

Contents

Contents	1
Introduction.....	2
Who This Guide Is For	2
Caveats	2
Install Cygwin.....	2
Download FlightGear And The Dependent Packages	6
OpenAL.....	6
OpenAL For FlightGear 0.9.8.....	6
OpenAL For FlightGear 0.9- CVS	6
Plib 1.8.4 Source	6
SimGear 0.3.8 Source	6
SimGear 0.3- CVS Source.....	6
FlightGear 0.9.8 Source	6
FlightGear 0.9- CVS Source	6
FlightGear 0.9.8 Base Package	6
FlightGear 0.9- CVS Base Package	6
Store The Packages In ~/tar.....	7
Build FlightGear Release 0.9.8	8
Build Plib 1.8.4 Source	8
Extract OpenAL.....	9
Build SimGear 0.3.8 Source	9
Build FlightGear 0.9.8 Source	10
Extract FlightGear 0.9.8 Base Package.....	11
Execute FlightGear 0.9.8	11
Build FlightGear 0.9 CVS (First Time).....	12
Build Plib 1.8.4 Source	12
Extract OpenAL.....	12
Get SimGear 0.3 CVS Source	13
Build SimGear 0.3 CVS Source	14
Get FlightGear 0.9 CVS Source.....	14
Build FlightGear 0.9 CVS Source	15
Get FlightGear 0.9 CVS Base Package.....	15
Execute FlightGear 0.9 CVS	15
Build FlightGear 0.9 CVS (Subsequent Times).....	15
Update SimGear 0.3 CVS Source	15
Build Updated SimGear 0.3 CVS Source.....	16
Update FlightGear 0.9 CVS Source.....	16
Build Updated FlightGear 0.9 CVS Source	16
Update FlightGear 0.9 CVS Base Package.....	16
Execute FlightGear 0.9 CVS	16
Summary.....	17
Install Cygwin.....	17
Additional Downloads	17
Build Plib.....	17
Get And Build SimGear	17
Get And Build FlightGear.....	17
Get Base Package.....	18
Execute	18
Update And Build SimGear	18
Update And Build FlightGear	18
Update Base Package	18
Execute	18

Introduction

Having discovered and installed the pre-packaged Windows FlightGear binary it's easy to get hooked on what FlightGear has to offer. It quickly becomes obvious that FlightGear is a very active project and that the best way to get the most from the rapid development is to build the latest code. That's when it's easy to give up; it's not easy trawling through the forums for the information needed to build a working version of FlightGear on Win32 systems.

That's why this guide was created.

Who This Guide Is For

You might find this guide useful if:

- You want to build the latest version of FlightGear
- You want to test the latest version on Win32 and report problems
- You want to contribute to the development by building your own code
- You just want to say "hey, I can build FlightGear!"

Caveats

This guide assumes some, but not much, familiarity with software systems.

This guide does not assume any knowledge about Unix. Phew! Most Win32 users will regard Unix as the most unfriendly operating system ever invented. This isn't true; it's not that Unix is unfriendly – it just chooses its friends carefully.

This guide will undoubtedly become out of date. The screen shots were taken in October 2005.

Install Cygwin

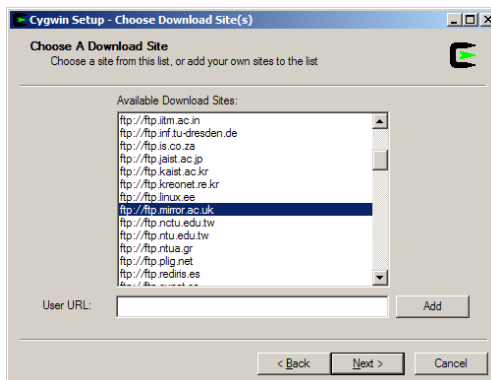
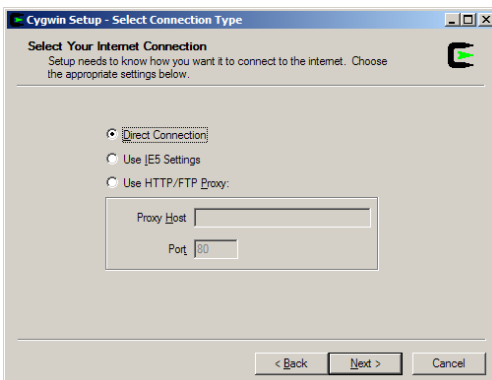
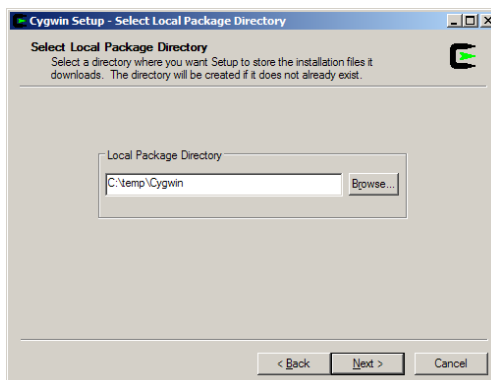
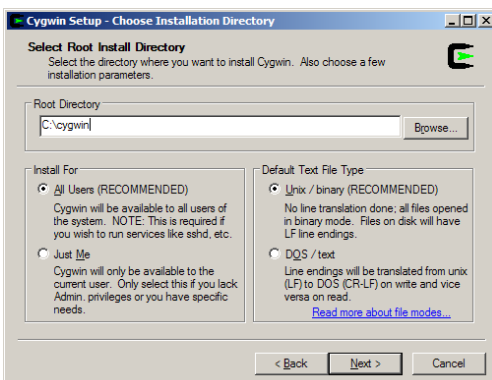
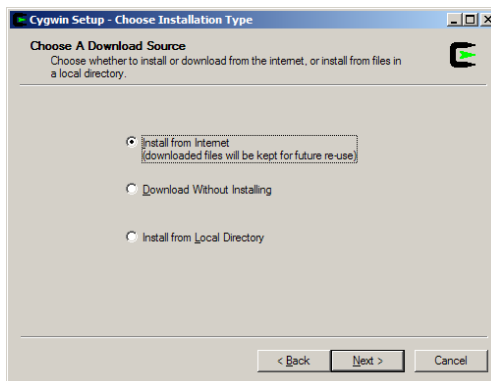
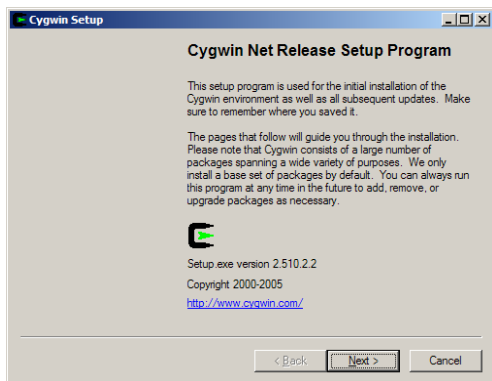
This guide is not meant to be a comprehensive Cygwin tutorial. If you've never before used Cygwin then this section should offer some useful information. It's important to recognise that Cygwin, like FlightGear, is an active project and so the version numbers of Cygwin packages may differ to those illustrated here.

This section describes enough of the Cygwin environment to allow you to build both the stable version of FlightGear (version 0.9.8) and the development version of FlightGear (version 0.9-CVS).

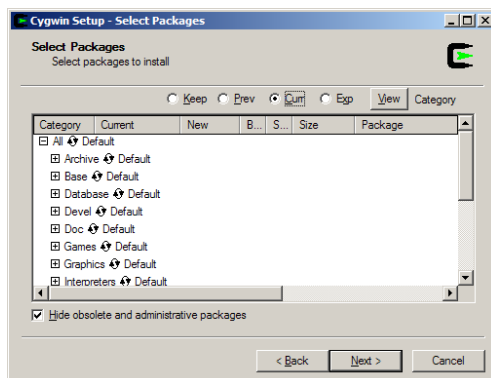
To download and install Cygwin go to <http://www.cygwin.com> then find the "Install Cygwin Now" link and save setup.exe.



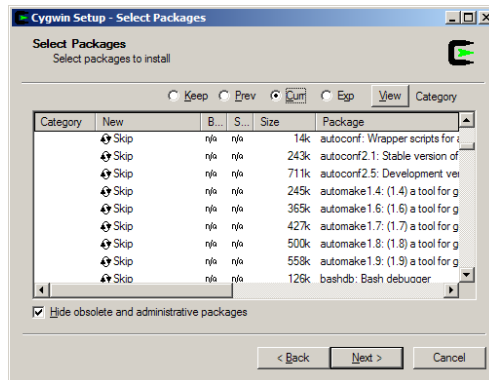
Run setup.exe and follow the dialogue boxes.



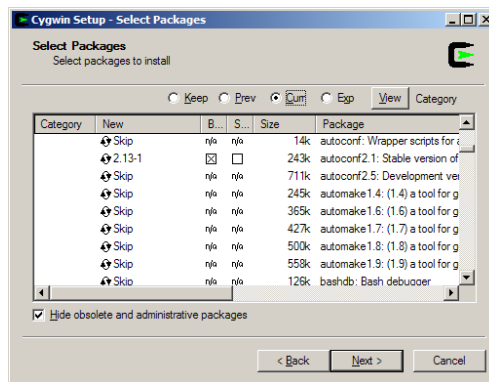
The following dialogue box will eventually be displayed. Use this dialogue to choose which Cygwin packages to install.



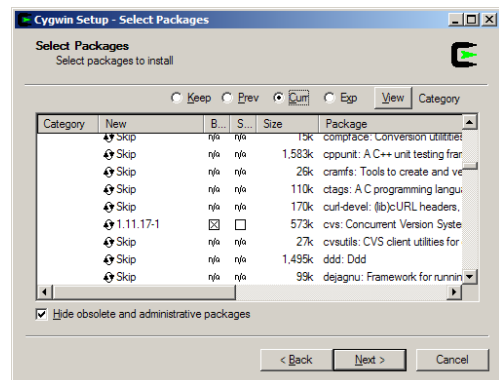
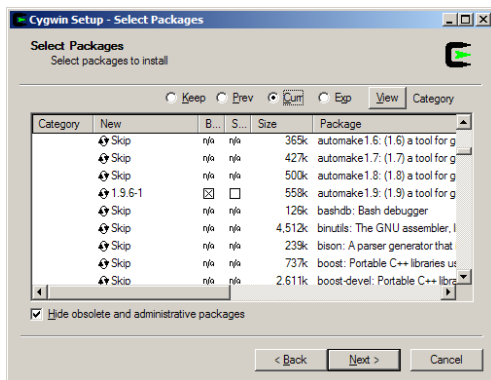
When setup is run for the first time a default set of “base” packages are already selected. To build FlightGear you need an additional set of development packages. Expand the “Devel” node then find the “autoconf” and “automake” packages.



Click on “Skip” next to the “autoconf2.1” package. The version number of autoconf is displayed in the “New” column. A marked checkbox appears in the “Binary” column and a cleared checkbox appears in the “Source” column; you’re selecting to use a pre-compiled binary of autoconf rather than building autoconf from source.

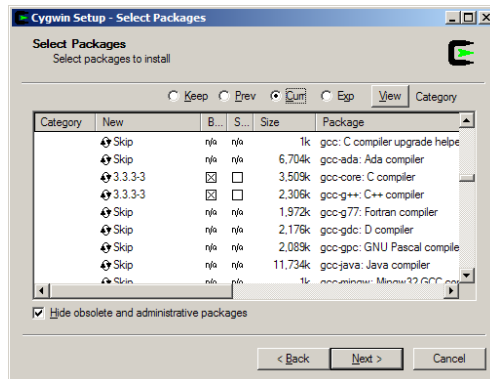


Select the most recent “automake” package and the “cvs” package.

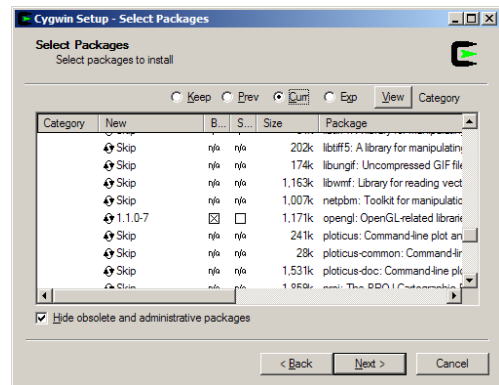
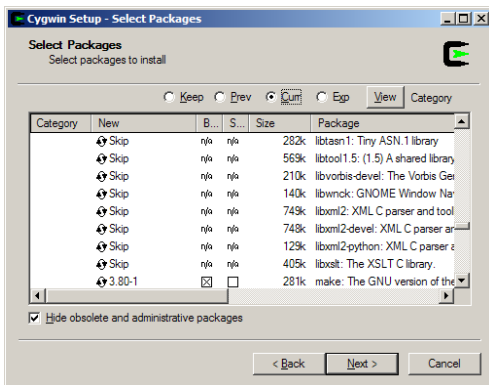


Select the “gcc-core” and “gcc-g++” compiler packages.

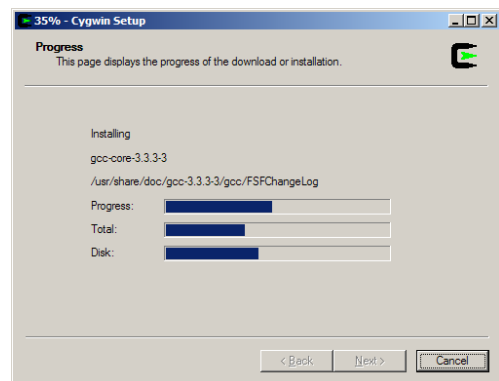
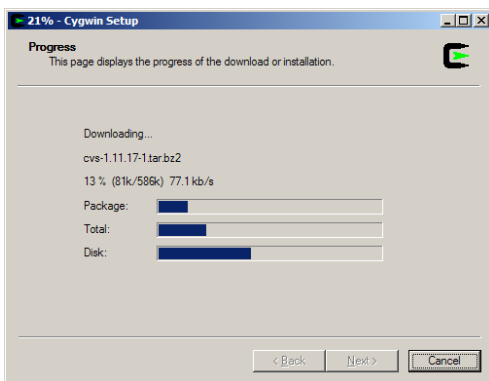
Important Note: This guide starts by building and installing the stable version of FlightGear (version 0.9.8). This version does not compile with gcc 3.4.4-4. Instead click on the version number in the Cygwin setup until it displays 3.3.3-3 as shown in the picture.



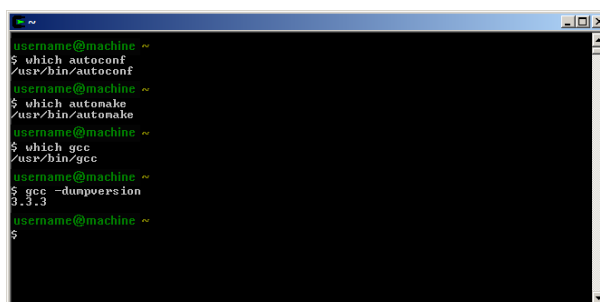
Select the “make” package. Finally expand the Graphics node and select the “OpenGL” package.



Click Next to begin the download and install process.



Start Cygwin. You can perform a very quick test that the required packages have been installed by using the Unix “which” command. You can also check that the correct version of gcc has been installed. The following screenshot illustrates this.



The remainder of this document uses the following style to represent the Cygwin window. The following commands use “which” to check that the remaining packages are correctly installed.

```
username@machine ~  
$ which tar
```

```
username@machine ~  
$ which cvs
```

```
username@machine ~  
$ which make
```

Download FlightGear And The Dependent Packages

FlightGear cannot be compiled in isolation; it relies on a few other packages. This section describes where to find those packages, which files to download and where to save them.

OpenAL

OpenAL is the hardest package to deal with under Cygwin. When this document was written there was no support to build OpenAL in Cygwin from source. Some very kind people (notably Norman Vine) have already addressed this so that you can simply install some pre-compiled OpenAL Cygwin libraries. The problem here is that you need one pre-compiled version of OpenAL to build FlightGear 0.9.8 and a different pre-compiled version of OpenAL to build FlightGear version 0.9-CVS. Take care when installing the OpenAL libraries and make sure you're using the correct pre-compiled library.

OpenAL For FlightGear 0.9.8

The pre-compiled files needed to build FlightGear 0.9.8 can be found by following this link. Store the file somewhere on your PC. Using this file is described later.

ftp://ftp.ihg.uni-duisburg.de/FlightGear/Win32/openal_cyg.tgz

OpenAL For FlightGear 0.9-CVS

The pre-compiled files needed to build FlightGear 0.9-CVS can be found by following this link. Store the file somewhere on your PC. Using this file is described later.

http://www.vso.cape.com/~nhv/files/cygwin/cyg_openAL.tgz

Plib 1.8.4 Source

Go to <http://plib.sourceforge.net>. Follow the links to save plib-1.8.4-tar.gz to your PC.

SimGear 0.3.8 Source

Go to <http://www.simgear.org>. Follow the links to save SimGear-0.3.8-tar.gz to your PC.

SimGear 0.3-CVS Source

This is described in a later section.

FlightGear 0.9.8 Source

Go to <http://www.flightgear.org>. Follow the links to save FlightGear-0.9.8-tar.gz to your PC.

FlightGear 0.9-CVS Source

This is described in a later section.

FlightGear 0.9.8 Base Package

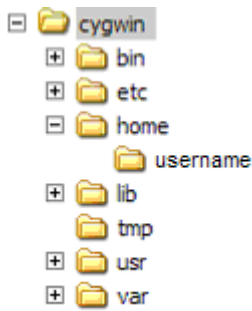
Go to <http://www.flightgear.org>. Follow the links to save fgfs-base-0.9.8-tar.bz2 to your PC.

FlightGear 0.9-CVS Base Package

This is described in a later section.

Store The Packages In ~/tar

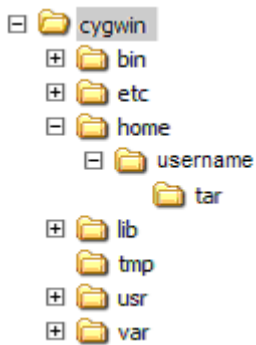
If you're new to Unix then the title will look confusing. What is "~/tar"? Use Windows Explorer to look at the Cygwin tree.



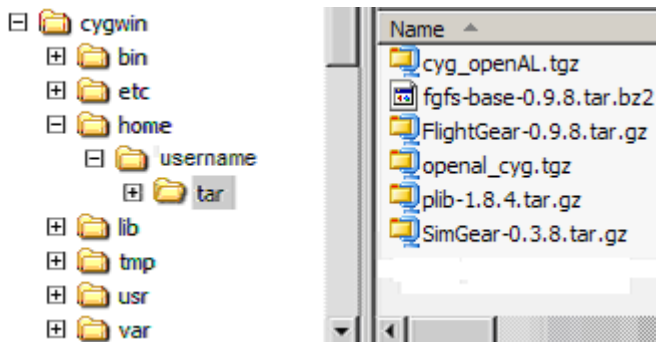
Now start Cygwin. Cygwin starts in your home directory called "~". Create a directory called "tar" as illustrated:

```
username@machine ~  
$ mkdir tar
```

Now refresh Windows Explorer to see where the directory was created.



The Unix path "~/tar" maps to the Windows folder you've just created. Copy the OpenAL, plib, SimGear and FlightGear files into this tar directory.



Build FlightGear Release 0.9.8

The following section describes how to build the released version of FlightGear 0.9.8. Each command is listed but the output of each command has been omitted. In general you should ensure that each command reports no errors before typing the next command. If you choose to use a different directory structure then commands must be changed accordingly.

Build Plib 1.8.4 Source

```
username@machine <any_path>
$ cd ~

username@machine ~
$ tar -xzf ~/tar/plib-1.8.4.tar.gz

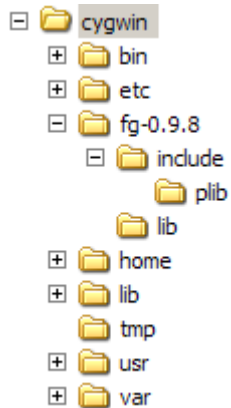
username@machine ~
$ cd plib-1.8.4

username@machine ~/plib-1.8.4
$ ./configure --prefix=/fg-0.9.8
```

```
username@machine ~/plib-1.8.4
$ make
```

```
username@machine ~/plib-1.8.4
$ make install
```

The directory tree now looks like this. Note how “/fg-0.9.8” has been created with some sub-directories containing the results of running “make install”.



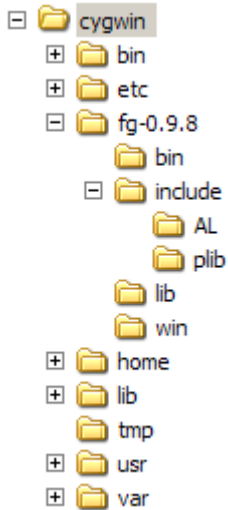
Extract OpenAL

The procedure for OpenAL with FlightGear 0.9.8 is different to the procedure for FlightGear 0.9 CVS. For FlightGear 0.9.8 you should extract the OpenAL files into the newly-created /fg-0.9.8 directory.

```
username@machine <any_path>
$ cd /fg-0.9.8

username@machine /fg-0.9.8
$ tar -xzf ~/tar/openal cyg.tgz
```

The directory tree now looks like this:



Build SimGear 0.3.8 Source

```
username@machine <any_path>
$ cd ~

username@machine ~
$ tar -xzf ~/tar/SimGear-0.3.8.tar.gz

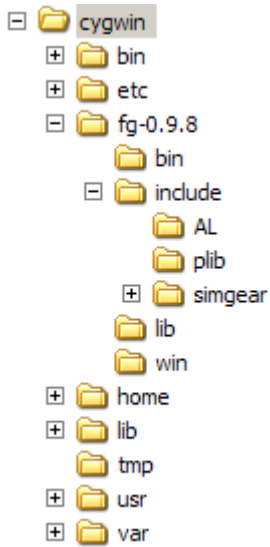
username@machine ~
$ cd SimGear-0.3.8

username@machine ~/SimGear-0.3.8
$ ./configure --prefix=/fg-0.9.8
```

```
username@machine ~/SimGear-0.3.8
$ make
```

```
username@machine ~/SimGear-0.3.8
$ make install
```

The directory tree now looks like this:



Build FlightGear 0.9.8 Source

```
username@machine <any_path>
$ cd ~

username@machine ~
$ tar -xzf ~/tar/FlightGear-0.9.8.tar.gz

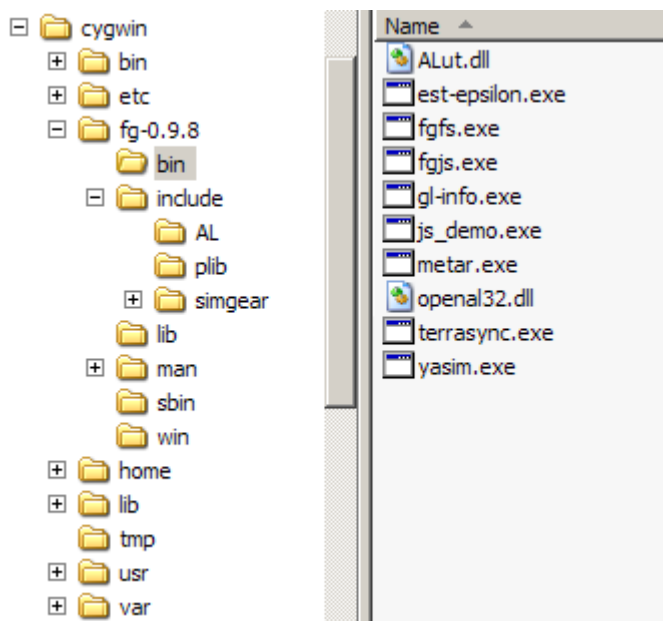
username@machine ~
$ cd FlightGear-0.9.8

username@machine ~/FlightGear-0.9.8
$ ./configure --prefix=/fg-0.9.8
```

```
username@machine ~/FlightGear-0.9.8
$ make
```

```
username@machine ~/FlightGear-0.9.8
$ make install
```

The directory tree now looks like the next picture (which illustrates the contents of /fg-0.9.8/bin):



Extract FlightGear 0.9.8 Base Package

The following commands will extract the FlightGear base package to the correct location. Note that previous tar commands have used a “-xzf” parameter but that this command uses a “-xjf” parameter (curious? Try “tar -help”).

```
username@machine <any_path>
$ cd /fg-0.9.8

username@machine /fg-0.9.8
$ tar -xjf ~/tar/fgfs-base-0.9.8.tar.bz2
```

Execute FlightGear 0.9.8

```
username@machine <any_path>
$ /fg-0.9.8/bin/fgfs.exe --fg-root=/fg-0.9.8/data
```

Hopefully, after a lengthy heart-stopping delay, you should see this beautiful picture of a Cessna sat at the end of the KSFO runway.



Build FlightGear 0.9 CVS (First Time)

The following section describes how to build the CVS version of FlightGear 0.9. Each command is listed but the output of each command has been omitted. In general you should ensure that each command reports no errors before typing the next command. If you choose to use a different directory structure then commands must be changed accordingly.

Build Plib 1.8.4 Source

This section assumes that ~/plib-1.8.4 has already been extracted in order to build FlightGear 0.9.8. Plib can be re-used with a slightly different "configure".

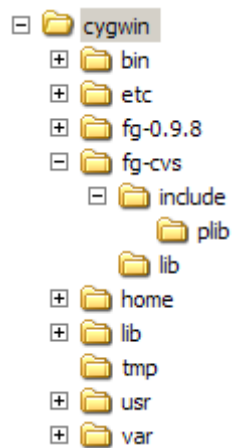
```
username@machine <any_path>
$ cd ~/plib-1.8.4

username@machine ~/plib-1.8.4
$ ./configure --prefix=/fg-cvs
```

```
username@machine ~/plib-1.8.4
$ make
```

```
username@machine ~/plib-1.8.4
$ make install
```

The directory tree now looks like this. Note how "/fg-cvs" has been created with some sub-directories containing the results of running "make install".



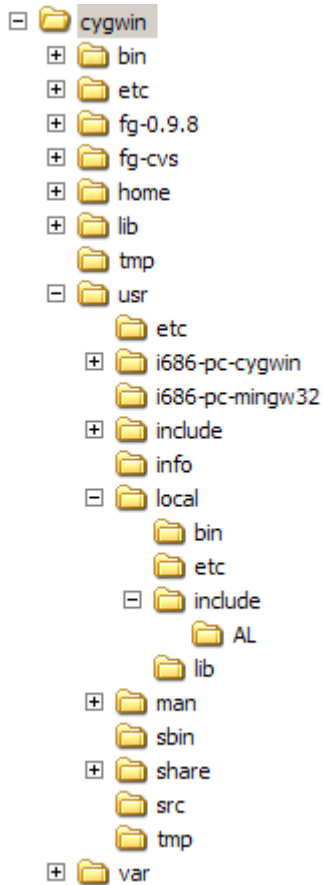
Extract OpenAL

The procedure for OpenAL with FlightGear 0.9 CVS is different to the procedure for FlightGear 0.9.8. For FlightGear 0.9 CVS you should extract the OpenAL files into the root directory. This will extract the files into /usr/local.

```
username@machine <any_path>
$ cd /

username@machine /
$ tar -xzf ~/tar/cyg_openAL.tgz
```

The directory tree now looks like this:



Get SimGear 0.3 CVS Source

Create a new directory for SimGear 0.3 CVS then use the CVS commands explained on the SimGear web site. The CVS password is "guest".

```
username@machine <any_path>
$ cd ~

username@machine ~
$ mkdir SimGear-0.3-cvs

username@machine ~
$ cd SimGear-0.3-cvs

username@machine ~/SimGear-0.3-cvs
$ cvs -d :pserver:cvsguest@cvs.simgear.org:/var/cvs/SimGear-0.3 login
The password is "guest"

username@machine ~/SimGear-0.3-cvs
$ cvs -d :pserver:cvsguest@cvs.simgear.org:/var/cvs/SimGear-0.3 co source
```

Build SimGear 0.3 CVS Source

There is an additional step that wasn't required when building the SimGear 0.3.8 release version. The SimGear 0.3 CVS version requires you to run `./autogen` from the `~/SimGear-0.3-cvs/source` directory before running `configure`.

```
username@machine <any_path>
$ cd ~/SimGear-0.3-cvs/source

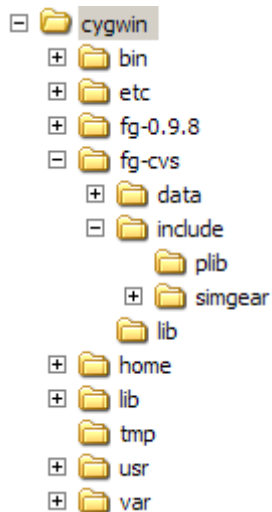
username@machine ~/SimGear-0.3-cvs/source
$ ./autogen.sh
```

```
username@machine ~/SimGear-0.3-cvs/source
$ ./configure --prefix=/fg-cvs
```

```
username@machine ~/SimGear-0.3-cvs/source
$ make
```

```
username@machine ~/SimGear-0.3-cvs/source
$ make install
```

The directory tree now looks like this:



Get FlightGear 0.9 CVS Source

Create a new directory for FlightGear 0.9 CVS then use the CVS commands explained on the FlightGear web site. The CVS password is "guest".

```
username@machine <any_path>
$ cd ~

username@machine ~
$ mkdir FlightGear-0.9-cvs

username@machine ~
$ cd FlightGear-0.9-cvs

username@machine ~/FlightGear-0.9-cvs
$ cvs -d :pserver:cvsguest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 login
The password is "guest"

username@machine ~/FlightGear-0.9-cvs
$ cvs -d :pserver:cvsguest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 co source
```

Build FlightGear 0.9 CVS Source

There is an additional step that wasn't required when building the FlightGear 0.9.8 release version. The FlightGear 0.9 CVS version requires you to run `./autogen` from the `~/FlightGear-0.9-cvs/source` directory before running `configure`.

```
username@machine <any_path>
$ cd ~/FlightGear-0.9-cvs/source

username@machine ~/FlightGear-0.9-cvs/source
$ ./autogen.sh
```

```
username@machine ~/FlightGear-0.9-cvs/source
$ ./configure --prefix=/fg-cvs
```

```
username@machine ~/FlightGear-0.9-cvs/source
$ make
```

```
username@machine ~/FlightGear-0.9-cvs/source
$ make install
```

Get FlightGear 0.9 CVS Base Package

Change your working directory to the `/fg-cvs` directory then use the CVS commands explained on the FlightGear web site. The CVS password is `guest`.

```
username@machine <any_path>
$ cd /fg-cvs

username@machine ~/fg-cvs
$ cvs -d :pserver:cvsguest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 login
The password is "guest"

username@machine ~/fg-cvs
$ cvs -d :pserver:cvsguest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 co data
```

Execute FlightGear 0.9 CVS

```
username@machine <any_path>
$ /fg-cvs/bin/fgfs.exe --fg-root-/fg-cvs/data
```

Hopefully you will see a Cessna sat at the end of the KSFO runway.

Build FlightGear 0.9 CVS (Subsequent Times)

Re-building FlightGear to the latest CVS version is very straightforward. Simply update the code from the CVS server, make and install. There is no need to re-run `autogen` or `configure` (unless, in very rare cases, the autogen files or configure files have themselves been updated).

Remember to update SimGear at the same time as you update FlightGear. These projects are working in parallel.

Update SimGear 0.3 CVS Source

Change your working directory to the SimGear CVS source directory then use the CVS command explained on the SimGear web site.

```
username@machine <any_path>
$ cd ~/SimGear-0.3-cvs/source

username@machine ~/SimGear-0.3-cvs/source
$ cvs update -d -P
```

Build Updated SimGear 0.3 CVS Source

Run “make” and “make install” from the ~/SimGear-0.3-cvs/source directory.

```
username@machine <any_path>
$ cd ~/SimGear-0.3-cvs/source

username@machine ~/SimGear-0.3-cvs/source
$ make
```

```
username@machine ~/SimGear-0.3-cvs/source
$ make install
```

Update FlightGear 0.9 CVS Source

Change your working directory to the FlightGear CVS source directory then use the CVS command explained on the FlightGear web site.

```
username@machine <any_path>
$ cd ~/FlightGear-0.9-cvs/source

username@machine ~/FlightGear-0.9.cvs/source
$ cvs update -d -P
```

Build Updated FlightGear 0.9 CVS Source

Run “make” and “make install” from the ~/FlightGear-0.9-cvs/source directory.

```
username@machine <any_path>
$ cd ~/FlightGear-0.9-cvs/source

username@machine ~/FlightGear-0.9-cvs/source
$ make
```

```
username@machine ~/FlightGear-0.9-cvs/source
$ make install
```

Update FlightGear 0.9 CVS Base Package

Change your working directory to the /fg-cvs/data directory then use the CVS commands explained on the FlightGear web site.

```
username@machine <any_path>
$ cd ~/fg-cvs/data

username@machine /fg-cvs/data
$ cvs update -d -P
```

Execute FlightGear 0.9 CVS

```
username@machine <any_path>
$ /fg-cvs/bin/fgfs.exe --fg-root-/fg-cvs/data
```


Summary

If you've been through this document a few times then this should be the only section you need. This is a summary of the steps required to build FlightGear CVS.

Install Cygwin

Install Cygwin with these packages:

- automake
- autoconf
- cvs
- make
- gcc
- gcc++
- OpenGL

Additional Downloads

```
cd ~
mkdir tar
  download cyg_openAL.tgz from http://www.vso.cape.com/~nhv to ~/tar
  download plib-1.8.4.tar.gz from http://plib.sourceforge.net
cd /
tar -xzf ~/tar/cyg_openAL.tgz
```

Build Plib

```
cd ~
tar -xzf ~/tar/plib-1.8.4.tar.gz
cd plib-1.8.4
./configure --prefix=/fg-cvs
make
make install
```

Get And Build SimGear

```
cd ~
mkdir SimGear-0.3-cvs
cd SimGear-0.3-cvs
cvs -d :pserver:cvsquest@cvs.simgear.org:/var/cvs/SimGear-0.3 login
  password is "guest"
cvs -d :pserver:cvsquest@cvs.simgear.org:/var/cvs/SimGear-0.3 co source
cd source
./autogen.sh
./configure --prefix=/fg-cvs
make
make install
```

Get And Build FlightGear

```
cd ~
mkdir FlightGear-0.9-cvs
cd FlightGear-0.9-cvs
cvs -d :pserver:cvsquest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 login
  password is "guest"
cvs -d :pserver:cvsquest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 co source
cd source
./autogen.sh
./configure --prefix=/fg-cvs
make
make install
```

Get Base Package

```
cd /fg-cvs
cvs -d :pserver:cvsquest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 login
    password is "guest"
cvs -d :pserver:cvsquest@cvs.flightgear.org:/var/cvs/FlightGear-0.9 co data
```

Execute

```
cd ~
/fg-cvs/bin/fgfs.exe --fg-root=/fg-cvs/data
```

Update And Build SimGear

```
cd ~/SimGear-0.3-cvs/source
cvs update -d -P
make
make install
```

Update And Build FlightGear

```
cd ~/FlightGear-0.9-cvs/source
cvs update -d -P
make
make install
```

Update Base Package

```
cd /fg-cvs/data
cvs update -d -P
```

Execute

```
cd ~
/fg-cvs/bin/fgfs.exe --fg-root=/fg-cvs/data
```

October 2005

END OF DOCUMENT