FTOOLS
for
Windows® 95*/98/NT

Welcome to the **FTOOLS for Windows ®** tutorial. Before beginning, please make sure that your system has the following components *:

1. Netscape
2. Windows 95, 98 or NT
3. FTOOLS for Windows software package (see the FTOOLS CD for installation and setup instructions)
4. WinZip (the demo version is sufficient and can be downloaded for free from http://www.winzip.com)

*This tutorial was written and tested with Netscape 4.7, Windows 98 and FTOOLS 4.2. Other versions of these may cause unanticipated problems. One known problem is that **some of the FTOOLS functions do not work with Windows 95**.

This tutorial contains background material, examples, and a variety of exercises to supplement the skills you'll be developing. Upon completion of the tutorial, you will be able to do the following:

1. Use the internet to download or study information related to X-ray astronomy and astrophysics
2. Download data files from a high energy astrophysics archive and store these files in a useful format
3. Use the Windows operating system to access and organize your files
4. Use the FTOOLS software package to visualize and analyze your data

Take your time throughout the tutorial. If you find one section particularly difficult, try doing it a couple of times until you become comfortable with the task at hand. Also, you should definitely read the README file that is included in your FTOOLS software. It contains an abundance of helpful information on the ins and outs of FTOOLS. On the FTOOLS webpage, from which you accessed this tutorial, there is a Troubleshooting Guide containing useful tips, known problem areas, and solutions to common questions.

Happy sky-gazing!

I. Getting Started

Your Desktop

To facilitate your work, you will find it quite beneficial to have the following icons on your screen, or ***desktop***:

FTOOLS (a white box with the word FTOOLS across it)
Microimages (the X-server; a red box with and 'MI' across it; it is probably called Xs.exe and can be found in the MIX_95 folder)
Windows Explorer (a folder with a magnifying glass on it)
Netscape (icon will vary depending on version of program)

An ***icon*** serves as a shortcut to a particular program--you can quickly open up a program by simply double-clicking on its icon. To create a program icon that isn't already on your desktop, open Windows Explorer, go to the *Tools* menu, select *Find*, and then select *Files or Folders*. In the *Named* entry field, type the name of the program you're looking for and its extension (probably .exe. Extensions are covered later in the tutorial.) In the *Look In* field, select *Local hard drives*. Finally, click *Find Now*. The program should soon appear below. Press and hold down the left mouse button to select the program, and then, still holding down the button, drag your selection to anywhere on the desktop. (This is called "click and drag".) Once you close the *Find* window and the Windows Explorer, you can place the icon wherever you wish simply by clicking on it and dragging it there.

Figure 1: Icons as they will appear on your desktop

Your File System

Windows uses a tree-like structure to organize your file system. To locate a file in the tree, you must follow a path to get there. Every directory and every file has its own location, or ***pathname***, consisting of a sequence of all of the branches followed to get to it. The first part of the path name is the ***drive name***. Though different computers are organized differently, **A:** usually indicates your floppy disk drive, and **C:** usually indicates your hard disk drive where your computer stores your files.

Your files are organized into ***directories*** and ***subdirectories***, which create the tree-like structure. Thus, the remaining part of a pathname is a sequence of directory and subdirectory names, ultimately ending with the name of the file (or folder) that you're looking for. This is perhaps more clearly represented by the following picture:

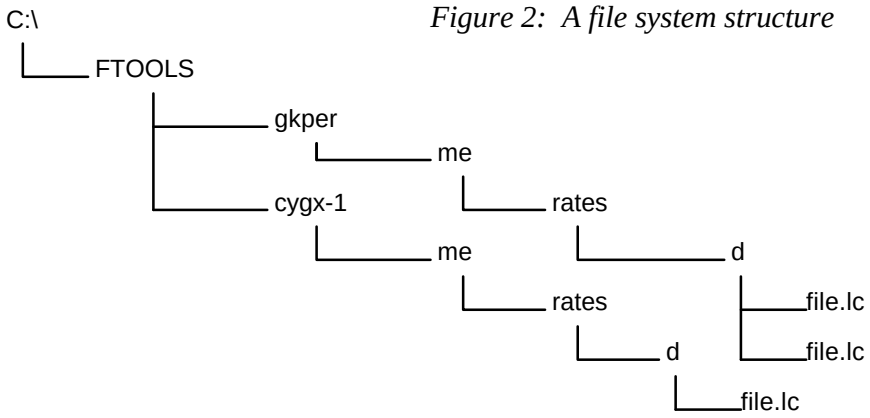


Figure 2: A file system structure

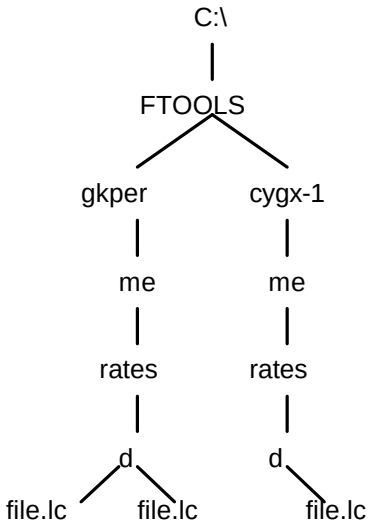


Figure 3: An alternate file system structure

Figures 2 and 3 show slightly different representations of a directory structure similar to the one you will be using to organize your data. Each source that you analyze, like GK Per and Cyg X-1 for example, will have its own directory that is a subdirectory of FTOOLS. (The directories *me, rates* and *d* will be created automatically when you untar and uncompress your downloaded files. More on this later.) Thus, the pathname for d05853.lc, for example, is C:\FTOOLS\gkper\me\rates\d\d05853.lc.

Exercise: *Draw **one** tree diagram (in either of the above styles) showing the following pathnames:*

C:\Planets\Mars\Red.txt
C:\Planets\Saturn\Rings.dat
C:\Planets\Jupiter\Moons\Europa.lc
C:\Planets\Jupiter\Moons\Ganymede.lc

Types of Files

As you may have noticed in the previous section, a filename has two parts: the actual name, and the ***extension***. The extension indicates the type of file. There are many file types; below is a list of the most common ones.

.fits -- image files commonly used for data taken at national telescope labs
.gif, .jpeg -- black and white or color image files
.ps -- postscript code for figures, pictures, etc.
.tar -- tape archive file
.txt -- text file
.Z -- file compressed with the compress command
.dat -- data file
.exe -- executable file; a program

(The files you will be downloading will be mostly .fits, .lc and .pha files, where .lc indicates lightcurve data and .pha stands for pulse height analysis, indicating spectral data.)

Final Steps

You're just about finished with Part I of this tutorial: there are just a few more things you must do before proceeding to Part II. First, find the X-Server icon you placed on your desktop earlier (remember Xs.exe?) and double click on it. ***Note:*** *upon opening the X-Server for the first time in each session, a window saying "This program has performed an illegal function and will be shut down" may appear. Just click OK: the program will not shut down.* In the upper right-hand corner of the X-Server window that opens, click the box with the '_' on it to minimize Xs. **You must make sure to open and minimize the X-Server, or FTOOLS will not function properly.** Finally, find the

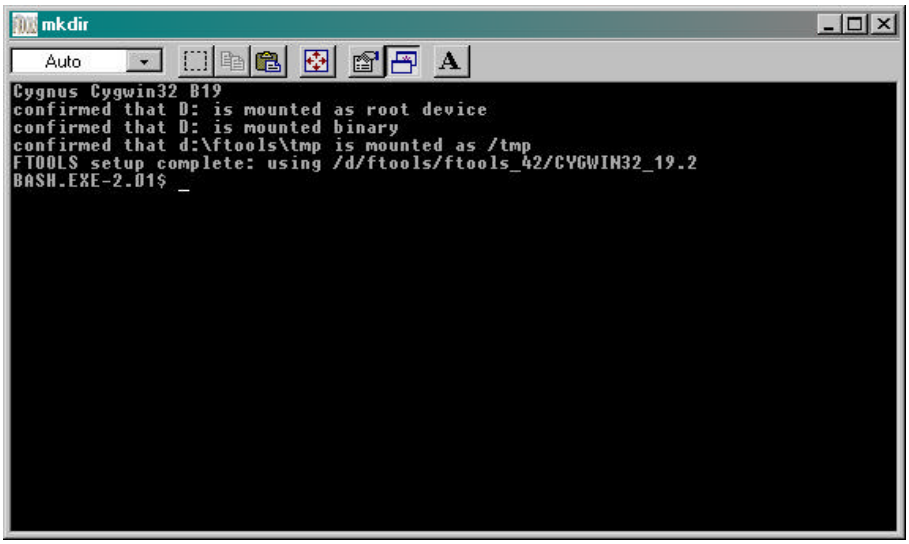


Figure 4: The BASH window

FTOOLS icon you placed on your desktop, and double click on it. A window like the one in Figure 4 will open. This is the Cygwin BASH window. It provides you with an environment in which you can utilize the various tools of analysis provided by the FTOOLS package, using a set of commands. These commands and their wide range of capabilities are the subject of Part II.

II. File System Commands

File system commands allow you to maneuver through and manage the files you download or create. The commands you are most likely to use for your work are: **cat, cd, compress, cp, file, grep, ls, mkdir, more, mv, pwd, rm, rmdir, tar, tail, gunzip, history** and **exit**. These commands are entered at the ***prompt***, which looks like this:

```
BASH.EXE-2.01$
```

and are followed by hitting 'Enter' or 'Return'. **IMPORTANT: when dealing with commands, the BASH window is case-sensitive. In other words, it will recognize 'cd' but not 'CD'.**

The commands can be organized by function into three groups**:** moving around, viewing and searching, and creating and deleting. (As a sidenote, don't worry if this seems like a lot to memorize: all it takes is some practice.)

Moving Around

- **pwd** displays the current working directory. For example:

```
BASH.EXE-2.01$pwd
/C/ftools
BASH.EXE-2.01$
```

(Note: Notice that the parts of the pathname are separated by /'s, not \'s like in Windows. This is just a characteristic of this type of environment, similar to a UNIX environment.)

- **cd <filename>** changes from your current directory to the directory you specify (if it exists). For example:

```
BASH.EXE-2.01$pwd
/C/ftools
BASH.EXE-2.01$cd fv
BASH.EXE-2.01$pwd
/C/ftools/fv
BASH.EXE-2.01$
```

- **cd ~** moves you from your current directory to your home directory (c/ftools).

- **cd ..** moves you from your current directory to one level higher in the directory tree. For example:

```
BASH.EXE-2.01$pwd
/C/ftools
BASH.EXE-2.01$cd gkper/me
BASH.EXE-2.01$pwd
/C/ftools/gkper/me
BASH.EXE-2.01$cd ..
BASH.EXE-2.01$pwd
/C/ftools/gkper
BASH.EXE-2.01$
```

Viewing and Searching

- **ls** is the flexible command allowing you to view the contents of your current directory. This is the first example of a command which takes ***switches***, that is, extra letters preceded by a '-', enabling the command to do different things. The common switches used with **ls** are:

  -a : list all entries inluding files starting with '.'
  -l : list all entries, one per line
  -r : list all files in reverse alphabetical order
  -u : list files, show when file was last modified rather than last created

  The various switches can be combined in any order behind one '-'. For example, **ls -al** lists all entries, including files starting with '.', one per line

Exercise: *Try each of the above ways to list the contents of the ftools directory. Now try combining a couple of them. What do you prefer? What situations do you think each command is best suited for? Try **ls -l** in ftools, then try **ls -al**. What is the difference in output?*

- **cat <filename>** puts the contents of the file on the screen. This only works if the file is less than one screen page long.

- **more <filename>** dumps the contents of a file on the screen, one page at a time. Hit the Space Bar to go from page to page, Return or Enter to step forward line by line, and 'q' to go back to the prompt.

- **tail <filename>** prints the final ten lines of a file on the screen. **tail -n** prints the final n lines of a file on the screen (you replace n with the number of lines you wish see)

- **grep <string><filename>** searches for a particular word or phrase (what you put in place of <string>) in the file you specify

Exercise: *Use **grep** to find the word 'nasa' in /C/ftools/readme.*

Exercise: *Try out the **cat**, **more**, **tail**, and **tail -n** commands on /C/ftools/readme.*

- **compress** allows you compress files in order to save space during transport of the files over the internet. There are several ways to compress files, but the one you will use most often is the **compress** command. The switch '**-v**' displays how efficient the compression was. For example:

```
BASH.EXE-2.01$compress -v lightcurve.ps
lightcurve.ps:  compression 71.74%
```

- **gunzip <filename>** decompresses a file (note that the file in place of <filename> must have the extension **.Z**, indicating that it is a compressed file.)

- **tar** is usually used to construct tape archives for uploading to or downloading from some magnetic medium like tape. It is convenient because one can upload an entire directory as one archive entry, which eases bookkeeping greatly. **tar** has several switches. The most common are:
  - -x : extract archive
  - -v : verbose listing of archive contents
  - -f : use 'tar' argument as name of archive
  - -c : create archive

  For example:

```
BASH.EXE-2.01$tar -xvf w3browse-16280.tar
me/rates/d/d45390.lc.Z
me/rates/d/d45397.lc.Z
me/rates/d/d45404.lc.Z
me/rates/d/d45413.lc.Z
BASH.EXE-2.01$
```

  creates the subdirectory tree 'me/rates/d', and places all the files (which, in this case, end with 'lc.Z') in subdirectory 'd'.

Creating and Deleting

- One way to create a file is with the **cat** command, but in the following strange way: **cat > <filename>**. **cat** is really a multipurpose utility which you will see again and again, and actually appends one file to another. **cat >** means append input from the keyboard to the file you specify in place of <filename>. (This is an example of

redirection, which is explained later in the tutorial). If the filename you specify already exists, it will be overwritten by the new file you type. If it does not yet exist, a file with the name you specify will be created. To enter information into the file, type what you wish, hitting return after each line; when you are finished, go to the next line and hit the 'Control' and 'D' keys simultaneously to exit and save.

- **emacs** is an easy-to-use word processor that provides another way to create (or edit) a file. To use emacs, you need to change directories. Go to **c/emacs-19.34/bin**. (Note that the program is not located inside the FTOOLS directory.) To create a file, type **emacs <filename>**, where <filename> is replaced with what you wish to name the file. To edit an already existing file, type **emacs <filename>**, where <filename> is the name of the file you would like to edit. In either case, when you are finished, hit the 'Control' and 'X' keys simultaneously, and then hit the 'Control' and 'S' keys simultaneously to save. Then hit the 'Control' and 'X' keys at the same time again, followed by 'Control' and 'C' together to exit emacs. The file will be saved in **c/emacs-19.34/bin**.

- **cp** is another way to create a file; in this case, you are copying one file to another. For example:

```
BASH.EXE-2.01$cp test.txt test2.txt
```

creates the file 'test2.txt' which is identical to 'test.txt'. This is a useful way to create backup copies of files in which you might make significant changes. That is, if you screw up, you can always go back to your backup copy.

- **mv** is yet another way to create a file; however, you lose the original file. **mv** has two functions: moving files from one directory to another, and renaming files in the same directory. You can also, in principle, move a file to another directory while changing its name. For example: (*Note: don't try to perform this example*)

```
BASH.EXE-2.01$mv test.txt trial
```

moves 'test.txt' from the current directory to the subdirectory 'trial.'

```
BASH.EXE-2.01$mv test.txt test2.txt
```

copies 'test.txt' to 'test2.txt' and removes the file 'test.txt'. In essence, you've renamed the file from 'test.txt' to 'test2.txt'.

```
BASH.EXE-2.01$mv test.txt trial/test2.txt
```

moves 'test.txt' to the directory 'trial', and changes the filename to 'test2.txt'.

- **rm <filename>** removes the file you specify. For example:

BASH.EXE-2.01$**rm test.txt**

removes (deletes) the file 'test.txt'.

- **rmdir <directoryname>** removes a directory. However, the directory must be empty of all files before you can delete it.

- **mkdir <directoryname>** creates a directory with the name you specify.

Exercise: *Using the commands you've learned, create a file called 'myfile.txt' that contains the words "This is my test file." Then, create a directory called 'mydirectory'. Put 'myfile.txt' in 'mydirectory', and rename it 'myfile2.txt'. Now, delete everything you've done in this exercise. (Hint: use **ls** and **cd** along the way to help you to keep track of what your file system looks like.)*

- The final two important commands are **history** and **exit**. By typing **history** at the prompt (followed by the 'Enter' or 'Return' key, of course), you will receive a chronological listing of all of the commands you entered during your session. Typing **exit** allows you to terminate your session. Then, depending on how your BASH window is set up, the window will either close by itself, or you will have to close it by clicking on the X in the window's upper right-hand corner.

III. Time Saving Tips

- Command line completion: if you are in the middle of typing a long filename, bit the 'Tab' key to automatically complete that name. (**Note**: this only works if *exactly one* file has that name or starts with the letters you've typed.)

- Re-executing commands: at the prompt, you can press the up arrow key to get the last command you entered. Keep hitting it to get more prior commands.

- In Windows Explorer, you can select multiple, consecutive files by clicking on the first one, pressing shift and then, while holding shift down, clicking on the last file you wish to select. All of the files in between and including the ones you chose will be selected (highlighted).

IV. Redirection and Combining Commands

Redirection and combining of commands are both relatively simple concepts, and both greatly enhance the versatility of your commands. These very compact expressions just make your work a little easier:

<, >, >> -- redirection
| -- piping

; -- a simple way to combine commands
* -- a wildcard

The simplest way to combine commands is to use the wildcard symbol, '*'. This allows you to apply the command to files of similar types. For example:

```
BASH.EXE-2.01$grep start *.dat
```

searches for the string 'start' in all files with the extension '.dat' (which usually indicates your data files).

If you would like to send output or take input from nonstandard directions, use one of the redirectors: <, >, or >>. These get input from a file, send output to a file, or append to a file, respectively. An example is the **cat >** command you've already learned, which redirects input from the keyboard into a file. Likewise, **cat >>** redirects input from the keyboard to the end of an existing file.

Another way of combining commands uses the semicolon, ';'. It allows you to combine several commands on one line. For example:

```
BASH.EXE-2.01$cd text;pwd;ls -al
```

changes from your current directory to the 'text' directory, checks the directory name, and lists all of the files in the 'text'directory.

The final tool for combination of commands is '|'. The statement **<command1>|<command2>** tells the computer to send the output from your first command into the input of your second command. For example:

```
BASH.EXE-2.01$grep turn test.txt
And returned on the previous night.
BASH.EXE-2.01$
```

gives the line from the text file 'test.txt' containing the string (word) 'turn'. This output can be reproduced in the following way:

```
BASH.EXE-2.01$cat test.txt|grep turn
And returned on the previous night.
BASH.EXE-2.01$
```

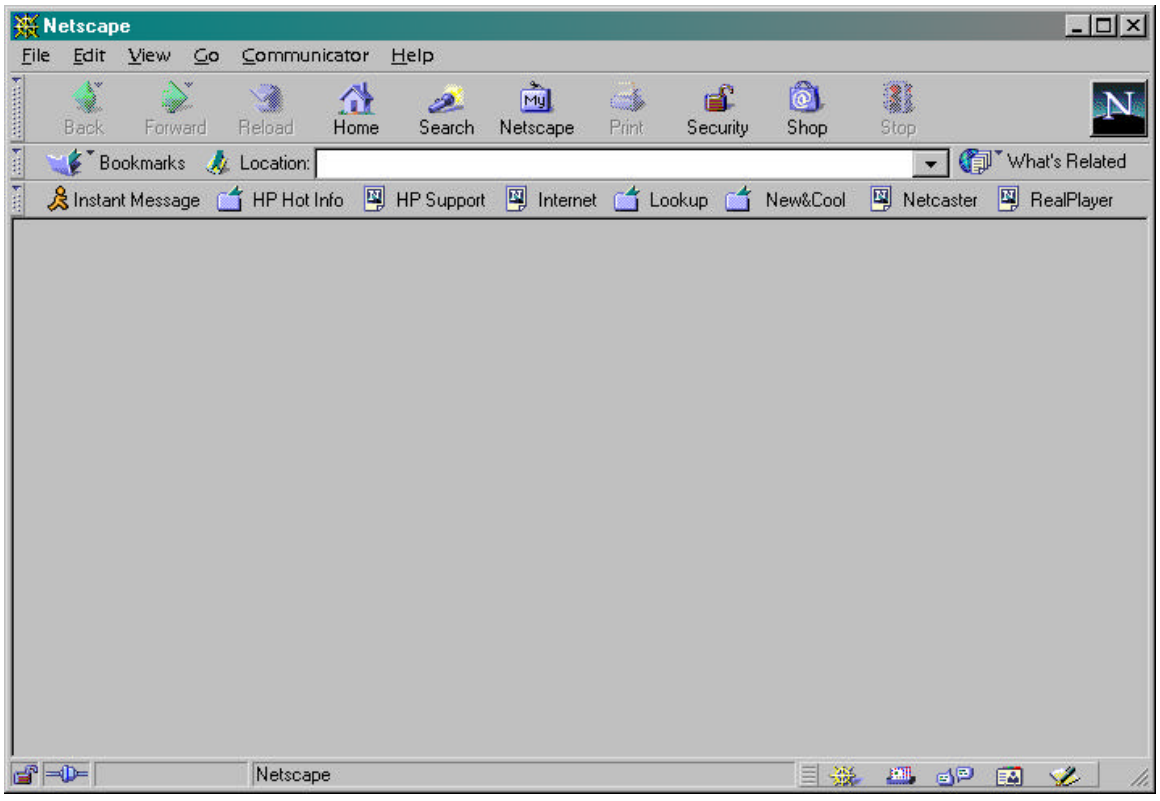
In the second case, we've used two commands instead of one. The input to **grep** was the entire file 'test.txt'. **grep** then proceeded to search for the string *turn* in that input. In this example, note how '|' was used like a 'pipeline' between two commands. This pipeline is what makes '|' different from ';', in that the pipe allows the commands to share information. ';' only separates the commands. There is no information transferred through the ';'.

V. Exploring Astronomy and Astrophysics on the Internet

Now that you have a command of the core background information you'll be using, it's time to proceed with the real reason you're doing this tutorial: to learn how to use the internet in combination with FTOOLS to explore various areas of Astronomy and Astrophysics.

First, just to give you an idea of just how much information is out there, check out the list of sites on the next page. Get started by opening the Netscape browser. (Double-click on the Netscape icon you placed on your desktop earlier.) Once the browser opens, you can reach a site by entering the site's address in the 'Location' entry field (see Figure 5).

Figure 5: The Netscape browser (Version 4.7)



- **Astronomy Picture of the Day** – *antwrp.gsfc.nasa.gov/apod:* Each day, a different image or photograph of our fascinating universe is displayed, along with a brief explanation written by a professional astronomer.

Exercise: *Locate the archive on this site. How many years does the archive contain? Select an interesting picture and describe it in your own words.*

- **HEASARC Homepage** – *heasarc.gsfc.nasa.gov:* The **H**igh **E**nergy **A**strophysics **S**cience **A**rchive **R**esearch **C**enter. This will be the archive you'll be using to study X-ray sources, **so take some time now to familiarize yourself with its contents.**

- **Hubble Space Telescope Public Pictures** – *oposite.stsci.edu/pubinfo/pictures.html:* Contains the latest images taken by the Hubble, in addition to an image database archive.

- **Imagine the Universe!** – *imagine.gsfc.nasa.gov/docs/homepage.html:* This site contains the important questions (with some answers) about the cosmos, and discusses how astronomers and astrophysicists come to know what they know.

- **NASA Homepage** – *www.nasa.gov/NASA_homepage.html:* "NASA is deeply committed to spreading the unique knowledge that flows from its aeronautics and space research…" An astronomical amount of information about NASA and its past, present and future missions can be found here.

Exercise: *Describe three important things you can find on the above site.*

- **Ned Wright's Cosmology Tutorial** – *www.astro.ucla.edu/~wright/cosmology.htm:* Probes the questions about the history, present state and future of the universe.

- **SkyView** – *skyview.gsfc.nasa.gov/skyview.html:* Provides Virtual images of the sky in different wavelength regimes from radio to X-ray.

- **Space Telescope Science Institute** – *marvel.stsci.edu:* Contains more information about the Hubble telescope and its images.

VI. HEASARC

As mentioned earlier, HEASARC will be your most important source for X-ray astronomy data. It provides information about past and present X-ray telescope missions,

satellite instruments, data imaging, and analysis software among other things. It also has an extensive archive of data sets from these missions.

The data you will be analyzing first, comes from the X-ray source Cyg X-1. Thus, our first step is to create a file folder by that name. Minimize Netscape and go to your BASH window. (Remember, make sure that your X-Server is open and minimized, or your BASH window will not work properly.) At the prompt, type **pwd** to make sure that you are in /c/ftools/. If you are not, go there. Now, create a directory called 'cygx-1' (typed *exactly* as seen in the '') using the **mkdir** command you learned earlier. Now, when you type **ls**, you should see 'cygx-1' in the list of file and directories that appears. Open it with the **cd** command. Now you are ready to go. (You will execute the previous process for every source you analyze: every source should, for organizational purposes, have its own folder, and you must be *in* that folder before you gather your data from HEASARC.)

Now, maximize Netscape (by clicking on its minimized form in the taskbar on the bottom of your screen), go to the HEASARC website, and do the following:

- Click on *Data Archive* at the top of the page.
- Click on the *W3Browse* archive in the yellow bar on the left side of the screen.
- Click on *Basic*.
- Click on *EXOSAT* in the ***Past Mission Archives*** and on *X-ray* in the ***Master Catalogs***. Then type **cyg x-1** in the ***Object Name or Coordinates*** field back at the top of the page.
- Click *Submit Query*.
- Scroll down to the Medium Energy archive, ***EXOSAT ME Spectra and Lightcurves (HEASARC ME)***. You'll notice that the data has several parameters distinguishing amongst them. The most important are *exposure* (in seconds), *qflag_me*, and *sequence number*. The *exposure* gives the length of the measurement in seconds. Higher exposure means that longer periods can be inferred from the data. *qflag_me* is a quality indicator for the data set. **0** means unusable, **1** is very poor, **2** is poor, **3** is acceptable, **4** is very good, and **5** is excellent. Finally, we will use the *sequence number* to keep track of each data set. Choose the data set with sequence number 931, exposure of 3969 seconds, and quality number 5 (by clicking on the box to the left of the data set).
- Scroll down and click the *submit* button to the right of the ***Categories of data products available for HEASARC_ME*** box.
- Click on *Multiband Lightcurve* (GIF), in blue. You should see a plot of its

lightcurve (shown in two different energy ranges). You'll have to scroll down to see the entire thing. (Figure 6)

Figure 6: Multiband Lightcurve plot for Cyg X-1

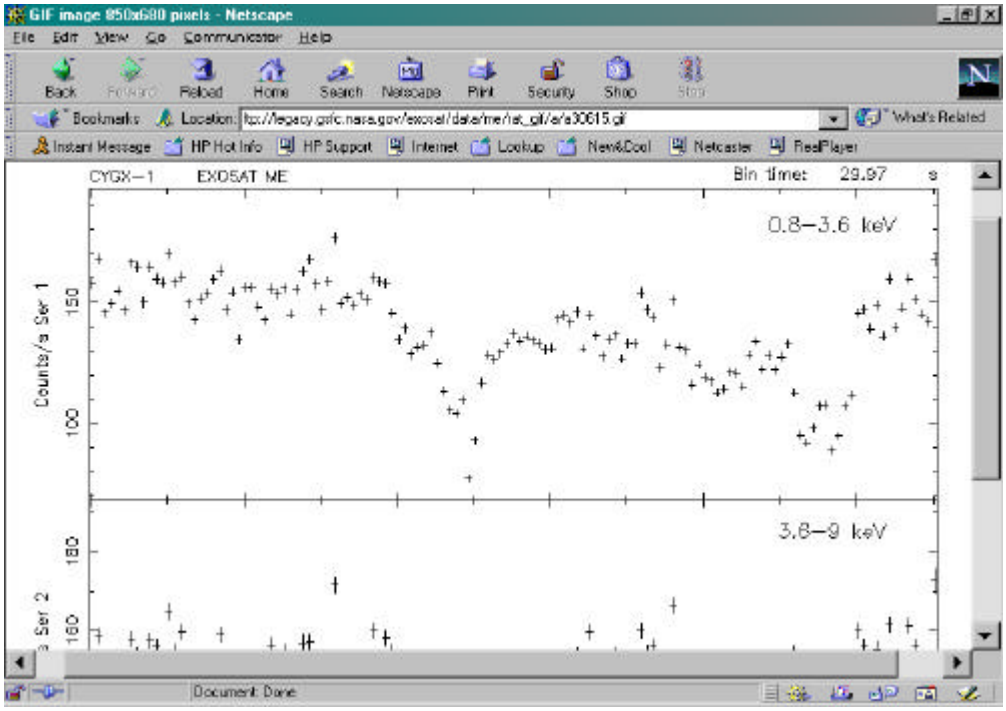
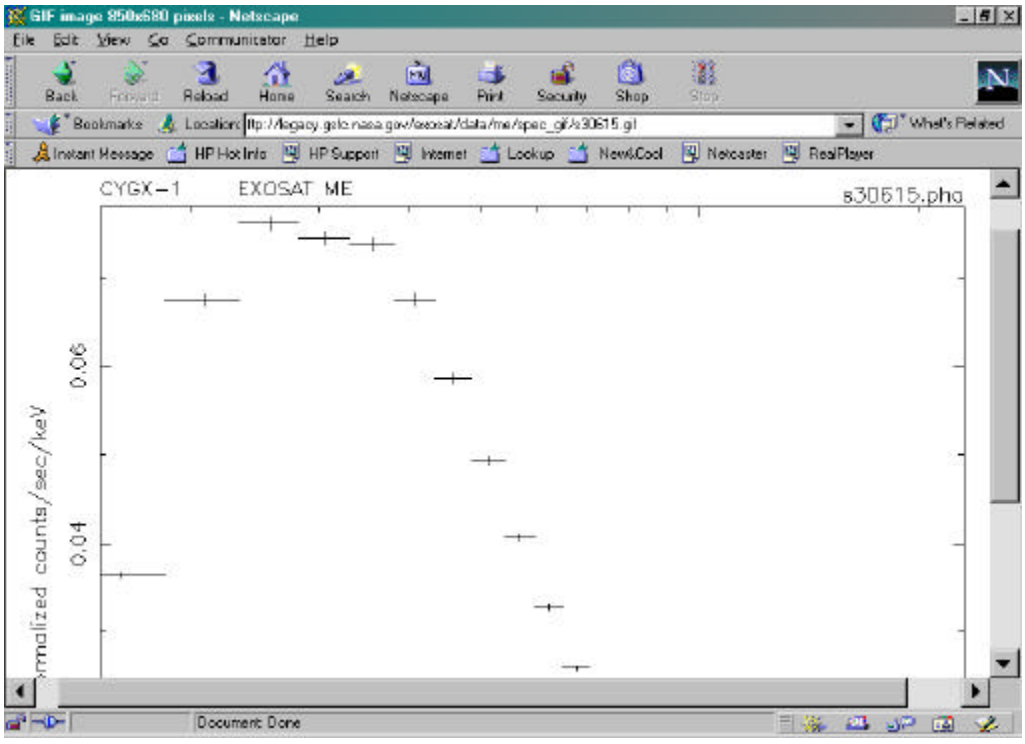


Figure 7: Argon 1-15keV Spectrum plot for Cyg X-1



- Click ‘Back’ on your browser. Then, click on *Argon 1-15keV Spectrum* (GIF). **Make sure you select the GIF file, NOT the FITS file**. You should now see its emission spectrum (Figure 7). Again, you’re going to have to scroll to see the full plot.

- To study the actual data, you need to download the FITS files:

    - Click ‘Back’. You should be brought back to the page where you selected *Multiband Lightcurve* and *Argon 1-15keV Spectrum (GIF)*.

    - Click on the boxes to the left of *Argon 1-15keV Spectrum (FITS, s30615.pha.Z)* and *0.8-8.9keV Lightcurve (FITS, d30615.lc.Z)*.

    - Click on *Retrieve Data Products*.

    - You should see the paths of the two files you want to download. (Are these files compressed? How do you know?) These files are automatically tarred together if you click on *Download TAR file*. Go ahead and do that. A Warning window will pop up. Its purpose is just to warn you about security considerations when downloading. Make sure that ‘Save it to disk’ and ‘Always ask before opening this type of file’ are selected. Then click ‘OK’. Now, a window entitled ‘Save As’ will appear. In the ‘Save in’ field, make sure it says [C:] (or whatever the letter is of the drive in which your FTOOLS package is stored). Next, find the FTOOLS folder in the list of file folders below ‘Save in’ and double click on it.

    - Displayed now are the contents of the *ftools* folder. You should see the folder you created earlier, *cygx-1*. Double click on it. Now you are finally in the directory in which you want the data files to be placed. Click ‘Save’.

    - Once the files have been downloaded, you need to untar and uncompress them. Minimize Netscape and go back to your BASH window. At the prompt, type **pwd**: you should still be in **c/ftools/cygx-1**. If you aren’t, make sure you go there because FTOOLS will not work properly if you are in a different directory. Now use **ls** to look at the contents of your ‘cygx-1’ folder. It should contain the file ‘w3browse-<*nnn…*>.tar’, where the *n*’s are various numbers.

    - Type **tar –xvf w3browse-<nnn…>.tar** (replacing **<nnn…>** with the correct numbers, of course).

- Two pathnames will be returned to you, for the two data files you must decompress: **me/spectra/s30615.pha.Z**)and **me/rates/d/d30615.lc.Z**. (Note that **s30615.pha.Z** is spectral data, and **d30615.lc.Z** is lightcurve data). Now type **ls** and notice that you now have another directory in your cygx-1 folder: **me/**. According to the pathname for **d30615.lc.Z**, that file is located a couple of levels further inside the cygx-1 folder. In order to uncompress it, you must get to its directory. To do this, type **cd me/rates/d**. Now, when you type **ls**, you should see **d30615.lc.Z**.

- To uncompress the file, use the **gunzip** command you learned earlier. Type **gunzip d30615.lc.Z**.

- Typing **ls** now will give you **d30615.lc**. This file is ready for analysis. Now you need to uncompress the other file you downloaded, the spectral file. In order to do this, you need to go back up a few levels, to **c/ftools/cygx-1**. Use the **cd ..** and **pwd** commands as many times as necessary to get you back to **c/ftools/cygx-1**. According to the other pathname you received earlier, the spectral file is located at **me/spectra/s30615.pha.Z**. Thus, begin by typing **cd me/spectra**, of course.

- Typing **ls** reveals the compressed file. Again, use the **gunzip** command: type **gunzip s30615.pha.Z**. Typing **ls** again shows you the uncompressed file, **s30615.pha**, ready for analysis.

- Finally, go back to the **cygx-1** folder, and delete the **.tar** file. Every time you finish the untarring and uncompressing process, you should go back and delete the **.tar** file. (If you don't, you will soon run out of disk space!)

Exercise: *Repeat the entire previous process with the EXOSAT ME data sets for the source AM Her. The sequence number is **85**. Download both the broadband lightcurve (FITS, d05479.lc.Z) and the Argon spectrum (FITS, s05479.pha.Z). Don't forget to place **everything** in an AM Her folder, which itself goes in your FTOOLS folder. (Directory names must only be one word long, so either 'amher' or 'am_her' would be an appropriate name for this folder.)*

Exercise: *Download one other data set for AM Her, sequence number **167**. Download **both** 0.8-8.9keV lightcurves (d08040.lc and d08054.lc). As before, untar and uncompress these files to prepare them for analysis.*

Exercise: *Download some ROSAT AM Her data. Follow the initial steps as for EXOSAT, but click on ROSAT under **Active Mission Archives** instead of EXOSAT under **Past Mission Archives**. Download the second PSPCB file (data set: **RP300067a00**, exposure: **11647** sec). Click on the **Basic Science Events** box on the data retrieval page. Untar and uncompress the file as before. **Note**: If, during the untarring and*

decompression process, you receive a message in the BASH window about and "EOF exception", go back and download the data again to get a different .tar file. This error may indicate that there is something wrong with the data.

VII. Essentials of FTOOLS

FTOOLS is the analysis package you'll be using to study the lightcurves. This package should already been installed and available to you, though you can't tell immediately. In other words, there is no separate FTOOLS prompt: you'll still be working with your BASH prompt, **BASH.EXE-2.01$**. (If, for some reason, FTOOLS is not installed on your computer, please refer to the FTOOLS CD for installation and setup instructions.) The package contains many routines called **tasks** designed to view, study and analyze astronomical image files, commonly called FITS (Flexible Image Transport System) files.

To get help on any of these tasks, type **fhelp <task>** at the prompt, replacing <task> with the name of the task you need assistance with. FTOOLS will return a synopsis of the task, and definitions of its inputs or parameters. Use the down arrow key to scroll through the information, and when you're finished, press 'q' to exit **fhelp** and go back to the prompt. During a session, you may want to know more about the meanings of the inputs a task uses. To do this, type **plist <task>**. **plist** is short for **p**arameter **list** and does exactly what its name implies. On each line, you see a parameter name, an = sign, and the value of the parameter, followed by a very short definition of the parameter.

The FTOOLS tasks we'll use most often are **flcol**, **fstatistic**, **fplot**, **lcurve**, **powspec**, and **concatenation**. These tasks are helpful when dealing with EXOSAT data; a separate series of tasks are used for ROSAT data and will be covered later. (*Note: while an **xspec** command exists for the UNIX version of FTOOLS, it does not yet exist for the Windows version.)*

- **flcol**: This task searches a FITS file for the names of columns containing data, and lists these column. This is a crucial command, because you'll need these column names for any statistical or plotting tasks in FTOOLS. For example:

```
BASH.EXE-2.01$flcol s30615.pha


___Column_Names____Formats_____Dims______Units___
CHANNEL                I
RATE                   E                   count /s
STAT_ERR               E                   count /s
QUALITY                I
GROUPING               I

___Column_Names____Formats_____Dims______Units___
DETNAM                 16A
```

```
OBSFACT                 E
GEOAREA                 E                       cm**2
ONTIME                  E                       s
DETGAIN                 E
GNCOEFF                 4E
RSCOEFF                 8E

BASH.EXE-2.01$
```

As you see, it returns a variety of information, most importantly the units of your data. You see that the data in this file is in several columns. The most important of these are the number of collected counts per second: *rate* and the energy *channel* in which these counts were collected.

- **fstatistic**: returns simple statistics about any column of your data file, such as sum, mean, standard deviation, min and max. If you type **fstatistic** at the prompt, you will be asked for the file and column names. Leave the range of rows unchanged for now. If the input file was *s30615.pha* (the energy spectrum of GK Per) and you chose *rate* as the column, then the output would be:

```
The sum of the selected column is 595.44027
The mean of the selected column is 18.607508
The standard deviation of the selected column is 21.412663
The minimum of selected column is 0.46192050
The maximum of selected column is 59.494179
The number of points used in calculation is 32
```

Exercise: *Try the above exercise using d30615.lc as the input file. (Hint: make sure you are in the right directory before you start the exercise.)*

- **fplot**: is a very useful environment, allowing you to make simple plots of the FITS files you have. After typing **fplot**, you are prompted for the filename and then for names of two columns you would like to plot (use **flcol** to get these column names **before** using **fplot**). In this case, enter 's30615.pha' for the filename. The columns you choose are the X- and Y- axis parameters: here, enter 'channel' and 'rate', respectively. Next, it asks for 'Lists of rows', for which you enter '-' to indicate the default value. You are then prompted for output type: enter **/xw**, which stands for XWindow and indicates output to the screen. Next is '…plot command': enter 'marker off'. *Upon hitting enter, if this is the first time you are using the X-Server in a session, you may get a message saying "This program has performed an illegal function and will be shut down." Just click 'OK'. The program will not shut down.* Finally, go down to your taskbar and click on the minimized Microimages X-Server. When it opens, you'll see your plot. You'll reproduce Figure 8 if you use **fplot** on the *rate* and *channel* columns of *s30615.pha* while keeping the marker off. Adding markers gives you Figure

9 (on the next page).

*Figure 8: Plot produced by **fplot** using the rate and channel columns of s30615.pha*

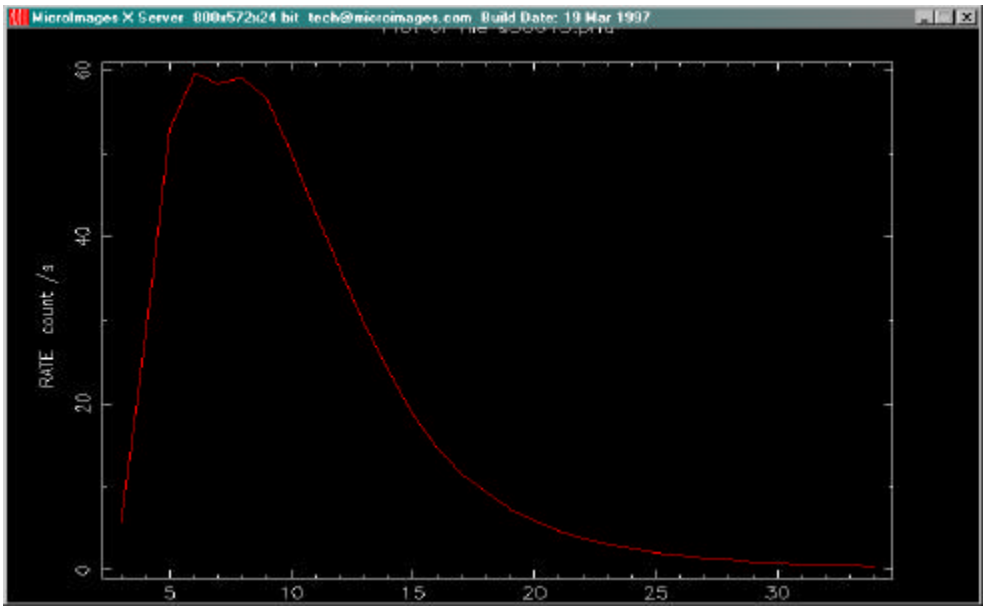
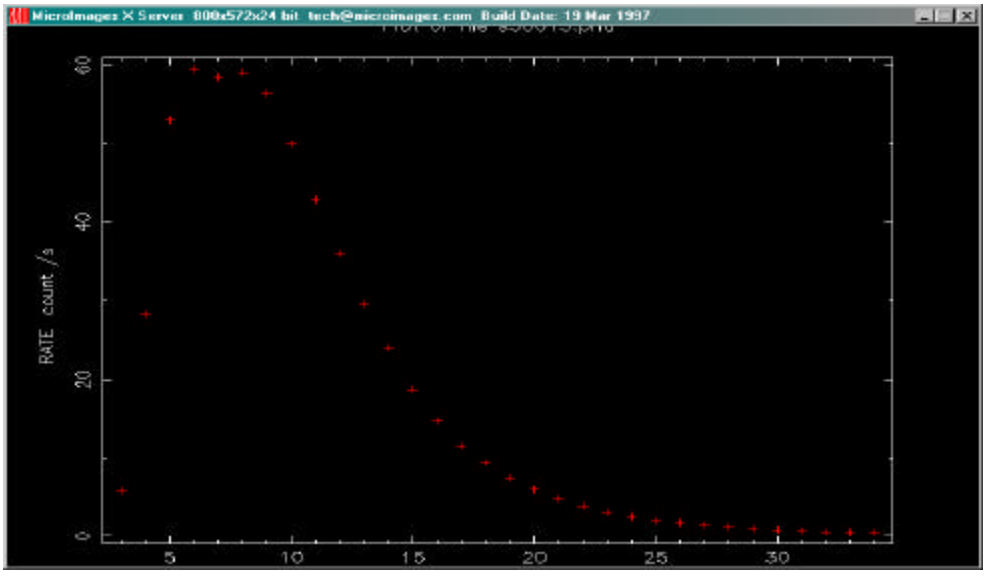


Figure 9: Plot produced in same way as in Figure 8, but with marker on



Notice also the **PLT>** prompt. This is the plotting environment which leaves you in charge of the data you want to see. The following commands will help quite a bit in this environment:

- **help**: once you get the **help>** prompt, type the command you want to know about. Hitting the 'Control' and 'D' keys simultaneously gets you back to **PLT>**. It will also exit you from the PLT> environment, and return you to the BASH prompt.

- **plot**: gives you either a line or scatter plot.

- **marker off/on**: switches from line to scatter plot.

- **marker size N on**: changes the marker size to N where N is a real number from 1.0 to 5.0

- **marker Z on**: switches to a specific marker type. Z goes from 1 to at least 16.

- **rescale x xmin xmax**: rescales the plot to zoom in on a part of your data set. FPLOT plots the entire data set by default, so rescale would be used to alter this. The plot is automatically updated after rescaling.

- **hardcopy <filename>.ps/ps**: makes a Postscript copy of the current plot for you to print out later using Ghostview (software included in the FTOOLS package). The file will be called whatever name you insert in place of <filename>. (Example: gkresult.ps/ps) Instructions for using Ghostview can be found at the end of this tutorial.

- **device [device]**: sends the output to the specified device. The device can usually be either /xw or /ps. This is useful if you want to switch from screen to hardcopy in the middle of an analysis.

- **exit**: exits the pgplot environment. This is useful if you get tangled up while trying to adjust your plot. Also, if this is the first command you type upon entering the pgplot environment after image/light curve/spectrum extraction, it will clear events and start the extraction process all over again.

Moving back to that Cyg X-1 plot, we now have the tools to examine the data in more detail and make more sophisticated plots. At the **PLT>** prompt, type **marker on** and hit 'Enter' (or 'Return'). At the next **PLT>** prompt, type **marker size 5.0 on**. This plots each datum with a cross of size 5.0 point. Typing **plot** gives you that same plot, but with discrete data this time, so you can get a much better idea of the errors in your data. By default, **fplot** places labels on your plot. These labels are nothing more than the

output from **flcol**.

Exercise: *Use **flcol** and **fplot** to make a scatter plot of the file s05479.pha. This is an EXOSAT ME spectrum of the source AM Her (sequence number 85), so it is a good idea to plot CHANNEL vs. RATE. Use triangular markers (you will have to experiment to find them).*

- **lcurve**: is a task which takes an input file and plots an appropriately binned lightcurve. The simplest way to use lcurve is to type **lcurve** at the BASH prompt. After loading, FTOOLS will step you through the inputs to **lcurve**. The inputs are as follows: number of time series, name of input file, name of outside window file (for which we will always enter '-'), newbin time (i.e. time bin width), number of newbins, output file name, and a couple of plotting preferences.

  Try this example: after making sure that you're in the directory *cygx-1/me/rates/d*, enter **lcurve** at the prompt. Type **1** for 'number of time series' and then **d30615.lc** for the 'input file'. After **lcurve** loads the file, you will see some information about the file, including the start/stop time of the experiment and the bin width (usually in seconds). Next, enter '**-**' to indicate the default window file. More information will appear. One piece of data is the *Minimum Newbin Time*, which, in this case, is **29.97** seconds. Enter this number at the prompt for *Newbin Time*. Now, look at the *Maximum Newbin Number* that is given. Here, it is 133. Thus, if you choose **133** as your 'Number of Newbins/Interval', 133 "newbins" of 29.97 seconds each will be put on the graph, and the graph will thus show the entire observation on one plot. By varying the *Newbin Time* and *Number of Newbins/Int.*, you can figure out how to best view the data.

  Next, you'll be prompted for the name of the output file. To avoid confusion, choose the same prefix as the input file but append a **.flc** to distinguish it from the original (i.e. **d30615.flc**). Finally, you will be asked if you want to plot your results, followed by a query about a plotting device. Answer **yes** and **/xw**, respectively. Click on the minimized X-Server to view your results.

Exercise: *Look at the lightcurve and the power spectrum of some data taken of the source AM Her. The file is called d05479.lc (sequence number 85, broadband EXOSAT ME). This is a nice, dramatic source once you get around the fact that mot of the data taken over nearly a day has zero counts. Enter the following parameters when prompted:*
Number of time series for this task: 1
Ser. 1 filename: d05479.lc
Name of window file: -
Newbin Time: 30
Number of newbins per interval: 20000
Name of output file: d05479.flc

Plot your results?: yes
PGPLOT device: /xw
With the number of newbins/interval = 20000, you'll get a squeezed plot. Hit 'Control' and 'C' together to get out of lcurve, and try again with newbins/int. = 6000, then 3000. At 3000, you should see the data most clearly. The interesting part of the curve is between 0 and $1x10^4$ seconds. To see error bars on the lightcurve, the x axis should be smaller than the entire data set. Use, for instance, newbins/int. = 2000. Estimate the dominant frequency.

- **powspec**: These lightcurves have some (not particularly simple) periodicity. It's important for astronomers to be able to determine any periodic behavior because this may offer insights about the underlying physics of the X-ray source. The set of periodicities present in the spectrum plotted with respect to intensity (i.e., strength of signal vs. frequency) is called a power spectrum, and can be seen using **powspec**. **powspec** uses essentially the same information inputs as **lcurve**. *(This tool will be covered in greater detail in the "Analysis of GK Per Data from EXOSAT" tutorial.)*

- **Concatenation**: This is not a command but an extremely useful technique to use with **powspec**. There are situations where the lightcurve you're studying has so much noise that the peaks in the power spectrum that are real signals are hard to distinguish from those due to random noise, since they are about the same height. One way to fix this problem is to find the power spectrum of several different data sets of the *same* source taken at *different* times. The resulting power spectrum is the average of the power spectra from each data set. A real signal will likely appear in all sets, and so the average for that signal will be strong. Random errors will appear in some, but not all data sets, and they will have different frequencies in each. Thus the average at any one frequency will be very weak. Therefore, the signal will look significantly stronger relative to the noise after concatenation than it would before.

  The steps for concatenation are as follows:

  1. Run **lcurve** on each file (i.e. each data set) to generate a *.flc* file for each.

  2. Place all *.flc* files, one per line, in a text file (named '*powcat.txt'* for example).

  3. Start **powspec** and type **@<filename>** (*@powcat.txt* in this example), when prompted for the file.

Exercise: *Use **powspec** on the file d05479.lc, from the last exercise, and compare your estimated frequency to the calculated frequency. Use the following parameters when prompted:*
Ser. 1 filename: d05479.lc

Default window: -
Newbin time: 30
Newbins per interval: 2000
Intervals per Frame: 1
Rebin results?: 0
Output file: d05479.pow (**Note**: your powspec output file must always end with **.pow**)
Plot results?: yes
PGPLOT device: /xw

*Note the huge peak to the far left of the plot. Note also the PLT> prompt and realize that you can use something like **rescale x 0 1e-3** to rescale the plot. Determining the peak frequency will be impossible without rescaling.*

Exercise: *Look at the power spectrum of another measurement of AM Her. These files, EXOSAT ME sequence no. 167, are named d08040.lc and d08054.lc. First look at the lightcurves using **lcurve**, and then look at the spectrum of a single data set (just choose one). Then, examine the spectrum when you look at both lightcurves. Use concatenation to find the power spectrum of both lightcurves. Use concatenation to find the power spectrum of both lightcurves.*

- ROSAT **xselect** is the environment which allows you to examine ROSAT files (in addition to several other missions). The following is a list of commands to get you started with **xselect**:

  - **xselect**: invokes **xselect**, prompts for name of session

  - **read events**: enters data into the program. You are prompted for the file's name and location (path), and whether you'd like to change a mission. Whenever you wish to enter a new set of data, you can do it with this command.

  - **set device [/xw** or **/ps]**: changes output destination to either /xw or /ps, where /xw indicates XWindow, or printout to the screen, and /ps indicates postscript format to be printed out on paper, using Ghostview

  - **extract [curve** or **spectrum** or **image]**: extracts a light curve, energy spectrum or image of the current data file, depending on which of the three words follows the **extract** command

  - **plot [stuff]**: plots the current extracted object. For instance, you can plot an extracted lightcurve with **plot curve**.

  - **rescale xmin xmax**: serves the same purpose as it does in **lcurve** and **powspec**

  - **filter**: does no work. **filter** commands just enter filters selecting a particular region or time interval that will be applied to the next run of

the **extract** command.

- **clear**: used to clear any changes you have made to data, thus restoring the data to the way it was originally

- **show data**: can be used at any type to receive an output of your working directory and the names of the files you are using

- **list**: similar to the **ls** command. To list all the files in your current directory, you can type **list workdirectory** or **list datadirectory**.

A typical **xselect** session may go as follows. Suppose you want to view the lightcurve and energy spectrum of AM Her. You would go to the directory (that you've created) *c/ftools/amher/rp300067a00*. Then, type **xselect**, and call your session **sess1**. Now type **read events**. Type ./ for the directory, **rp300067a00_bas.fits** as the file, and **yes** to change the mission from ASCA to ROSAT. So far, your session would look something like this:

```
BASH.EXE-2.01$xselect

                ** XSELECT V1.4 **

Enter session name >[xsel] sess1

Notes: XSELECT set up for ASCA
Time keyword is TIME     in units of s
Default timing binsize =  16.000

sess1:ASCA >read events
> Enter the Event file dir >[./] ./
> Enter Event file list >[] rp300067a00_bas.fits
Got new mission: ROSAT
> Reset the mission? >[yes] yes

Notes: XSELECT set up for ROSAT
Keywords for time and pha are  TIME    PI
Units of time are  s
Default timing binsize =  16.000

Setting...
 IMAGE  keywords   = X    Y      with binning = 15
 WMAP  keywords   = DETX  DETY   with binning = 15
 Energy  keywords  = PI          with binning = 1

Getting Min and Max for Energy Column...
Got min and max for PI:  1    500

Could not get minimum time resolution of the data read
```

```
Number of files read in:  1

*****************Observation Catalogue***************

Data Directory is: c/ftools/amher/rp300067a00/
HK Directory is: c/ftools/amher/rp300067a00/

INSTRUME OBJECT   DATE   RA_NOM DEC_NOM ROR_NUM LIVETIME
1 PSPCB     AM HER 16/01/96 0.27E+03  0.50E+02  300067    0.11E+05

sess1:ROSAT-PSPC >
```

Now type **set device /xw.** At the subsequent prompt, type **extract curve**. Next type **plot curve**. (*Note: you may get an error message here that says "This program has performed an illegal function and will be shut down." Just click 'OK'. The program will not shut down.)* Then, upon maximizing the X-Server, you should see a light curve like that in Figure 10 (next page). Hit 'Control' and 'D' simultaneously to exit the **PLT>** prompt. When you are back to the **xselect** prompt, enter **extract spectrum** to get the energy spectrum of this source. Finally, type **plot spectrum** to get a plot of AM Her (see Figure 11). (You may want to try rescaling to find the best view of the plot.)

VIII. Ghostview® **********INCOMPLETE **********

Included in your FTOOLS package is a program called Ghostview that allows you to view and print files in postscript format (files with the extension **.ps**). Ghostview is placed on your computer upon installation of FTOOLS. Open Windows Explorer and under *Folders*, click on the letter of the drive on which FTOOLS is installed (most likely your **[C:]** drive). Next, click on the *Tools* menu, go to *Find*, and then to *Files or Folders*. Type **gsview**. After clicking *Find Now*, two files should appear in the lower field: **GSview21.zip** and **gsview.zip**. In order to use Ghostview, these files must be unzipped. (This process need only be done once, in order to initially install and set up Ghostview.)

To uncompress (unzip) these files, we will use a program called WinZip. If WinZip has not already been installed on your computer, the demo version can be downloaded for free from http://www.winzip.com; this version is sufficient for our purposes. Once you know that WinZip has been installed on your system, double click on **GSview21.zip**, which you located above. WinZip will automatically open. The first thing you will receive is a window explaining copyright information. Click on 'I Agree'. Now you will see a window with various buttons at the top and some files listed below them. Click once on the first file, and then hold down the shift key and click on the last file in the list. All of the files in the list should be highlighted. Click the *Extract* button. A window will appear in which you must enter the location in which you would like the files to be placed. In the *Folders/Drives* field, select [C:] and then your **FTOOLS** folder. Finally, click *Extract* again. If a window pops up asking you if you'd like to overwrite an existing file, click on *Yes to All*.

Go back to Windows Explorer and go to **c/ftools**. After scrolling down a bit, you'll see a file called **setup.exe**. Double click on it. A small window entitled 'GSview

Install-Introduction' will appear. Click 'Ok'. Next, a window entitled 'GSview Install-Copyright' will appear. Again, click 'Ok'. The next window you receive asks you for the base directory for GSview and Ghostscript. It should say **c:\gstools**. If it does not, type **c:\gstools** in the entry field. Click 'Ok'.

Figure 10: Lightcurve for AM Her

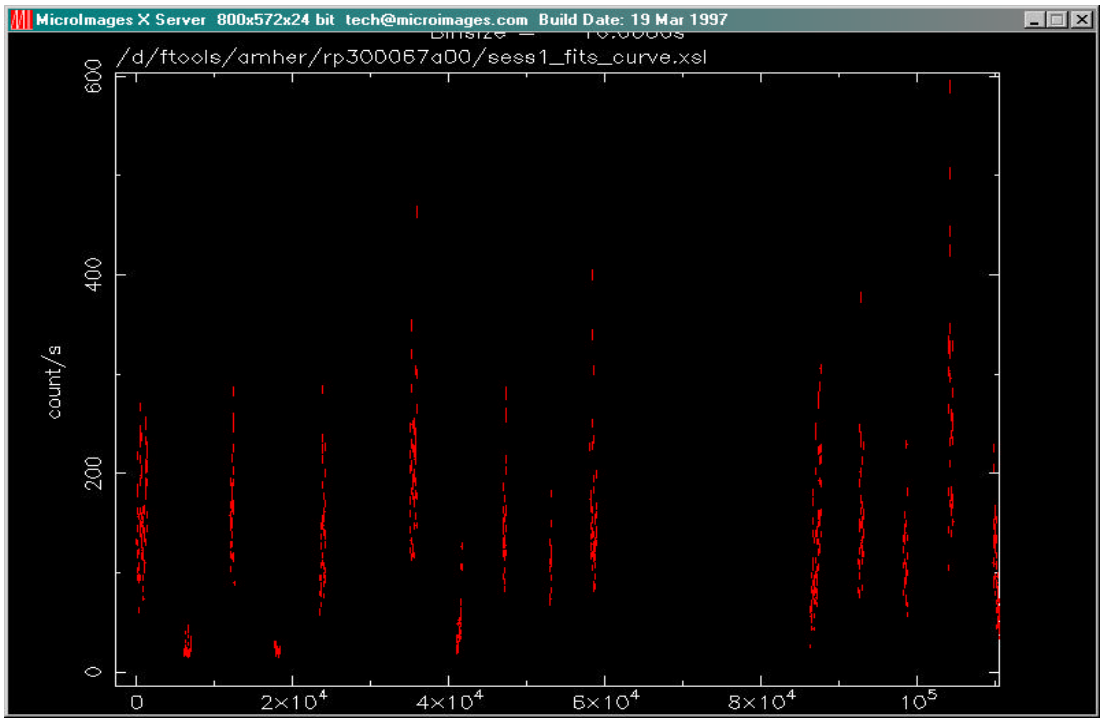
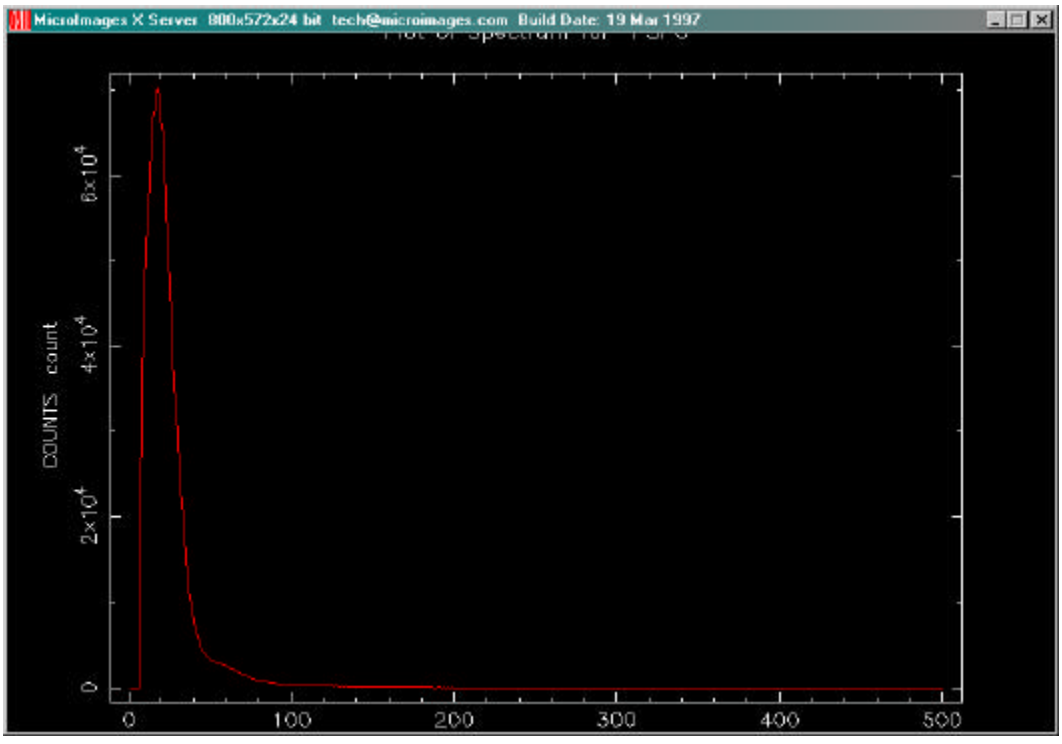


Figure 11: Energy Spectrum of AM Her



IX. Troubleshooting

This section contains some common problems that may be encountered when using the FTOOLS package and tutorials and also some known bugs in the software.

- Some of the FTOOLS functions will not work if you are using Windows 95. Windows 98 or NT is required to maximize FTOOLS' capabilites.

- In order for FTOOLS to work, you must have the X-Server running. Thus, before beginning each section, double click on the Microimages X-Server icon on your desk top to open the X-Server window, and then click on the '-' in the upper right hand corner of the window to minimize it.

- The *first time you use the X-Server in each session*, (by employing lcurve, xselect, powspec or fplot), you may get a box that pops up with an error message that says "This program has performed an illegal function and will be shut down…" Just hit 'OK'. The program will not be shut down. You may also get a message in your BASH window that says "To plot vs. Time(s) , please enter PGPLOT file/type:" Enter **/xw** (even if you already entered that earlier when you were prompted for a device), and the program should then proceed correctly, plotting the graph in your X-Server window and providing you with a **PLT>** prompt.

- In order to analyze a data set, you must be ***in*** the directory in which that file is located. Anytime you get a message saying something like "file not found", chances are, you're in the wrong directory. Likewise, if you would like to access a directory, like *gkper* for example, and you are in another directory outside of *gkper*, like *amher* or *cygx-1*, you must go back up to the parent directory, *ftools*, before you can type **cd gkper**. The directory *gkper* is located in *ftools*, not in *amher* or *cygx-1*: *amher* and *cygx-1* are other subdirectories of *ftools*.

- If, during the untarring and decompression process, you receive a message in the BASH window about and "EOF exception", go back and download the data again to get a different .tar file. This error may indicate that there is something wrong with the data.

- Filenames are NOT case sensitive under Cygwin BASH. Also important to note, the command **cd** IS case sensitive: thus the command **CD gkper** will not be recognized.

- Cygwin BASH is unstable under Windows 95/98 when running large and complex shell scripts. Particularly, after the FTOOLS hard disk version performs its additional setup steps when being run for the first time, you may find that some commands do not work, or that commands are echoed on a separate line before being executed. Currently, the only solution is to exit

from FTOOLS and restart. In severe cases, Cygwin BASH may cause the computer to lock up, in which case you'll have to reboot your computer.

- FTOOLS plotting using the X Windows driver does not work under Windows 95. For FTOOLS installed to run off a CD-ROM under Windows 98, plotting using the X Windows driver also does not work, but plotting does work under Windows 98 when FTOOLS is installed to run off a hard disk. Plotting works in all cases under Windows NT.

- The Windows version of fv can only print line plots. Any images in a graph being printed will be ignored.