

PPG

(ParkPosGenerator)

Manual



Version 9.5

February 20, 2019

Nico W. Kaan



Copyright © 2017-2019, Nico W. Kaan, Delft, Netherlands, All Rights reserved.

No part of this publication may be reproduced and/or published by print, photo print, microfilm or any other means without the previous written consent of Nico Kaan.

1	Introduction	3
1.1	System Requirements	3
1.2	Forum	3
2	Installation.....	4
2.1	Download and install.....	4
2.2	Update and removal.....	4
2.3	Version Check	4
3	Operations.....	5
3.1	Input files used (in folder data)	5
3.2	Define the paths to the folders with your airports (mandatory).....	5
3.3	Define the aircraft types per parking position type (optional)	5
3.4	Save and Restore buttons.....	6
3.5	Checkbox Generate options for parking positions in the xml files	6
3.6	Go! button	6
3.7	Output folder	6
3.8	Log.txt	6
3.9	Make the airport files accessible in PSXT!	6
4	Airport file format.....	7
4.1	Meta	7
4.2	Parameters	7
4.3	Parkpositions	8
4.3.1	parkpos	8

1 Introduction

Although PSXT (PSXseeconTraffic) comes with a folder **installation_airports** with **9916** airport files generated by **PPG** (ParkPosGenerator) for P3Dv4 without any add-on airports, you should generate new ones if you have FSX, or add-on airports, or the same add-on airports but from a different supplier.

1.1 System Requirements

PPG is a C++, **64** bit Windows application for **Windows 7** or later

You need the Microsoft Visual C++ Redistributables for Visual Studio **2015**. The standalone installer for x64 (vc_redist.x64.exe) can be [downloaded here at Microsoft](#).

1.2 Forum

For questions & answers and discussions use my [Support Forum at AVSIM](#).

2 Installation

2.1 Download and install

Download from [my website](#). Unzip and run the installer. You should install PPG at your PC running your Flight Simulator. You better not install it in the C:\Program Files folder.

After installation, you will have a **ParkPosGenerator** folder containing the application **ParkPosGenerator.exe** and two sub folders:

- The **data** folder is described in section 3.1.
- The **docs** folder contains the PPG_Manual and the Release Notes. The EULA is the End-User License Agreement. This is a legally enforceable contract between you and the Author. By accepting the installation you agree to be bound by the terms of this EULA.

If you run PSXT networked, it is recommended to let PPG generate the airport files over the network in a folder at the PC running PSXT. Use **\\PCname\Sharename** as Output folder (see section 3.7) where PCname is the name of the PC running PSXT and Sharename the name of the folder at that PC for the airport files (also see the [FAQ](#)).

2.2 Update and removal

Download the new version, unzip and run the installer. It will first uninstall your previous version and then install the new one.

Removing PPG from your PC can be done running the uninstaller from the Programs list or the apps list, just like any other Windows program.

2.3 Version Check

At each start-up, but not more than once a day, PPG will check if there is a new version available. If so the -normally- green version message in the main window will turn yellow, and the new version number will be indicated.

This is only an indication. It is up to you when you download and install the new version. It is recommend to always update.

It may also happen that you get the message "**error 99: This version has expired**". In that situation you cannot go further. You have to upgrade to a new version.

3 Operations

3.1 Input files used (in folder data)

In the folder **data** there are **22 text files** named by one letter. The letter represents the first letter of the ICAO code for an airport

Each file contains 1 line of ICAO airline codes separated by comma's. These airlines are the airlines that normally fly in the region of the airports starting with that letter.

Next to that there is a file **Cargo.txt** with cargo operators. The cargo operators are used to allocate to cargo parking positions.

These files are just a first shot; feel free to add/remove airlines.
(and save your work otherwise it will be overwritten by each new version of the PPG).

Next to these .txt files there are input files with all kinds of data: **wingspan.xml**, **aircraft.xml** and **airlines.xml**. The **IATAICAO.txt** file contains a translation from ICAO to IATA codes airports.

3.2 Define the paths to the folders with your airports (mandatory)

There are 10 text boxes for entering full paths to folders with airports, numbered (internally) from 1 .. 10.

You should enter full path names to the folder(-s) in which PPG will search for airports.



a path may not end with "\\Scenery" or "\\scenery".

Push the **change button**, specify a full path to a folder with add-on airport(-s) and push the change button again. If the folder exists the folders turns to grey and the **Go! button** will become enabled.

You can copy and paste text in the different folders. There is no need to work top down, you may leave "holes". The order is not important.

With the **x button** you can delete a folder. With the **S (swap) button** you can swap two adjacent entries.

A folder search includes all sub-folders, so specifying:

D:\Program Files\Lockheed Martin\Prepar3D v4

will make it search Prepar3D v4 completely. (The same for Microsoft Flight Simulator, of course)

If you have add-on airports outside the Prepar3D v4 folder you should add extra line(-s) pointing to these add-ons, for example:

D:\AddOns

The priorities are: a stock airport will not overwrite a non-stock airport. An ORBX airport will overwrite a stock airport. An add-on airport will overwrite a stock airport and/or an ORBX airport. So you do not have to worry about that.

PPG remembers what has been generated between runs of a session, so do not close PPG if you need more than 10 paths to folders!

3.3 Define the aircraft types per parking position type (optional)

There are 11 text boxes with pre-defined ICAO aircraft types for specific parking positions.

The aircraft types it generates for each position depends on the type of the parking position. There are 11 possible parking positions, for each an example number of ICAO aircraft types are already specified, but you can change that:

Push the **change button**, edit the ICAO aircraft types for a position. Specify types separated by comma's and push the change button again. Wrong types will be removed from your input.

PPG checks whether an aircraft type fits the radius of the parking position. If not, it is removed from the list (for that parking position). If there are no types left a default type that will fit is inserted.

3.4 Save and Restore buttons

When you push the Save button, the paths to your folders will be saved. When you push the Restore button, all your paths will be put back in the user interface (up to 10 of course) for editing and or selecting. This is just for convenience. Note that these buttons will only be enabled if there is anything new to save or restore.

3.5 Checkbox Generate options for parking positions in the xml files

This box is default checked. If you unselect it, the airport files will have info about parking positions only, no options for what aircraft and types can be placed at the parking position. This way you have a clean file that PSXT can add with real live data.

3.6 Go! button

If you push the Go! Button, the search for airports begins. The message line at the bottom shows the path from where PPG is searching.

After each path the number of airport files written to your Output folder is indicated together with the number of airport files that were written to your Output folder that were updates of airports already written from the same path (in the same run).

The status line shows the total number of airport files written and the total number of airport files that were updates of already written airports in this run.

3.7 Output folder

The <ICAO airport code>.xml files are generated in the Output folder, default sub folder **airports** of ParkPosGenerator.

You may change the Output folder to any folder that you have created first. Push the **change button** and type in the **Output folder text box** either a the name of a subfolder of PPG or a full path to another folder. It may be a full path at the same PC or at a networked PC (see section 2.1). Push the change button again to make it active.

If you want to start fresh, just manually empty the contents of the Output folder.

Note: Do not generate your airport files in PSXT's **installation_airports** folder while that folder may be overwritten with every new version of PSXT.

3.8 Log.txt

The file Log.txt contains a log of your run(-s).

3.9 Make the airport files accessible in PSXT!



Change the Airport files folder parameter in the PSXT UI (see PSXT Manual section 4.3.1) into a path to the Output folder of PPG. If PSXT and PPG are at the same PC, you can just push the **connect button!** (also see the [FAQ](#)).

4 Airport file format

Airport files are **xml** files. They are generated by PPG and read and updated by PSXT. You may make changes with an editor such as Notepad++, if you like.

First a simple example, the file **AGGA.xml**

```
<?xml version="1.0"?>
<meta>
  <info name="author" value="ParkPosGenerator" />
  <info name="version" value="9.2" />
  <info name="date" value="Thu Feb 7" />
</meta>

<parameters>
  <parameter name="ICAO" value="AGGA" />
  <parameter name="IATA" value="" />
  <parameter name="latitude" value="-8.702686" />
  <parameter name="longitude" value="160.681697" />
  <parameter name="airport_name" value="Auki" />
  <parameter name="supplier" value="P3D stock airport" />
  <parameter name="num_real_live_options" value="0" />
  <parameter name="num_real_live_parkpos" value="0" />
  <parameter name="parking_percentage" value="35" />
  <parameter name="last_update" value="" />
</parameters>

<parkpositions>
  <parkpos id="r1" latitude="-8.705341" longitude="160.682007" heading="98.8" radius="14.0">
    <aircraft>
      <option airlines="GIA,CPA,THY" types="GLF3" real="false" />
    </aircraft>
  </parkpos>
  <parkpos id="r2" latitude="-8.705578" longitude="160.681992" heading="98.8" radius="10.0">
    <aircraft>
      <option airlines="PVT" types="C172" real="false" />
    </aircraft>
  </parkpos>
  <parkpos id="r3" latitude="-8.705793" longitude="160.681946" heading="98.8" radius="10.0">
    <aircraft>
      <option airlines="PVT" types="C172" real="false" />
    </aircraft>
  </parkpos>
</parkpositions>
```

4.1 Meta

Obvious.

4.2 Parameters

The parameters, ICAO, latitude, longitude and airport_name are taken from the .BGL file in your Flight Simulator. The IATA code is added by PPG, if available. The supplier is deduced (if possible) by PPG from the path of the BGL file, while most of the time the suppliers name is part of the path. Stock airports are indicated by supplier=<P3D/FSX> stock airport.

Num_real_live options and num_real_live_parkpos are initially 0, they will be updated by PSXT when it adds real live data to the file.

The parking_percentage determines how many (*% of total*) of the parking positions will get an aircraft allocated by PSXT, the default value is 35. You can change that via the PSXT UI.

4.3 Parkpositions

There can be 0 or more parking positions (parkpos) in the file.

4.3.1 parkpos

Each parkpos has four attributes and n ($n \geq 0$) aircraft options. The attributes are latitude, longitude, heading and radius. The heading is in degrees and the radius is in meters.

4.3.1.1 aircraft options

An aircraft option has three attributes.

The **airlines** attribute is a text-string with either one ICAO airline code or more airline codes separated by comma's (and no spaces).

The **types** attribute is a text-string with either one ICAO aircraft type code or more aircraft type codes separated by comma's (and no spaces).

If there are more than one airline codes and/or more than one aircraft type codes PSXT will try randomly each possible combination to find a livery that matches the parking position.

The **real** attribute is a boolean indicating whether it was generated by the ParkPosGenerator (false) or by PSXT during a real live update (true).

A live aircraft option always has just one airline code and one aircraft type.

If it is a real live update there may be an **hours** attribute. The value is either one (utc) hour, or more hours separated by comma's. The hours indicates at what hour an aircraft of that airline and type was positioned or parked at the parking position. The hour is rounded such that 9:29 remains 9 while 9:30 becomes 10 and 24 becomes 0.

When a heavy aircraft parks, not only the current hour will be added to the option but also the next hour, assuming that a heavy aircraft normally stays at a gate more than an hour.

When non-heavy aircraft parks between 30 and 45 minutes after the hour nothing changes but outside that range the next hour will be added to the option too, assuming they have a turnaround time of about 40 minutes.

Finally a more advanced example, a snippet of the file EHAM.xml that was generated by PPG without aircraft options (see section 3.5), and was updated by PSXT with **real live** information:

```
<?xml version="1.0"?>
<meta>
  <info name="author" value="ParkPosGenerator" />
  <info name="version" value="7.2" />
  <info name="date" value="Fri Oct 26" />
</meta>

<parameters>
  <parameter name="ICAO" value="EHAM" />
  <parameter name="IATA" value="AMS" />
  <parameter name="latitude" value="52.308056" />
  <parameter name="longitude" value="4.764167" />
  <parameter name="airport_name" value="Schiphol" />
  <parameter name="supplier" value="Fly Tampa" />
  <parameter name="num_real_live_options" value="1495" />
  <parameter name="num_real_live_parkpos" value="160" />
  <parameter name="parking_percentage" value="30" />
  <parameter name="last_update" value="Thu Feb 7" />
</parameters>

<parkpositions>
  <parkpos id="A32" latitude="52.299538" longitude="4.753610" heading="237.8" radius="40.1">
    <aircraft>
      <option airlines="KLM" types="E75L" real="true" hours="7,11,12,13,15,16,17,18,19" />
    </aircraft>
  </parkpos>
  ...
  <parkpos id="E2" latitude="52.310280" longitude="4.766105" heading="304.0" radius="31.1">
    <aircraft>
      <option airlines="KLM" types="A333" real="true" hours="6,9" />
      <option airlines="DAL" types="B764" real="true" hours="5" />
      <option airlines="DAL" types="A333" real="true" hours="4,5,7,9,10,11,14,15,16,17" />
      <option airlines="KLM" types="B772" real="true" hours="5,6,20" />
      <option airlines="KLM" types="B789" real="true" hours="9,12,13" />
      <option airlines="DAL" types="B763" real="true" hours="12" />
      <option airlines="KLM" types="A332" real="true" hours="12,13" />
      <option airlines="CSN" types="A333" real="true" hours="12,13" />
    </aircraft>
  </parkpos>
  ...
</parkpositions>
```

This file has 1495 real live options (*in this snippet 9, one at A32 and 8 at E2*) and 160 real live parking positions (*in this snippet two: A32 and E2*).

We can see that the DAL B763 has visited gate E2 around noon (utc), while the DAL B764 was at gate E2 around 5 o'clock in the morning.