Speech Recognition on a Remote Computer

The purpose of this tutorial is to show you how to use speech recognition over a network using a registered version of FSUIPC and WideFS. The tutorial is based on using the Windows XP operating system and using FS9.

One of the problems in running a home cockpit with just one projector is that it is difficult to have situational awareness when taxiing around an airport. My set-up is an A320 which uses a replica side stick controller. There is therefore no provision on the stick to use an 8-way POV switch to pan around the environment.

The initial solution I used was to assign buttons on the pedestal which, when pressed, selected a pre-assigned view e.g. forward left, forward right etc. Whilst it works well, the problem is that finding your assigned taxi route or gate can mean a heads down situation. The result can lead to a loss of control in taxiing and not ending up where you intended. What I wanted was a quick and instantaneous heads up way of changing my view. The answer is speech recognition.

Mulling over the use of speech recognition, my initial thought was that the problems of using the technology would significantly outweigh the benefits. Furthermore, the microphone I have is not connected to the computer running FS9 but is connected to a separate computer running, amongst other things, SquawkBox.

My two concerns were therefore:

1) Is the technology robust enough to allow error free use?
2) Can the technology be used over a network?

I am pleased to say that the answer to both questions is a resounding 'yes'.

Before I go through the actual steps of installing and using the software, an overview of how the system works is worth a mention. The idea is to transmit a press of a virtual joystick button, through voice activation, across the network via WideFS. This press of the virtual joystick button is recognised by FSUIPC and converted into a key press of its own. The key press is chosen by the user to initiate an action in FS9 e.g. to change a particular view. That is the basics of how the system works.

I have depicted this on the next page.

How to use Voice Recognition over a Network
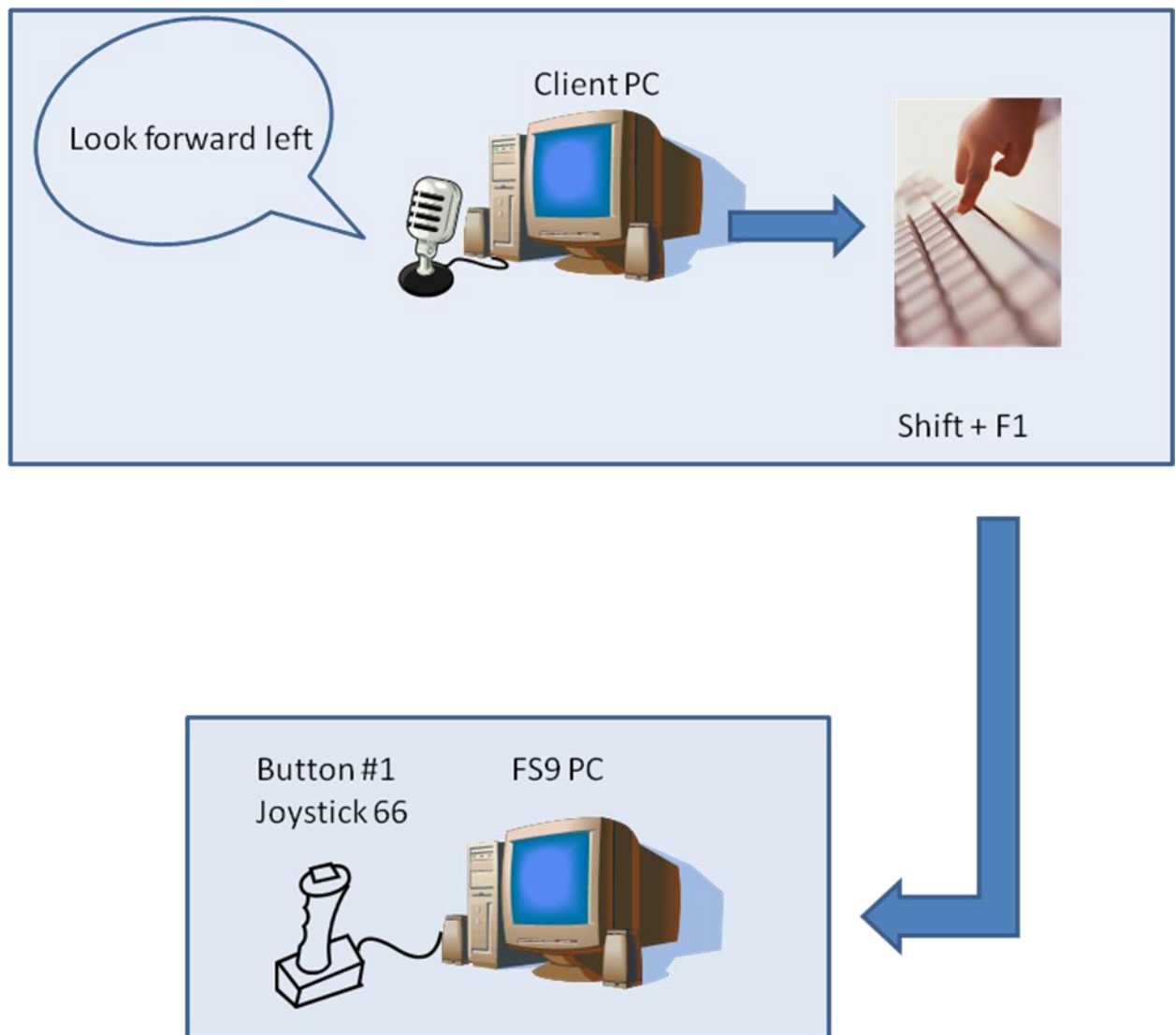
Step 1: Finding and Installing Speech Recognition Software

There are a number of free programmes available on the Internet and so the choice is yours. What is important is that the software needs to be able to translate a spoken phrase into a key press. The one I chose was PiLfIuS! v0.9 Beta. It's free, flexible and very easy to use.

Before using PiLflus! you need to download and install Microsoft's Speech SDK 5.1. The file you should download and install is 'SpeechSDK51.exe' – 68.0MB in size.

Both programmes are installed on your client computer i.e. the non-FS computer which the microphone is connected to.

Step 2: Setting up Virtual Joystick Buttons

After installing the software, you now need to think about assigning keys on the client computer which will be recognised as virtual joystick buttons by FSUIPC. The important thing to consider is that the keys (or combination of keys) should not conflict with programmes that are already running on that computer.

Let us suppose that I want to assign the key combination, Shift + F1, as a virtual joystick button to send across the network.

I need to edit the WideClient.ini file to tell it that if the key press combination of Shift + F1 occurs, it will treat it as a virtual joystick button and 'send it' to FSUIPC to process on the computer running FS9.

The way to do this is explained very well in the Adobe Acrobat document, *WideFS Technical* which is included in the current WideFS 6.78 download. See the section called, "*ButtonKeys: making use of FSUIPC's virtual buttons facilities*" on pages 16, 17 and 18 (document rev. 6th November 2008).

Read these three pages before moving on!

The WideClient.ini file will now have a section looking something like this:

[ButtonKeys]
T65=112,9
T66= 113,11
T67 = 80,8

+---+
| **Explanation** |
| |
| T65=112,9 |
| |
| T65 = Joystick 66, button #1 |
| 112 = F1 |
| 9 = Shift |
| |
| When Shift + F1 are pressed on the keyboard, WideClient will 'send' |
| the keyboard combination to FSUIPC on the FS9 computer which will |
| treat it as a joystick button being pressed (Joystick 66, button #1). |
| |
| T67=80,8 |
| |
| T67 = Joystick 66, button #3 |
| 80 = P |
| 8 = no shift keys pressed |
| |
| When P is pressed on the keyboard, WideClient will 'send' the |
| keyboard press to FSUIPC on the FS9 computer which will treat it as |
| a joystick button being pressed (Joystick 66, button #3). |
+---+

I have started at T65 only because I have virtual joystick buttons assigned elsewhere in my setup.

Step 3: Configuring Speech Recognition Software

The purpose of this exercise is rather than pressing Shift + F1 ourselves on the client computer, we will let PiLfIuS! do it for us upon a given spoken command.

When you run PiLfIuS! for the first time, you will be asked to go through a series of stages whereby the software learns how you speak. Once this is completed, you are able to assign a keyboard press (or a combination of presses) to be activated by a spoken command.

The spoken command that you use is entirely up to you. It does not have to be one word. That is the beauty of it. By being able to use more than one word, you can ensure the command is not a phrase that you would ordinarily use during the course of the flight. This is important so as not to become frustrated with the software carrying out unintended commands. By using carefully constructed phrases, I have been able to fly online, using Squawkbox and Teamspeak, without the software mistakenly thinking that I have asked it to carry out a key press.
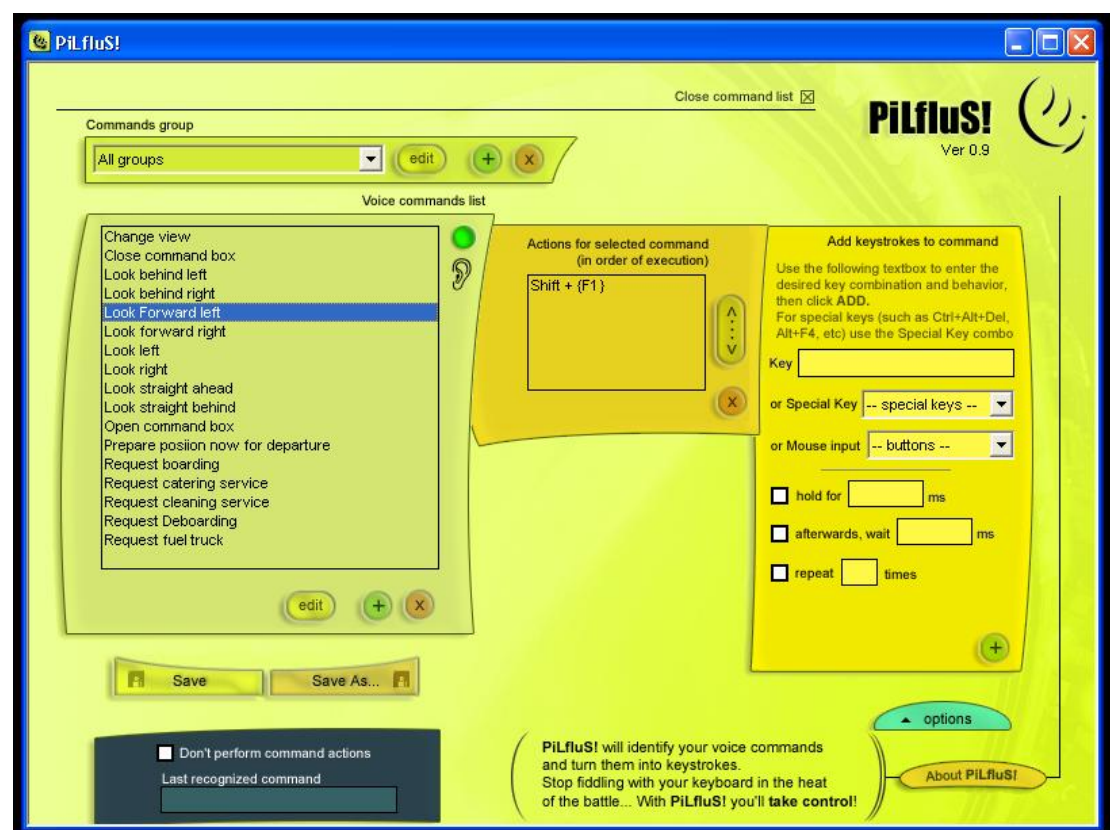
Figure 1 PiLfluS! Main Screen

Figure 1 above shows the main screen of PiLfluS! In the left hand box are the commands that I have created (to add a command, click on the green + button). The highlighted command is 'Look forward left' to which I have assigned the key press, Shift + F1. This is shown in the middle box. The box on the right is where you assign

4

the key press to the command. That's it! Now just minimise the programme. There are options to toggle the programme to listen for commands or to turn it off with a keyboard command. I will let you explore that for yourself.

Recap: How it Works

    i)   When I say, 'look forward left', PiLfluS! 'presses' the Shift + F1 key combination.

    ii)   In the WideClient.ini file, I have told WideClient to treat the key press combination of Shift + F1 as a press of a button on a virtual joystick (T65=112,9).

    iii)   FSUIPC on the FS9 computer will see that Joystick 66 button #1 has been pressed over the network.

    iv)   We now need to tell FSUIPC what to do when this button is pressed.

Step 4: Assigning the Virtual Joystick button to FSUIPC

1) You now need to tell FSUIPC what to do with this press of the joystick button. Start your flight simulator and in the FSUIPC menu, select the '*Buttons and Switches*' tab.

2) On your client computer, with WideClient running, press Shift + F1 (or with PiLfluS!, talk into your microphone the verbal command you have assigned to Shift + F1).

3) You can see in the screenshot below that FSUIPC has received the pressed button of the virtual joystick 66 (button #1). All you need to do now is to assign that joystick command to whatever action you want. In this case I have chosen it to trigger '*View Forward Left*' from the drop down FS control menu. You could of course simply select a key press by selecting the '*Select for key press*' option on the left hand side of the dialogue box.
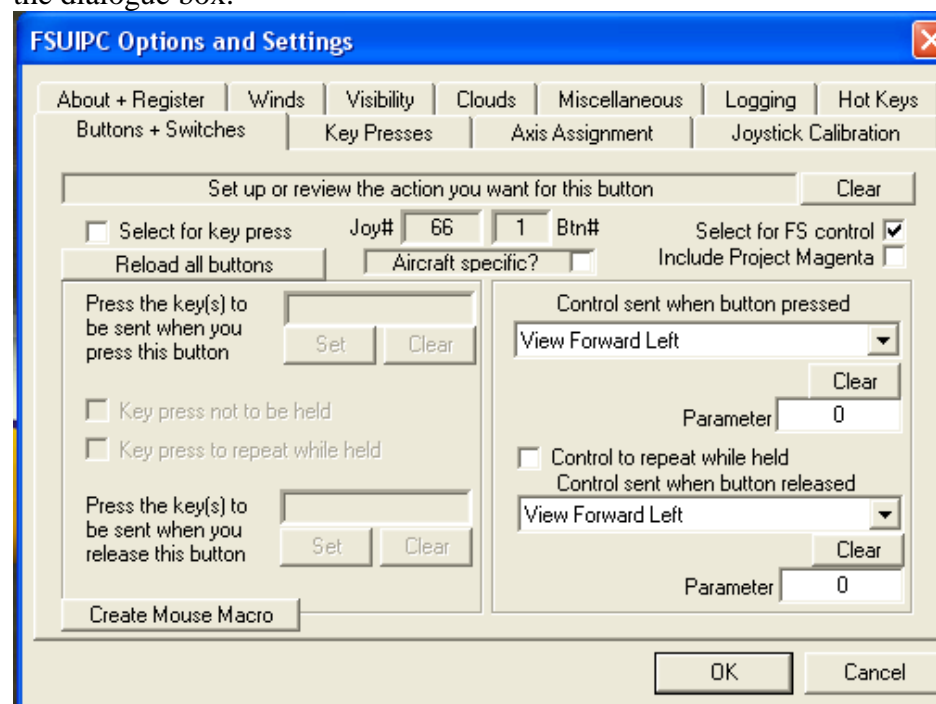
NOTE

On receiving Shift + F1, FSUIPC would interpret this as "Press" on the first press, and a "Release" on the second press, and so on, alternately.

In other words, you would have to speak the command twice before an action is taken. That is why I have told FSUIPC to carry out the command '*View Forward Left'* again (*Control sent when button released*) once the button is released (which happens the next time you send the (spoken) command.

In this way, the next time the virtual button #1 of joystick 66 is received (Shift + F1), it will carry out the command, '*View Forward Left'* because it is interpreting the command as a button release.

This is I suppose a rather cumbersome way of doing it. Instead, and perhaps more tidily, you can do it in the WideClient.ini file.

Instead of using the Tn=<keycode>,<shifts> format (e.g. T65=112,9) you can instead use the following formats in the [ButtonKeys] section of the Wideclint.in file :

Pn=<keycode>,<shifts> to Press the virtual button
Rn=<keycode>,<shifts> to Release the virtual button

That's it!

As you can see, there is enormous potential to use virtual joystick buttons across the network. The only limitation is your imagination!

I hope this tutorial has given you an insight into the finer details of what lurks beneath WideFS's bonnet.

Darren Sugden
December 2010