P3D4 TextMenu displays via WideClient

Note that these facilities need FSUIPC 5.13 or later and WideClient 7.145 or later. (WideClient 7.148 for certain improvements as noted).

This package contains 7 small Lua plug-ins for WideClient which are provided to handle the various types of display possible via the event.textmenu function.

There are two different installation possibilities. They both rely on all the used Lua plug-ins being pre-loaded, for instance by an [Auto] section in your FSUIPC's INI file.

The easier and probably more reliable installation just uses AllTexts.lua. This is a combination of of all six other plug-ins, giving a more responsive and coordinated result if you want all or most of the functions of the others. In fact I would recommend you use it even just for one of them. The others may still be of interest as they may be easier to understand and modify should you choose.

The separate ones, described below, can be used when only one or two of the facilities are required, or when you are using different PCs or displays for each type.

The sizes and positions of the displays are set by parameters at the top of the AllTexts and ShowCycle plugins. Change those to suit your needs. You may also want to change the trigger method, way you select each display when they are sharing the one display area (as designed into them).

The rest of this document describes the displays as if you were using the separate plug-ins, but the descriptions also do apply to the combined one, AllTexts.

The different display types (bearing in mind that they are *all* also handled by AllTexts) are:

SimConnect Text display, handled by ShowSimCText.lua
SimConnect Window display, handled by ShowSimCWin.lua
SimConnect Menu display, handled by ShowMenu.lua
FSUIPC's filtered ActiveSky weather display, handled by ShowWthr.lua
A WideClient text file display, handled by ShowFile.lua

There's a ShowCycle.lua provided too. Whilst this doesn't directly make any displays, it is essential to manage the others, and provides the user a way of setting screen positions and sizes in the one place instead of in the 5 separate display plug-ins. You don't need to install any of the others you don't want to use.

The screen positions are defined by these lines in ShowCycle.lua:

```
ipc.set("TextLeft", 0)
ipc.set("TextTop1", 0)
ipc.set("TextTop2", 130)
ipc.set("TextWidth", 1024)
```

```
ipc.set("TextHeight1", 130)
ipc.set("TextHeight2", 638)
```

This suits a small screen of 1024 x 768 resolution. Modify them as you wish. The Tops and Heights are doubled as the small SimConnect Text window is kept separate from the four switchable ones. The small one is #1 and the larger, #2, is below using the rest of the screen.

Of course, these are only ready-to-use examples, and you are completely free to modify them as you wish, or of course write your own.

The SimConnect displays

The first three are standard SimConnect display facilities, and currently the first two types can be eliminated from the P3D screen via a User option in the P3D Settings. The menu cannot be removed from that screen in current versions of P3D.

The ActiveSky weather display

The ActiveSky weather data facility depends upon a plan being loaded into P3D or ActiveSky. In the P3D case, you enable an option in ActiveSky to automatically load it. In either case, when this happens, ActiveSky creates a text file which contains a full weather briefing for the flight.

You need to tell FSUIPC5 where it can find this file. This is by adding a [General] section parameter as follows:

    ASNplanWx=C:\Users\<UserName>\AppData\Roaming\Hifi\AS_P3Dv4\Weather\activeflightplanwx.txt

The plan is sent to WideClient, via the "TextMenu" event system, when you activate an additional FSUIPC control which can be assigned to any button on any PC in the WideFS+FSUIPC network. The control is "ASNweatherBroadcast" (number 1143).

FSUIPC reduces the displayed output to a summary suitable for providing the essential weather information to your FMS.

Text File display facilities

This option is completely local to WideClient. There are two parameters for the [User] section of WideClient.INI – here are examples:

        TextFileforDisplay=\\EXTRAS\Pilot2ATC\ConversationText.txt
        MaxTextRead = 3084

These are just my own parameters. The text file is the conversation text file produced by recent versions of Pilot2ATC. I have shared the path where I have this file placed as "Pilot2ATC" and access it from that PC, "EXTRAS". The "MaxTextRead" option is simply a way of making the updating of this file, as it is added to, more efficient by only reading the last so many characters in the file (3084 is more than will fit on my Window, but I added some just in case).

As the file is updated, so will the display, scrolling as necessary. Thus it will always show the most recent event, very useful for ongoing ATC interaction and also for live log files, etc.

The ShowFile.lua example provided colour codes the paragraphs it receives from Pilot2ATC according to a prefix, nicely added by Pilot2ATC. The colours may not be the same as that program uses on its own display, but can be changed as you need.

For unprefixed paragraphs a default, different, colour is used.

With WideClient version 7.148 or later, any occurrence of numbers in word form, and letters in phonetic alphabet form, are converted automatically to numeric digits and letters, and the numbers are closed up (no spaces in frequencies and squawk codes, for example)

. If you don't want this to happen you can turn it off, individually, using these parameters:

    TextFileNumbers=No
    TextFilePhonetics=No

Coordinating the displays

One other Lua plug-in is provided: "ShowCycle.lua". This enables 4 of the above displays (SimCWin, Menu, Wthr and File) to share one window on your screen. The simpler SimCText needs are set to occupy a permanent site, but much smaller space, and ShowCycle does nothing with that except supply the window position and size.

A key or button can be assigned to cycle between the main four, which are implemented to use the same window space on screen. This key or button must be assigned to toggle bit 0 (togglebits value 01) in Byte offset 66E0. Obviously this trigger can be changed by you editing that Lua file. The coordination between the displays is done by Lua globals. Offset 66E0 is used in all the files included except SimCText, and will need to be changed if you are already using this for something else.

Note that only the Menu display takes priority. If there is no other displays active then the next one will activate, but a Menu will stay displayed until cleared. Really it should be cleared when answered or closed, but P3D4 versions before 4.3 do not send notification of this. So for P3D4.2 I have provided another FSUIPC assignable control to remove it manually: RemoteTextMenuToggle (number 1144). For P3D version 4.3 you will need FSUIPC version 5.131b or later to take advantage of automatic clearance.

You only need to use the Lua files you want for your displays. If you use different displays or windows for the different things, you don't need ShowCycle but you will need to set your own window positions and sizes, and remove all the ipc.get lines which fetch globals preset by ShowCycle. Also remove the event and togglebits functions for offset 0x66E0.

And of course you can change the colouring to suit your own needs.

I use all six displays in my cockpit, so of course I use the single, combined, plu-in, AllTexts!