter "*" to allow any runway.

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
  - “=” 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)

The resulting font size is limited to values between 1 and 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)

Symbol

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

Draws a predefined symbol on the screen. Optionally, can display a text label with the label's centerpoint offset a given number of pixels from the symbol centerpoint.

- SymbolName — Name of symbol
- Lat — Latitude of symbol centerpoint (decimal degrees or sector file format)
- Lon — Longitude of symbol centerpoint (decimal degrees or sector file format)
- PointName — Fix, VOR, NDB, airport code or runway (must be found in the active sector file)
- Label — Text label (text string)
- OffsetX — Number of pixels to offset the label centerpoint in the left(-)-right(+) direction
- OffsetY — Number of pixels to offset the label centerpoint in the up(-)-down(+) direction

Note: the syntax to define a runway threshold as a PointName is the 4-letter ICAO airport designator followed by a forward slash and the runway identifier.

Text

TEXT:Lat:Lon:Label
TEXT:Lat:Lon:Label:OffsetX:OffsetY
TEXT:PointName:Label
TEXT:PointName:Label:OffsetX:OffsetY

Draws a text label on the screen. Optionally, the label's centerpoint can be offset a given number of pixels from the given position.

- Lat — Latitude of label centerpoint (decimal degrees or sector file format)
- Lon — Longitude of label centerpoint (decimal degrees or sector file format)
- PointName — Fix, VOR, NDB, airport code or runway (must be found in the active sector file)
- Label — Text label (text string)
- OffsetX — Number of pixels to offset the label centerpoint in the left(-)-right(+) direction
- OffsetY — Number of pixels to offset the label centerpoint in the up(-)-down(+) direction

Note: the syntax to define a runway threshold as a PointName is the 4-letter ICAO airport designator followed by a forward slash and the runway identifier.

3.5 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 GRpluginAircraftInfo.txt).

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

3.6 GRpluginSettings.txt

This file allows adjusting the plugin settings. Each setting in the file 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 available settings, their default values and short descriptions containing the acceptable values are listed below.

3.6.1 Settings that must be system-wide

Setting name	Default value	Description
Equip_ProMode	0	Pro mode on(1)/off(0)
Equip_ModeS	WZLCISG	List of "mode S equipped" equipment code letters
Equip_NonAlt	XTDBMNYCVS	List of "non-altitude-reporting" equipment code letters
Aircraft_DefaultWTC		Default wake turbulence category when unknown (L, M, H or J to force a category or empty to leave as unknown)
Aircraft_DefaultUse_L	P	Default use for WTC category L aircraft when unknown (one character, see AircraftInfo data file, or empty to leave unknown)
Aircraft_DefaultUse_M	A	Default use for WTC category M aircraft when unknown (one character, see AircraftInfo data file, or empty to leave unknown)
Aircraft_DefaultUse_H	A	Default use for WTC category H aircraft when unknown (one character, see AircraftInfo data file, or empty to leave unknown)
Aircraft_DefaultUse_J	A	Default use for WTC category J aircraft when unknown (one character, see AircraftInfo data file, or empty to leave unknown)
System_UseReportedGS	1	Use reported(1)/calculated(0) ground speed
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_Unblock_GS	5	Groundspeed to unblock a manually blocked stand (1-999)
System_AutoAssignDist	100	Distance from destination [nm] to auto-assign a stand (1-999) Setting this value to zero disables the automatic assignment
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	BI,EB,ED,EE,EF,EH, EK,EL,EN,EP,ES,ET, EV,EY,GC,LE,LF,LG, LH,LI,LJ,LK,LM,LO, LP,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_Push	START UP	Text for "PUSH" ground state (ES default state)
System_State_Taxi	TAXI	Text for "TAXI" 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)

3.6.2 Settings that must be airport-specific

Setting name	Default value	Description
Airport_Elevation	0	Airport elevation in feet (-1000-30000)
Airport_Radius	0	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 ":" Can be entered either as decimal degrees or in the sector file format.

The airport elevation and radius can be defined also in the ASR file, but if defined, the values in the settings file override them. 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 to filter out traffic from the Approach Window.

Airport_Runway_End	from sct file	Runway end coordinates if different from sct file data. Runway id, lat and lon, all separated by ":" Coordinate formats are decimal degrees or the sector file format. This means the end of the runway 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_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 either one, 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 (0.0-999999.9), separated by ":"
Airport_Runway_MaxWingspan	999.9	Maximum wingspan for the runway

		Runway id and wingspan in meters (0.0-999.9), 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 the runway id specified. To limit just approaches or departures to a specific runway end, the following settings can be used instead:

Airport_Runway_MaxCode_Arr
Airport_Runway_MaxCode_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

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

Setting name	Default value	Description
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_APM	1	Area Penetration Monitoring on(1)/off(0)
System_CVM	1	Clearance Violation Monitoring on(1)/off(0)
System_ECM	1	Emergency Code Monitoring on(1)/off(0)
System_OSM	1	Occupied Stand Monitoring on(1)/off(0)
System_RIM	1	Runway Incursion Monitoring on(1)/off(0)
System_RVM	1	Restriction Violation Monitoring on(1)/off(0)
System_SRM	1	SID/Runway Monitoring on(1)/off(0)
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_Arrival	255,255,64	R,G,B value of the arrival flight color
Color_Background	0,0,0	R,G,B value of the APP window background 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_Departure	64,255,255	R,G,B value of the departure flight 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_Runway	160,160,160	R,G,B value of the APP window runway color
Color_RunwayExtension	80,80,80	R,G,B value of the APP window runway extension color
Color_Selected	255,255,255	R,G,B value of color for mouse-over flights
Color_SelectedLabelBorder	0,0,0	R,G,B value of edge color for mouse-over ground mode label
Color_SelectedLabelFill	255,255,255	R,G,B value of fill color for mouse-over ground mode label
Color_Text_Notes	255,255,255	R,G,B value of the text notes 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	160,160,160	R,G,B value of the Inbound window background color
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 Inbound window texts
Color_WForeground2	128,128,128	R,G,B value of the window titles and APP window scale
Color_WTitleBar	80,80,80	R,G,B value of the window title bar color
GroundLabel	1	Show(1)/hide(0) the label, track symbol, history and heading line in ground mode
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)
GroundLabel_ATYP	0	Show(1)/hide(0) ATYP in ground mode track label
GroundLabel_RMK	0	Show(1)/hide(0) RMK in ground mode track label
GroundLabel_SID	0	Show(1)/hide(0) SID in ground mode track label
GroundLabel_STAND	0	Show(1)/hide(0) STAND in ground mode track label
GroundLabel_WTC	0	Show(1)/hide(0) WTC in ground mode track label
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
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)
AppLabel_AFL_VS	0	Show(1)/hide(0) AFL+VS in APP window track label
AppLabel_GS	0	Show(1)/hide(0) GS in APP window track label
AppLabel_STAND	0	Show(1)/hide(0) STAND in APP window track label
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)
TowerLabel_AFL_VS	1	Show(1)/hide(0) AFL+VS in tower mode track label
TowerLabel_AFL_Digits	3	Altitudes below 10000ft displayed as Axx(2) or Axxx(3)
TowerLabel_GS	1	Show(1)/hide(0) GS in tower mode track label
TowerLabel_GS_Digits	3	GS displayed in tens of knots(2) or knots(3)
Track_PredictionLine_APP	0	APP window prediction line length in minutes (0-99)
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_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

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)
List_Alerts_Items	10	Number of flights to display in Alerts list (5-999)
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_ETA	1	Show(1)/hide(0) ETA in Arrival list
List_Arrival_STAND	1	Show(1)/hide(0) STAND 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_Departure_RWY	0	Show(1)/hide(0) RWY 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_EOBT	1	Show(1)/hide(0) EOBT in Departure list
List_Departure_STAND	1	Show(1)/hide(0) STAND in Departure list
List_Departure_Items	10	Number of flights to display in Departure list (5-999)
List_Inbound_TYPE	0	Show(1)/hide(0) TYPE in Inbound lists
List_Inbound_WTC	0	Show(1)/hide(0) WTC in Inbound lists
List_Inbound_ETA	0	Show(1)/hide(0) ETA in Inbound lists
List_Inbound_STAND	1	Show(1)/hide(0) STAND in Inbound lists
List_Stand_Items	10	Number of stands to display in Stands list (5-999)
Window_APP_Scale	10.0	APP window scale in pixels/nm (1.0-100.0)
Window_APP_Extensions	0	Show(1)/hide(0) runway extensions in APP window
Window_APP_Rotation	0.0	APP window view rotation in degrees (-360.0 to 360.0)
Window_APP_AltFilter	99900	APP window altitude filter in feet (0 to 99900)
RawVideo	1	Show(1)/hide(0) raw video data
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_Brightness	100	Raw video data brightness (0-100)
RawVideo_Afterglow	30	Raw video data afterglow (0-100)
RawVideo_MaxHistory	10	Raw video data maximum history positions displayed (0-10)

3.7 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).

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.

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 (decimal degrees or sector file format)
- Lon — Longitude of the stand area (decimal degrees or sector file format)
- 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 (decimal degrees or sector file format)
- Lon — Vertex longitude (decimal degrees or sector file format)

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

Causes the stand to be excluded 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.

Blocks

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

Causes the stand to block the specified stands from assignment when 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 (set -1 for wingspan if only a length limit is desired). The wingspan value can be set also based on the aerodrome reference code letter.

- StandList Comma-separated list of blocked stands
- WingspanOrCode Wingspan in meters or element 2 of the aerodrome reference code (A-F)
- Length Length in meters

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 AircraftInfo 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 GRpluginCargoCallsigns data file. In case the information is not found, the flight will be assumed to be an airliner.

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)
  - P (piston)
  - T (turboprop/turboshaft)
  - J (jet)
  - E (electric)

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 AircraftInfo data file. In case the information for a specific type is not found, a fixed value based on the wake turbulence category is used (L=14m, M=35m, H=64m, J=79m, unknown=0m).

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 AircraftInfo 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 AircraftInfo 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 AircraftInfo data file. In case the information for a specific type is not found, a value of 0 kilograms 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