KMLLogger v 1.0 *by Mike Nixon 2026-01-03*

Intro

KMLLogger is a platform-independent **Lua** script for X-Plane 11 & 12 which runs in FlyWithLua.

It is based on my original script **MNFLightLogger** from 2024. If you are currently using that, then this is a drop-in replacement.

At the end of each flight it will create:

- A detailed record of your flight in **CSV** format, which can then be imported into **Excel**, **LibreOffice** etc. This shows dates, times, aircraft, departure & arrival airports, distance, flight duration, block times, fuel usage, altitudes flown etc.
- A record of your flight track in **KML** format, which can be opened in **Google Earth.** This will show your horizontal and vertical profile in 3D, along with any waypoints you pass (Airports, VOR, NDB, Fixes etc). At each waypoint you will be able to see exact status of your flight: ground speed, height agl, altitude, QNH etc). This function is **optional**: you will be prompted at the beginning of your flight whether you want to log KML or not.

Requirements

- You will need to install **FlyWithLua:** a simple, free, and extremely useful plugin.
- Some functions will require **Java.** A Java JRE is easily installed on all platorms

Installation

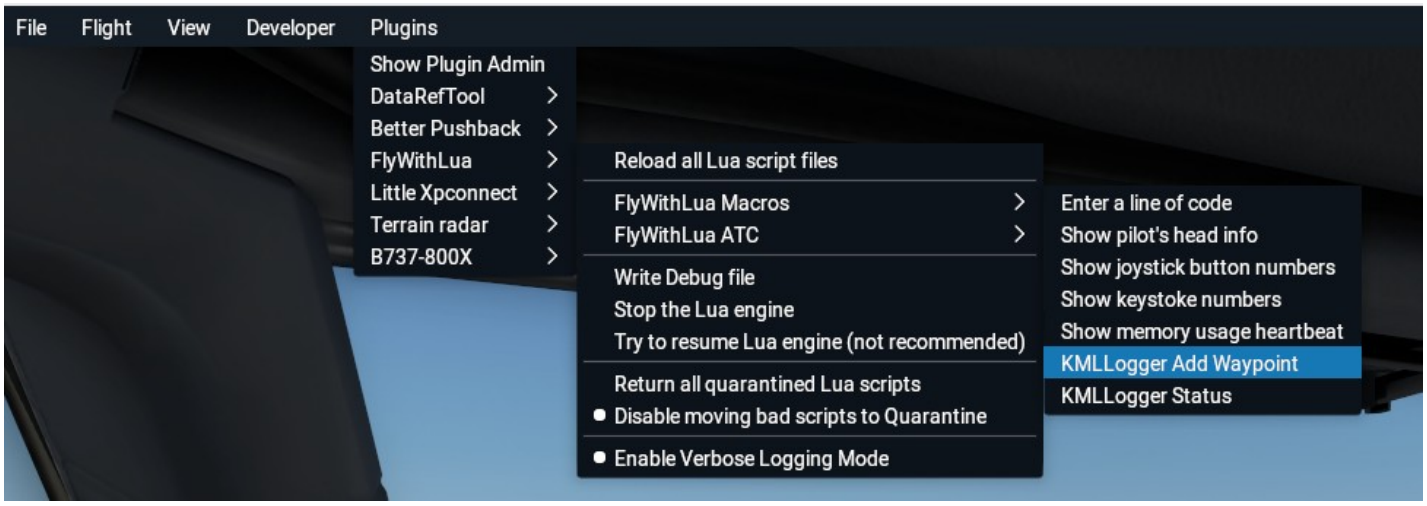
Simply unzip the KMLLogger script into your **/Resources/plugins/FlyWithLua/Scripts** inside your main X-Plane folder.

Setup

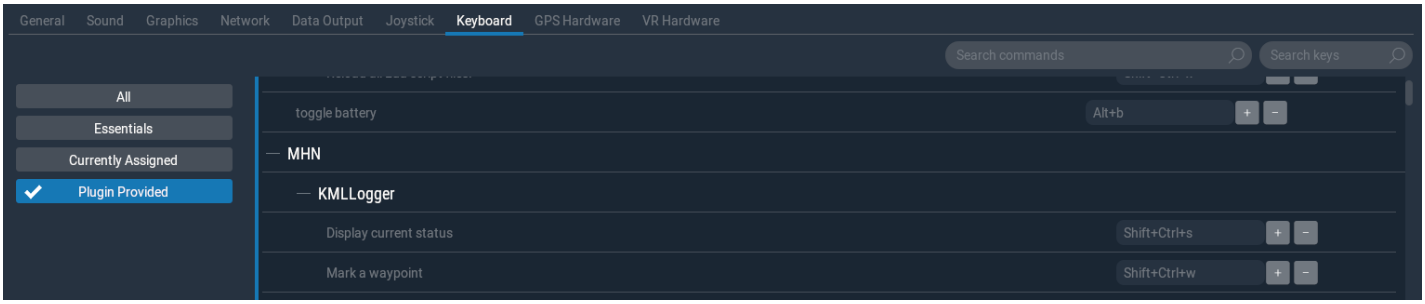
- No setup is actually required. When the script starts, it will create a **KML** folder in your main X-Plane folder: your KML flight records will be saved in here.
- To install FlyWithLua, follow one of these links:
    - https://www.x-plained.com/flywithlua-for-x-plane-12/
    - https://forums.x-plane.org/files/file/82888-flywithlua-ng-next-generation-plus-edition-for-x-plane-12-win-lin-mac/
    - https://forums.x-plane.org/forums/topic/71006-howto-first-steps-using-flywithlua/
- There are 2 **macros,** and you can assign keystrokes to the 2 functions:
    - Show a status window (current flight details). Closes automatically after 20secs
    - Show a window in which you can **manually add a waypoint**

Images

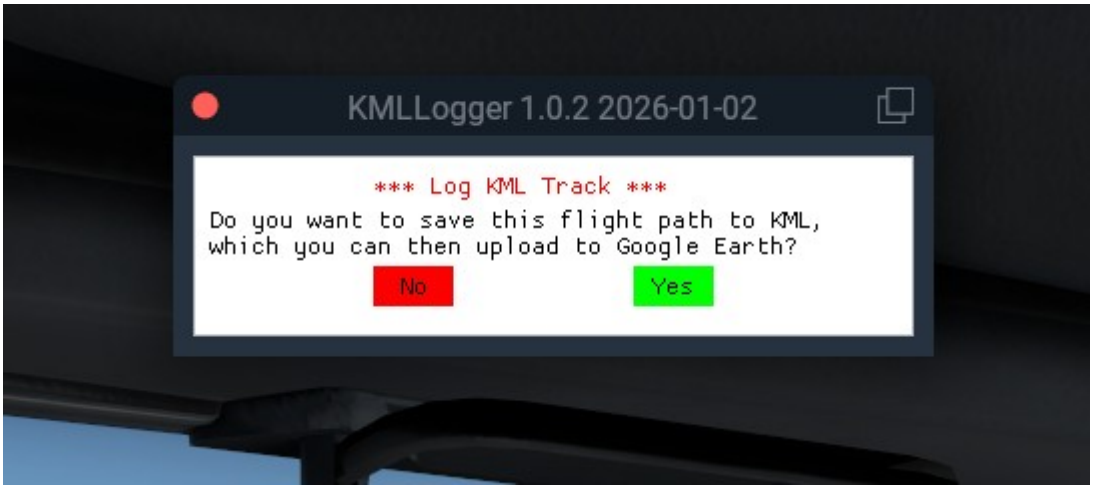
Macros



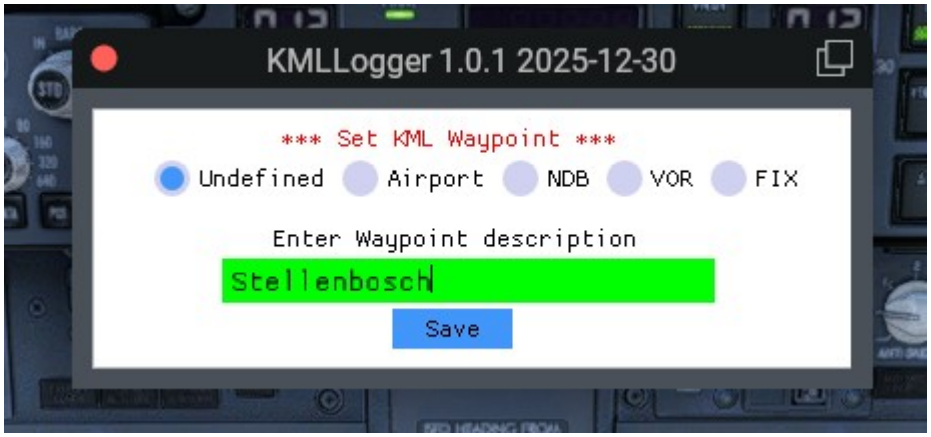
Keyboard Shortcuts



Flight Start

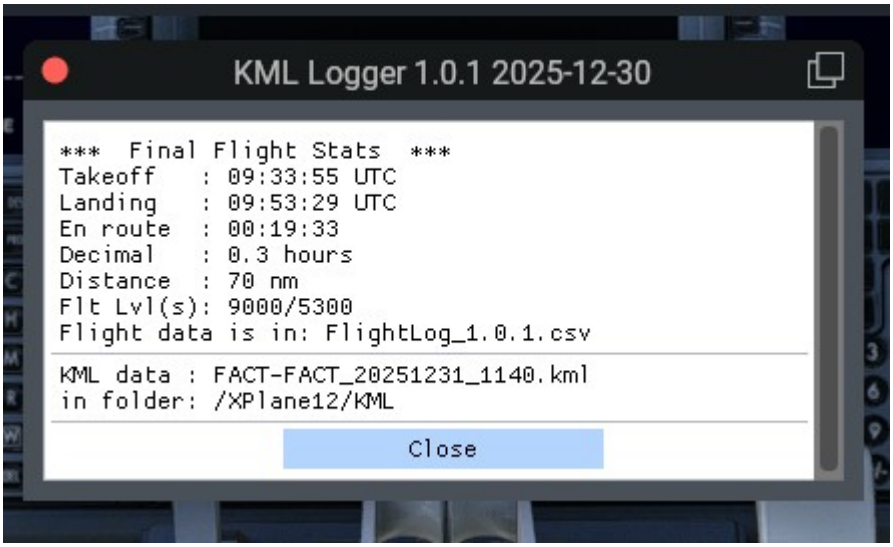


Manually adding a Waypoint



Flight Status

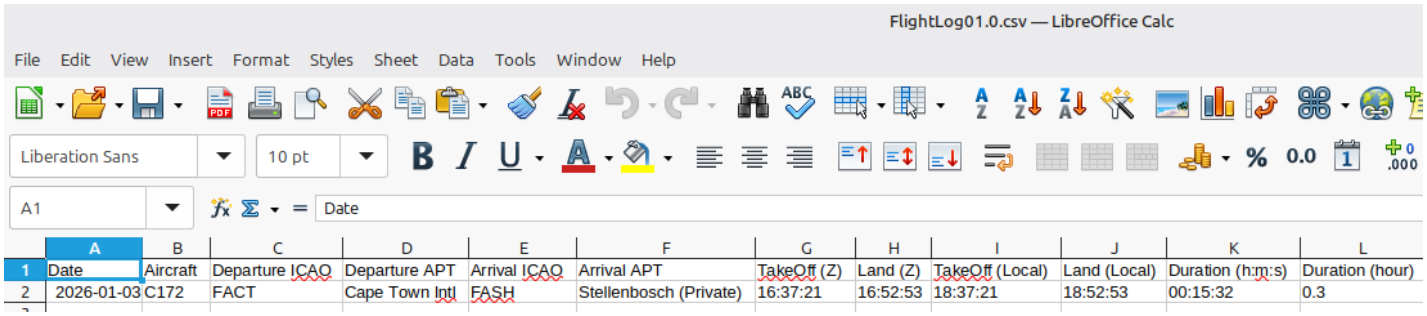
Shut-down



CSV Logfile

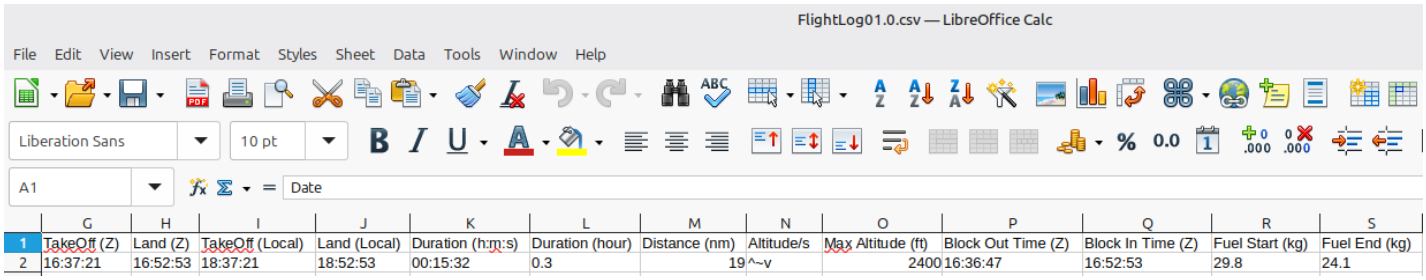
```
mike@Mike-nvme:/XPlane12$ cat FlightLog01.0.csv
Date,Aircraft,Departure ICAO,Departure APT,Arrival ICAO,Arrival APT,TakeOff (Z),Land (Z),TakeOff (Local),Land (Local),Duration (h:m:
s),Duration (hour),Distance (nm),Altitude/s,Max Altitude (ft),Block Out Time (Z),Block In Time (Z),Fuel Start (kg),Fuel End (kg)
2026-01-03,C172,FACT,Cape Town Intl,FASH,Stellenbosch (Private),16:37:21,16:52:53,18:37:21,18:52:53,00:15:32,0.3,19,^~v,2400,16:36:4
7,16:52:53,29.8,24.1
mike@Mike-nvme:/XPlane12$
```

Spreadsheet (left)



	A	B	C	D	E	F	G	H	I	J	K	L
1	Date	Aircraft	Departure ICAO	Departure APT	Arrival ICAO	Arrival APT	TakeOff (Z)	Land (Z)	TakeOff (Local)	Land (Local)	Duration (h:m:s)	Duration (hour)
2	2026-01-03	C172	FACT	Cape Town Intl	FASH	Stellenbosch (Private)	16:37:21	16:52:53	18:37:21	18:52:53	00:15:32	0.3

Spreadsheet (right)

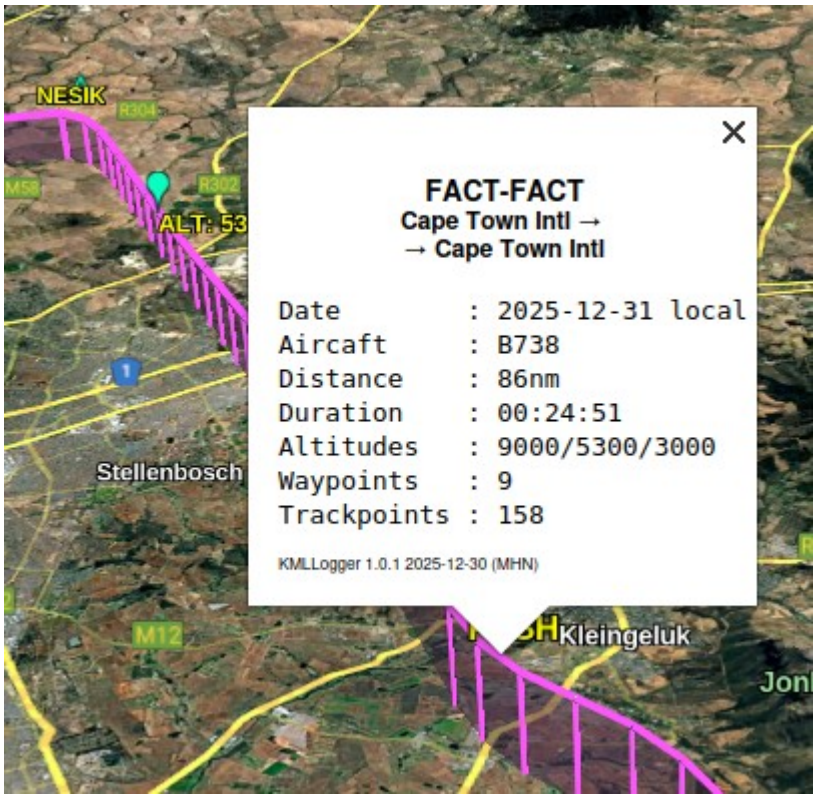


	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	TakeOff (Z)	Land (Z)	TakeOff (Local)	Land (Local)	Duration (h:m:s)	Duration (hour)	Distance (nm)	Altitude/s	Max Altitude (ft)	Block Out Time (Z)	Block In Time (Z)	Fuel Start (kg)	Fuel End (kg)
2	16:37:21	16:52:53	18:37:21	18:52:53	00:15:32	0.3	19	^~v	2400	16:36:47	16:52:53	29.8	24.1

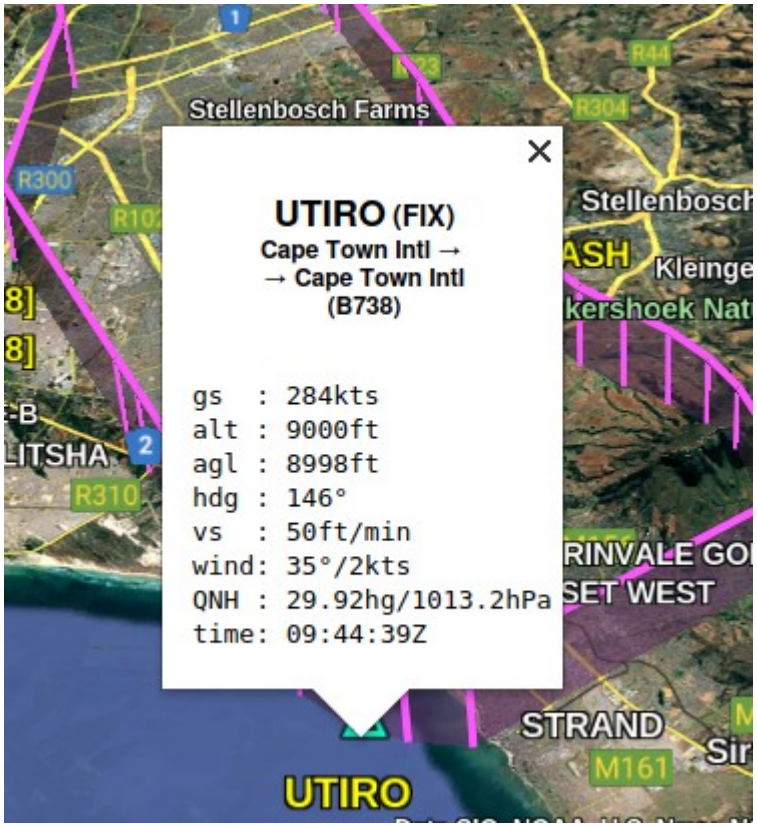
Google Earth

A short “scenic” flight here, starting and ending at Capetown International FACT.

Google Earth Track/Summary



Google Earth Waypoint



Technical Stuff

- With current settings, I did not detect any framerate drop when the script is running
- **Below** 600m agl (+- 2000ft), the script will log status **every 10 secs**
- **Above** 600m agl, the script will log **once per minute**
- With autopilot **on,** a flightplan loaded, and flight director set to follow the track, the script will log each successive waypoint as it is passed. For **ZiboMod** variants, as long as the script will log each planned waypoint as long as the flight director is **on.** Waypoints logged are:
  - Airport
  - VOR
  - Localizer
  - Fix
  - DME
- With autopilot **off,** the script will search for and log any **nearby** waypoints matching:
  - Airport
  - VOR
  - Localizer
  - Fix
- Certain waypoints can cause confusion, particularly runways, eg: "**RW19".** Consequently, these are **not** logged.
- The **csv** logfile will be updated provided the length of the flight **exceeds 10 mins.**
- Statistical logging:
  - Starts with **any engine on**
  - Ends with **all engines off**
  - I have ignored the state of the **parkbrakes**, as some simmers may start their flights with engines **already** running
  - With autopilot **on** when aircraft altitude = altitude dialled into flight director or...
  - With autopilot **off** provided altitude remains constant for **5 mins or more**, then flight altitude will be logged
  - At end of fight, all altitudes logged in descending order of time spent at each altitude
  - If your altitude has not remained constant, then it will be logged as: "**^~v**" *(shorthand for "climb", "variable altitudes" and "descent")*
- At **startup,** the script will **disable** the "**use system time**" option. This means that you can speed up X-Plane as much as you like, but **all times will be correct** relative to each other.

- The script uses states to determine its behaviour: on ground before, aloft after takeoff, back on ground. So if you reload the script mid-flight, it will not function as designed. At the start of a flight, you can reload the script, but once you have taken off, reloading FlyWithLua scripts will mean you will lose all the data gathered by the script thus far.
- **QNH.** If this is set incorrectly, all that will happen is that your altitudes logged will be out, and even if flying with autopilot with dialled-in altitude, this may not be logged.
- **Altitudes** are all rounded to nearest 100ft. Normally +-50ft is the accepted norm for altitude hold as I understand, but purely because this is a sim, and X-Plane 12 can be quite pernickety compared to X-Plane 11, I went with 100ft. This mostly for VFR pilots

Filenames

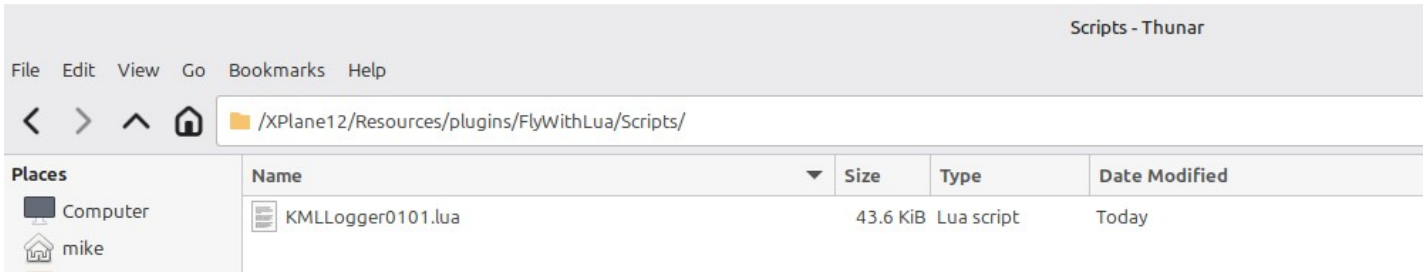
- The script is named **KMLLogger***n.n.n***lua** Where *n.n.n* represent the full version number. Subsequent updates will be named accordingly. When I post an update, simply download it and **replace** the previous version with the new one.
- **KML** files are named by departure, destination airport icaos, plus date. Eg:
    - FACT-FAFK_20251102_0749.kml
    - FACT-FIMP_20251105_0744.kml
    - FAOR-EDDF_20260102_1914.kml
- **CSV** logfile is named **FlightLog_***n.n***.csv.** Where *n.n* represent the main and sub version numbers of the script. The reason for this is that there may be changes to the csv layout from one version to another

Testing

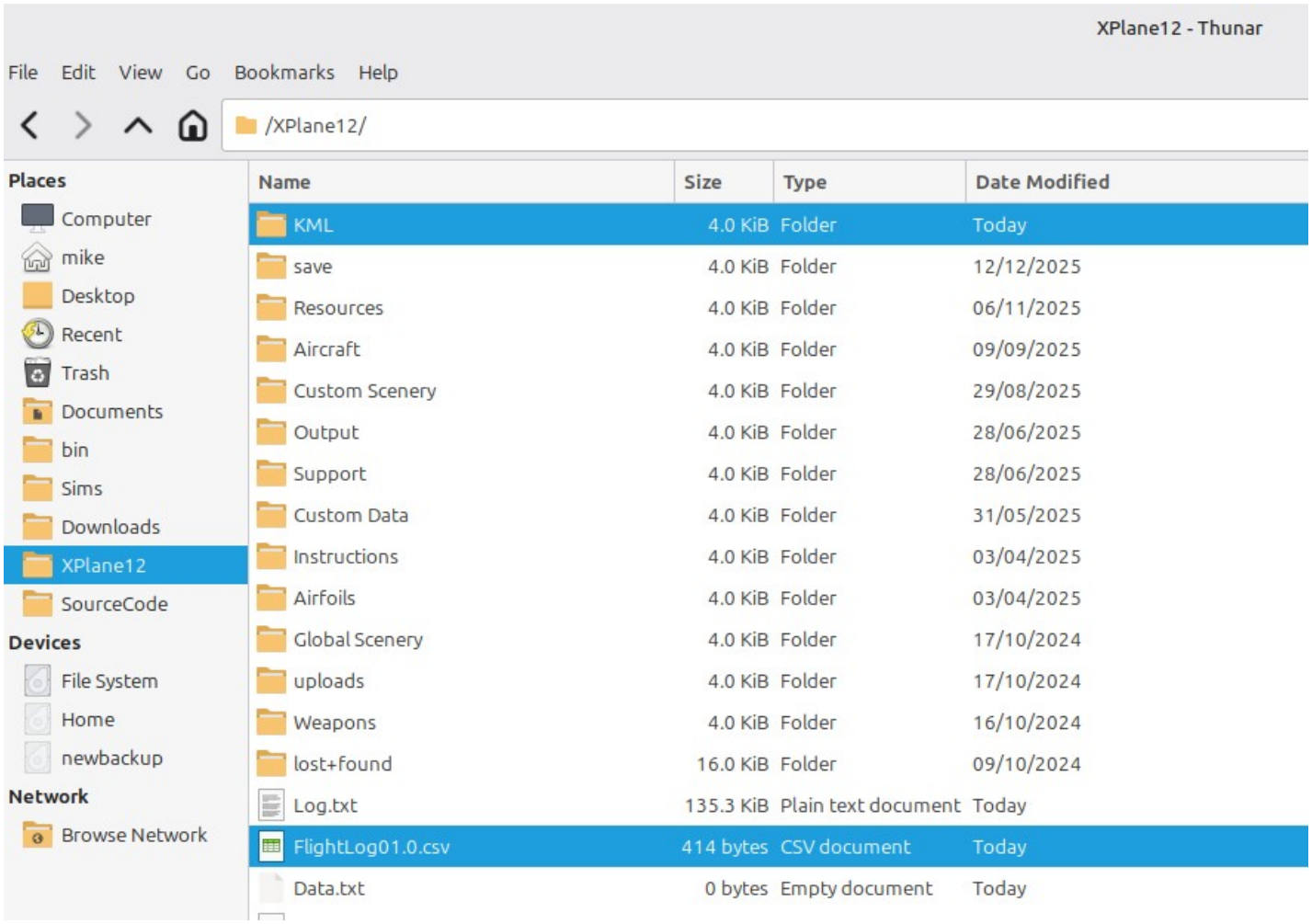
- As a pensioner, my finances are somewhat constrained, so I am unable to afford any of the really big commercial aircraft that I see on the market! I will rely on you the user to alert me to any anomalies that may arise when flying any of these. My script does rely on “vanilla” datarefs, so should work with pretty much any aircraft.
- The 2 main aircraft types are those that use *sim/....* based datarefs, and those like Zibomod which use *laminar/B738* based datarefs. My script copes with both types
- I use Linux (Linux Mint Cinnamon 22.2), but as FlyWithLua, and Lua are both platform-independent, my script should work on Windows or Mac.
- X-Plane 11: C172, MD80, Piaggio P180, Laminar B737, Zibomod 737-800X, Pilatus PC12
- X-Plane12: C172. Laminar B737, Zibomod 737-800X, Cirrus SR22, Airbus A333

A deeper dive

Script Location:



CSV Log file etc:



Tracing problems

If things go wrong, your first port of call is the X-Plane log file: **Log.txt**

This is in your main X-Plane folder, and is created by the sim itself, and contains very extensive logging data.

My script also writes to this file, so have a look through Log.txt to see if any errors are logged.

All entries will start with the word: **KMLLogger**

Extract from Log.txt showing only entries made by my script:

```
FlyWithLua Info: Start loading script file
/Sims/X-Plane11/Resources/plugins/FlyWithLua/Scripts/KMLLogger_260101.lua

KMLLogger1.0.2[13:11:55Z]: XPlane verson: 11550 Aircraft: C172 [1 engine/s].  Zibomod
aircraft? false

KMLLogger1.0.2[13:11:55Z]: script initialized.

FlyWithLua Info: Finished loading script file
/Sims/X-Plane11/Resources/plugins/FlyWithLua/Scripts/KMLLogger_260101.lua

DataRefToolDebug: Found 36 unique possible datarefs in file
"/Sims/X-Plane11/Resources/plugins/FlyWithLua/Scripts/KMLLogger_260101.lua"

KMLLogger1.0.2[11:37:23Z]: saveWPT FACT: Cape Town Intl [C172] ?true

KMLLogger1.0.2[11:37:23Z]: KML logging ENABLED

FlyWithLua Info: Start loading script file
/Sims/X-Plane11/Resources/plugins/FlyWithLua/Scripts/KMLLogger_260101.lua

KMLLogger1.0.2[11:38:18Z]: XPlane verson: 11550 Aircraft: C172 [1 engine/s].  Zibomod
aircraft? false

KMLLogger1.0.2[11:38:18Z]: script initialized.

FlyWithLua Info: Finished loading script file
/Sims/X-Plane11/Resources/plugins/FlyWithLua/Scripts/KMLLogger_260101.lua

KMLLogger1.0.2[11:38:26Z]: saveWPT FACT: Cape Town Intl [C172] ?true

KMLLogger1.0.2[11:38:27Z]: KML logging ENABLED

KMLLogger1.0.2[11:39:31Z]: **Takeoff**

KMLLogger1.0.2[11:40:53Z]: ==> Next waypoint: MD01 type: 512 dist: 1 prev dist: 0.3

KMLLogger1.0.2[11:41:23Z]: ==> Next waypoint: CT602 type: 512 dist: 1.2 prev dist: 1

KMLLogger1.0.2[11:41:23Z]: Passed VFR Wpt: MD01 DISTANCE: 1 Type: 512

KMLLogger1.0.2[11:48:24Z]: **Landed**
```

Credits

A special thanks go out to wahltho, and many other helpful people on the various X-Plane forums.

Thank you for taking the time to answer my questions.

Licence

You are completely free to improve or edit my script, or use parts of it.

All I ask is that if you do redistribute my script under a different name, you at least credit me as the original author.

Author:

Mike Nixon

mikenixon427@gmail.com

2026-01-03