

DXCluster

- ▶ [Home](#)
- ▶ [DXSpider Software](#)
- ▶ [DXSpider Bug Track](#)
- ▶ [Mailing Lists](#)
- ▶ [Technical Information](#)
- ▶ [Search](#)
- ▶ [FAQ-O-Matic](#)
- ▶ [Contact](#)
- ▶ [DX Links](#)

Visitor number

22259

Powered by



Sponsored by

THORCOM

Callbook Data from [QRZ](#)

Call:

This project is hosted at

[Sourceforge](#)

w w w . d x c l u s t e r . o r g

The web site for Amateur Radio DXcluster operators, development of DX cluster protocols and cluster software

You can try it out at [WR3D](#) (fast) [GB7DJK](#) (slower reference site). Log in with your callsign.

Unfortunately, because there seems to be a divergence of view between various telnet clients as to whether they echo or not, I have set the default to be 'remote echo'. If you use one of those telnet clients that actually obeys the rfcs (eg a unix telnet) you may feel happier doing a set/noecho after you have logged in.

This site is best viewed at 800 x 600 resolution

DXSpider

Dirk Koopman G1TLH

Last modified: Sun Apr 1 18:21:18 BST 2001

Introduction

The DXSpider DX cluster system is written in [perl 5.004](#) running under [Linux](#) as an exercise in self-training for both protocol research and teaching myself [perl](#).

What is a DX Cluster?

A DX Cluster is a means for Amateur (Ham) Radio operators to tell each other, in realtime, about DX stations (other interesting or rare Amateur Radio stations all over the world).

To quote what is probably the most comprehensive source of DX Cluster related information, the [DX PacketCluster WebNet](#), a Cluster is:-

One station is set up with PacketCluster and is linked to one or more other stations who have installed the software. These nodes when connected are called a cluster. Clusters are connected to clusters, expanding the network. Individual users connect to the nodes on a frequency different from what the node stations are linked on. Users are capable of announcing DX spots and related announcements, send personal talk messages, send and receive mail messages, search and retrieve archived data, and access data from information databases among its many features.

It's a rather specialised (and not as robust) form of IRC really.

The original package runs under DOS and was created by Dick Newell AK1A but is no longer under active development. Most replacements also run under DOS and/or are closed source. I wanted something in open source (so I don't have to do *all* the work) and for Linux.

This document will contain all the instructions for its installation and use - eventually...

Contents

1. [Download](#) the software and any patches.
2. DXSpider [Installation](#) manual.
3. DXSpider [Administration](#) manual.
4. DXSpider [User](#) manual.
5. Peruse the [FAQ](#)
6. If you are on the bleeding edge, see the [CVS Addendum](#).
7. Explaining the [client](#) programs.

8. [Periodic](#) jobs, e.g. starting connection to other clusters.
9. [Programming](#) new commands or altering existing ones.
10. [Local](#) customisation of the cluster daemon.

The DXSpider Administration Manual v1.49

Ian Maude, G0VGS, (g0vgs@ea1dav.net)

February 2002 revision 1.3

A reference for SysOps of the DXSpider DXCluster program.

1. Routing and Filtering

- [1.1 Introduction](#)
- [1.2 Route Filters](#)
- [1.3 The node default filter](#)
- [1.4 General route filtering](#)
- [1.5 General filter rules](#)
- [1.6 Types of filter](#)
- [1.7 Filter options](#)
- [1.8 Default filters](#)
- [1.9 Advanced filtering](#)
- [1.10 Basic hop control](#)
- [1.11 Hop Control on Specific Nodes](#)
- [1.12 Isolating networks](#)

2. Other filters

- [2.1 Filtering Mail](#)
- [2.2 Filtering words from text fields in Announce, Talk and DX spots](#)
- [2.3 Stopping \(possibly bad\) DX Spots from Nodes or Spotters](#)

3. Mail

- [3.1 Personal mail](#)
- [3.2 Bulletin mail](#)
- [3.3 Forward.pl](#)
- [3.4 The msg command](#)
- [3.5 Message status](#)
- [3.6 Filtering mail](#)
- [3.7 Distribution lists](#)
- [3.8 BBS interface](#)

4. Scripts

5. Databases

- [5.1 Creating databases](#)
- [5.2 Importing databases](#)
- [5.3 Checking available databases](#)
- [5.4 Looking up databases](#)
- [5.5 Removing databases](#)

6. Information, files and useful programs

- [6.1 MOTD](#)
- [6.2 MOTD_NOR](#)
- [6.3 Downtime message](#)
- [6.4 Other text messages](#)
- [6.5 The Aliases file](#)
- [6.6 Console.pl](#)
- [6.7 Updating kepler data](#)
- [6.8 The QRZ callbook](#)

7. Security

- [7.1 Registration](#)
- [7.2 Passwords](#)

8. CVS

9. The DXSpider command set

- [9.1 accept/announce \(0\)](#)
- [9.2 accept/announce \(extended for sysops\) \(8\)](#)
- [9.3 accept/route \(8\)](#)
- [9.4 accept/spots \(0\)](#)
- [9.5 accept/spots \(extended for sysops\) \(8\)](#)
- [9.6 accept/wcy \(0\)](#)
- [9.7 accept/wcy \(extended for sysops\) \(8\)](#)
- [9.8 accept/wwv \(0\)](#)
- [9.9 accept/wwv \(extended for sysops\) \(8\)](#)
- [9.10 announce \(0\)](#)
- [9.11 announce full \(0\)](#)
- [9.12 announce sysop \(5\)](#)
- [9.13 apropos \(0\)](#)
- [9.14 bye \(0\)](#)
- [9.15 catchup \(5\)](#)
- [9.16 clear/announce \(8\)](#)
- [9.17 clear/route \(8\)](#)
- [9.18 clear/spots \(0\)](#)
- [9.19 clear/spots \(extended for sysops\) \(8\)](#)
- [9.20 clear/wcy \(0\)](#)
- [9.21 clear/wcy \(extended for sysops\) \(8\)](#)
- [9.22 clear/wwv \(0\)](#)
- [9.23 clear/wwv \(extended for sysops\) \(8\)](#)
- [9.24 connect \(5\)](#)
- [9.25 dbavail \(0\)](#)

- [9.26 dbcreate \(9\)](#)
- [9.27 dbimport \(9\)](#)
- [9.28 dbremove \(9\)](#)
- [9.29 dbshow \(0\)](#)
- [9.30 debug \(9\)](#)
- [9.31 delete/user \(9\)](#)
- [9.32 directory \(0\)](#)
- [9.33 directory \(extended for sysops\) \(5\)](#)
- [9.34 disconnect \(8\)](#)
- [9.35 dx \(0\)](#)
- [9.36 export \(9\)](#)
- [9.37 export_users \(9\)](#)
- [9.38 filtering \(0\)](#)
- [9.39 forward/latlong \(8\)](#)
- [9.40 forward/opername \(1\)](#)
- [9.41 help \(0\)](#)
- [9.42 init \(5\)](#)
- [9.43 kill \(0\)](#)
- [9.44 kill \(5\)](#)
- [9.45 kill full \(5\)](#)
- [9.46 kill/expunge \(6\)](#)
- [9.47 links \(0\)](#)
- [9.48 load/aliases \(9\)](#)
- [9.49 load/badmsg \(9\)](#)
- [9.50 load/badwords \(9\)](#)
- [9.51 load/bands \(9\)](#)
- [9.52 load/cmd_cache \(9\)](#)
- [9.53 load/forward \(9\)](#)
- [9.54 load/messages \(9\)](#)
- [9.55 load/prefixes \(9\)](#)
- [9.56 merge \(5\)](#)
- [9.57 msg \(9\)](#)
- [9.58 pc \(8\)](#)

- [9.59 ping \(1\)](#)
- [9.60 rcmd \(1\)](#)
- [9.61 read \(0\)](#)
- [9.62 read \(extended for sysops\) \(5\)](#)
- [9.63 reject/announce](#)
- [9.64 reject/announce \(extended for sysops\) \(8\)](#)
- [9.65 reject/route \(8\)](#)
- [9.66 reject/spots \(0\)](#)
- [9.67 reject/spots \(extended for sysops\) \(8\)](#)
- [9.68 reject/wcy \(0\)](#)
- [9.69 reject/wcy \(extended for sysops\) \(8\)](#)
- [9.70 reject/wwv \(0\)](#)
- [9.71 reject/wwv \(extended for sysops\) \(8\)](#)
- [9.72 reply \(0\)](#)
- [9.73 send \(0\)](#)
- [9.74 set/address \(0\)](#)
- [9.75 set/announce \(0\)](#)
- [9.76 set/arcluster \(5\)](#)
- [9.77 set/baddx \(8\)](#)
- [9.78 set/badnode \(6\)](#)
- [9.79 set/badspotter \(8\)](#)
- [9.80 set/badword \(8\)](#)
- [9.81 set/beep \(0\)](#)
- [9.82 set/bbs \(5\)](#)
- [9.83 set/clx \(5\)](#)
- [9.84 set/debug \(9\)](#)
- [9.85 set/dx \(0\)](#)
- [9.86 set/dxgrid \(0\)](#)
- [9.87 set/dxnet \(5\)](#)
- [9.88 set/echo \(0\)](#)
- [9.89 set/email \(0\)](#)
- [9.90 set/here \(0\)](#)
- [9.91 set/homenode \(0\)](#)

- [9.92 set/hops \(8\)](#)
- [9.93 set/isolate \(9\)](#)
- [9.94 set/language \(0\)](#)
- [9.95 set/location \(0\)](#)
- [9.96 set/sys_location \(9\)](#)
- [9.97 set/logininfo \(0\)](#)
- [9.98 set/lockout \(9\)](#)
- [9.99 set/name \(0\)](#)
- [9.100 set/node \(9\)](#)
- [9.101 set/obscount \(9\)](#)
- [9.102 set/page \(0\)](#)
- [9.103 set/password \(0\)](#)
- [9.104 set/password \(9\)](#)
- [9.105 set/pinginterval \(9\)](#)
- [9.106 set/privilege \(9\)](#)
- [9.107 set/spider \(5\)](#)
- [9.108 set/sys_qra \(9\)](#)
- [9.109 set/qra \(0\)](#)
- [9.110 set/qth \(0\)](#)
- [9.111 set/register \(9\)](#)
- [9.112 set/talk \(0\)](#)
- [9.113 set/wcy \(0\)](#)
- [9.114 set/wwv \(0\)](#)
- [9.115 set/wx \(0\)](#)
- [9.116 show/baddx \(1\)](#)
- [9.117 show/badnode \(6\)](#)
- [9.118 show/badspotter \(1\)](#)
- [9.119 show/badword \(1\)](#)
- [9.120 show/configuration \(0\)](#)
- [9.121 show/configuration/node \(0\)](#)
- [9.122 show/connect \(1\)](#)
- [9.123 show/date \(0\)](#)
- [9.124 show/debug \(9\)](#)

- [9.125 show/dx \(0\)](#)
- [9.126 show/dxcc \(0\)](#)
- [9.127 sh/dxstats \(0\)](#)
- [9.128 show/files \(0\)](#)
- [9.129 show/filter \(0\)](#)
- [9.130 show/filter \(extended for sysops\) \(5\)](#)
- [9.131 show/hfstats \(0\)](#)
- [9.132 show/hftable \(0\)](#)
- [9.133 show/hops \(8\)](#)
- [9.134 show/isolate \(1\)](#)
- [9.135 show/lockout \(9\)](#)
- [9.136 show/log \(8\)](#)
- [9.137 show/moon \(0\)](#)
- [9.138 show/muf \(0\)](#)
- [9.139 show/newconfiguration \(0\)](#)
- [9.140 show/newconfiguration/node \(0\)](#)
- [9.141 show/node \(1\)](#)
- [9.142 show/prefix \(0\)](#)
- [9.143 show/program \(5\)](#)
- [9.144 show/qra \(0\)](#)
- [9.145 show/qrz \(0\)](#)
- [9.146 show/registered \(9\)](#)
- [9.147 show/route \(0\)](#)
- [9.148 show/satellite \(0\)](#)
- [9.149 show/sun \(0\)](#)
- [9.150 show/time \(0\)](#)
- [9.151 show/vhfstats \(0\)](#)
- [9.152 show/vhftable \(0\)](#)
- [9.153 show/wcy \(0\)](#)
- [9.154 show/wwwv \(0\)](#)
- [9.155 shutdown \(5\)](#)
- [9.156 spoof \(9\)](#)
- [9.157 stat/db \(5\)](#)

- [9.158 stat/channel \(5\)](#)
- [9.159 stat/msg \(5\)](#)
- [9.160 stat/route_node \(5\)](#)
- [9.161 stat/route_user \(5\)](#)
- [9.162 stat/user \(5\)](#)
- [9.163 sysop \(0\)](#)
- [9.164 talk \(0\)](#)
- [9.165 type \(0\)](#)
- [9.166 who \(0\)](#)
- [9.167 wx \(0\)](#)
- [9.168 wx \(enhanced for sysops\) \(5\)](#)

[Next](#) [Previous](#) [Contents](#)

The DXSpider User Manual v1.49

Ian Maude, G0VGS, (g0vgs@ea1dav.net)

December 2001 revision 4.1

A complete reference for users of the DXSpider DXCluster program.

1. [Introduction](#)

- [1.1 What is a DX Cluster?](#)
- [1.2 So what is DXSpider?](#)

2. [Logins and logouts.](#)

- [2.1 AX25 logins.](#)
- [2.2 Netrom logins.](#)
- [2.3 Telnet logins.](#)
- [2.4 Logouts.](#)

3. [Setting your personal details.](#)

4. [Getting and posting DX.](#)

- [4.1 Receiving DX.](#)
- [4.2 Posting DX.](#)

5. [Headings and propagation](#)

- [5.1 Sun](#)
- [5.2 Moon](#)
- [5.3 Heading](#)

6. Announcements.

- [6.1 Making announcements.](#)
- [6.2 Listing announcements.](#)

7. Nodes and users.

8. Talk mode.

9. Mail.

- [9.1 The "directory" command.](#)
- [9.2 Reading mail.](#)
- [9.3 Sending mail.](#)
- [9.4 Replying to mail.](#)
- [9.5 Deleting mail](#)

10. Filtering (From version 1.45)

- [10.1 General filter rules](#)
- [10.2 Types of filter](#)
- [10.3 Filter options](#)
- [10.4 Advanced filtering](#)

11. Hints, tips and common questions.

12. The DXSpider command reference

- [12.1 accept/announce](#)
- [12.2 accept/spots](#)
- [12.3 accept/wcy](#)
- [12.4 accept/wwv](#)
- [12.5 announce](#)
- [12.6 announce full](#)
- [12.7 apropos](#)

- [12.8 bye](#)
- [12.9 clear/spots](#)
- [12.10 dbavail](#)
- [12.11 dbshow](#)
- [12.12 directory](#)
- [12.13 dx](#)
- [12.14 help](#)
- [12.15 kill](#)
- [12.16 links](#)
- [12.17 read](#)
- [12.18 reject/announce](#)
- [12.19 reject/spots](#)
- [12.20 reject/wcy](#)
- [12.21 reject/wwv](#)
- [12.22 reply](#)
- [12.23 send](#)
- [12.24 set/address](#)
- [12.25 set/announce](#)
- [12.26 set/beep](#)
- [12.27 set/dx](#)
- [12.28 set/dxgrid](#)
- [12.29 set/echo](#)
- [12.30 set/here](#)
- [12.31 set/homenode](#)
- [12.32 set/language](#)
- [12.33 set/location](#)
- [12.34 set/logininfo](#)
- [12.35 set/name](#)
- [12.36 set/page](#)
- [12.37 set/qra](#)
- [12.38 set/qth](#)
- [12.39 set/talk](#)
- [12.40 set/wcy](#)

- [12.41 set/wwv](#)
- [12.42 set/wx](#)
- [12.43 show/configuration](#)
- [12.44 show/configuration/node](#)
- [12.45 show/date](#)
- [12.46 show/dx](#)
- [12.47 show/dxcc](#)
- [12.48 show/files](#)
- [12.49 show/filter](#)
- [12.50 show/moon](#)
- [12.51 show/muf](#)
- [12.52 show/prefix](#)
- [12.53 show/qra](#)
- [12.54 show/qrz](#)
- [12.55 show/route](#)
- [12.56 show/satellite](#)
- [12.57 show/sun](#)
- [12.58 show/time](#)
- [12.59 show/wcy](#)
- [12.60 show/wwv](#)
- [12.61 sysop](#)
- [12.62 talk](#)
- [12.63 type](#)
- [12.64 who](#)
- [12.65 wx](#)

[Next](#) [Previous](#) [Contents](#)

1. Introduction

1.1 What is a DX Cluster?

A DX Cluster is a packet node where DX chasers on any band or mode can post rare or interesting stations that they have worked or heard. Of course other people are doing the same thing too, so you can find new DX as well as telling others about the stations you have worked. Clusters tend to be linked to each other so that the amount of people using them is increased, thereby increasing the amount of posted DX. Other information can be found on clusters such as on-line call books, mail etc. You can talk to other stations connected to the cluster network too, in real time, whether at the node you are logged into or on another node connected to the network. You can also use converse mode, where several stations can talk to each other in the same way. Of course, the DX is still posted to you all the while!

1.2 So what is DXSpider?

PacketCluster nodes have been around since roughly 1985. The original PacketCluster idea came from Dick Newell, AK1A, and ran under DOS. In about 1992 Dick stopped the development of the PacketCluster software for amateur radio. Many systems are still using this relatively old DOS software today.

There are several new compatible cluster programs around now, including DXSpider. DXSpider is a clone of PacketCluster software that runs under the Linux operating system. Linux is fast becoming the choice for amateur radio stations because of it's flexibility, reliability and the lack of the memory limitations of DOS. Linux supports multitasking and is also multiuser. It has support for AX25, ROSE, NetROM and TCPIP built in, making it the ideal choice for amateur radio. It is also totally free!

DXSpider was conceived and begun in 1998 by Dirk Koopman, G1TLH as an exercise in perl programming. It has developed rapidly and today is a very powerful cluster program. It was designed to be totally compatible with the AK1A program, although several commands have been extended to try and improve functionality.

This manual is designed to help you become familiar with the commands that DXSpider supports and to help you get the best from the program so you can enjoy working that rare DX! As DXSpider is being improved all the time, commands will be added as time goes by, so make sure you have the most upto date version of this manual. The latest version will always be included with the cluster program so if you are unsure, simply ask your sysop. The manual will also be available on the wesite.

2. Logins and logouts.

You might not think that there is a lot of point of including a section on how to log in and out of DXSpider. However, you would be suprised at the difficulties some people have in simply getting in and out of the cluster!

There are several ways a login might be achieved, dependant on how the sysop has DXSpider configured. It is impossible for me to cover all variations but here are the basic ones.

2.1 AX25 logins.

Simplicity itself. The usual **CONNECT** command will log you straight into the cluster and you will not have to do anything else. Obviously, you will have to connect to the correct callsign. Some nodes use an SSID with their call so you would have to add that.

Examples:

```
connect GB7MBC
connect GB7MBC-1
```

2.2 Netrom logins.

There are several possibilities here, dependant on how the sysop has configured his system. If you are connecting via netrom then you are most probably connecting from another station. Listing the nodes in that station with the **NODES** command will tell you what callsign or netrom alias to connect to. Then just issue the connect command from there. It is possible that the netrom alias may connect you to a node with an alias for the cluster, such as DXC. Just type this and you will be connected.

Example:

```
connect MBCDX
```

2.3 Telnet logins.

Because DXSpider runs under the Linux operating system, it is possible to make a direct telnet connection into the cluster. With telnet connections, the source callsign is not seen by DXSpider, so you will be asked to login with your callsign. To telnet to DXSpider, you would connect to a specific port. There is no standard at the moment for a cluster telnet port but ask the sysop if you are unsure.

Example:

```
telnet gb7mbc 8000
```

All the above are possible ways of connecting to a DXSpider cluster. You may have some or all of these available to you. There may be one or two additional ways to connect dependant on the network local to you. However I am sure you get the idea.

2.4 Logouts.

Logging out can be done by simply issuing the standard **BYE** command.

You could also send a disconnect if you are using AX25, or a **CLOSE** command if you are connected via telnet. If you do not log out gracefully using one of the above commands, you may find you are unable to get a full connect next time. This may also happen if a NETRom connection drops. You may get connected, but nothing else will happen because the program thinks you are still connected and will not let you connect twice under the same call. However you could reconnect by adding a number to the end of your call, for example G0YLM2. This can be done by either altering your MYCALL setting in the TNC or by altering your program configuration.

[Next](#) [Previous](#) [Contents](#)

[Next](#) [Previous](#) [Contents](#)

3. Setting your personal details.

Once logged in to the cluster, you should set your details so that anybody who wishes to contact you can find out who and where you are. There are four items to set, your name, qth, location and home node. Setting these details also allows the use of the SHOW/HEADING and SHOW/SUN commands from within the cluster. Unless you set your QTH and location, these commands cannot function. Once you have set your name, DXSpider will greet you with it next time you login. Your QTH setting is where you live and it is a good idea to add your locator to this as the location setting is converted to latitude and longitude once inputted. You can actually set your location in latitude/longitude or as a locator. Setting your home node will tell the program where you wish mail to be sent to you.

Examples:

```
set/name Ian
set/qth Morecambe, Lancashire I084NB
set/location 48 34 n 12 12 e
set/qra I084NB
set/home gb7mbc
```

[Next](#) [Previous](#) [Contents](#)

4. Getting and posting DX.

When all is said and done, this is the main function of a DX cluster. In its simplest form you can just connect to the node and you will start to receive DX spots almost immediately! You can check on recent postings in either a general manner or on a particular band or mode. You can even check DX by callsign or a fragment of a callsign. Of course, once you get the hang of things, it is expected that you start posting some yourself! After all, there would be no clusters if people did not post DX and you get the added thrill of the hunt!

4.1 Receiving DX.

As we have already said, it is possible just to connect to the cluster and you will receive spots automatically. However, you may wish to check on spots just posted. Maybe you wish to see if a particular band is open or if a certain callsign is active, perhaps a DXpedition. The command to do this is **SHOW/DX**. Without any other arguments, this command will output the last 10 spots posted. It is possible to look at more than this, for example the last 20 or 50 spots, by adding the number to the command. You can make it even more specific by adding a band in either wavelength or frequency, and/or any additional information such as QSL details.

Examples:

```
show/dx
show/dx 5
show/dx 20
```

will show the last 10, 5 and 20 spots received by the cluster respectively.

Examples

```
show/dx on 20m
show/dx 10 on 20m
show/dx 20 on 20m
```

will show the last 5, 10 or 20 spots on 20 metres only.

It is also possible to check for certain callsigns, or fragments of callsigns in the same way.

Examples:

```
show/dx g0vgs
show/dx 10 g0vgs
```

would show the last 5 or 10 dx spots containing the callsign g0vgs.

You can check for DX by offset and also by specifying a comment to search for.

Examples:

```
show/dx 30-40
show/dx 14000-14033
show/dx iota
```

would show the spots that arrived between 30 and 40 spots ago and any spots with the word *iota* in the comment field. The case of the comment is not important.

Checking DX posted on a certain day is possible too. All you have to do here is to specify how many days ago it was like this ...

Example:

```
show/dx day 30
```

It is of course possible to specify multiple arguments.

Example:

```
show/dx 20 prefix 9a on vhf day 30
```

This would show the last 20 spots posted by or about calls with the prefix 9a on vhf 30 days ago.

As you can see the **SHOW/DX** command is very flexible, so if you are not sure whether something will work or not, try it and see! More information can be found in the Command Set section.

4.2 Posting DX.

To post DX you use the **DX** command. The syntax is shown below.

Example:

```
dx (frequency) (callsign) (remarks)
```

Where frequency is in kilohertz and the callsign is the callsign of the station you have worked or heard, (ie not your own callsign!). The remarks section allows you to add information like the operators name or perhaps a location. Actually DXSpider will allow the frequency and callsign fields to be entered in any order.

Example:

```
dx 14004 pa3ezl OP Aurelio 599
```

In fact, all the following will give the same result...

```
dx 14004 pa3ezl OP Aurelio 599  
dx pa3ezl 14004 OP Aurelio 599  
dx pa3ezl 14.004 OP Aurelio 599
```

This posting, or callout as it is known, will be forwarded to all other connected stations both at the cluster you are connected to and other active clusters in the network. The callout will also be sent to you as proof of receipt.

[Next](#) [Previous](#) [Contents](#)

5. Headings and propagation

There are three commands in DXSpider to help you get the best DX possible. These are **SHOW/SUN**, **SHOW/MOON** and **SHOW/HEADING**. These commands will only work for you if you have entered your personal details. They use your entered location as a reference, so if you have not entered it or have entered it incorrectly they will not return the correct information.

5.1 Sun

The **SHOW/SUN** command can be used in three different ways. It can be used to show sunrise and sunset times for your own station, a particular callsign or a prefix.

Example:

```
show/sun
```

The output from this would look something like this ..

sh/sun				
Location	Rise	Set	Azim	Elev
G0VGS Morecambe, Lancashire	07:08Z	17:39Z	205.3	24.1

```
sh/sun 5b4
```

would look like this ...

sh/sun 5b4				
Location	Rise	Set	Azim	Elev
5B Cyprus-5B	04:23Z	15:40Z	244.0	18.8

You can also specify multiple arguments like this ...

```
sh/sun gw4veq 5b4ab zs
```

and then the output would look like this ...

sh/sun gw4veq 5b4ab zs				
Location	Rise	Set	Azim	Elev
GW4VEQ Brungwan, Isle of Anglesey IO	07:14Z	17:48Z	204.9	24.9
5B Cyprus-5B	04:23Z	15:40Z	244.5	18.3
ZS So-Africa-ZS1-ZS	04:31Z	17:28Z	289.9	41.3
ZS So-Africa-ZS5-ZS	03:44Z	16:34Z	278.5	32.0
ZS So-Africa-ZS6-ZS	03:59Z	16:42Z	277.6	35.0

5.2 Moon

The **SHOW/MOON** command works in the same way as the **SHOW/SUN** command. This program however, calculates the rise and set times of the moon for a prefix or callsign, together with the current azimuth and elevation of the sun at these locations.

Example:

```
show/moon ea
```

The output from this command would look like this ..

```
sh/moon ea
Location
EA      Spain-EA
Rise    Set      Azim   Elev   RGain dB
08:15Z  20:09Z    257.2   9.5    -0.6
```

You can see that the output is similar to the **SHOW/SUN** command, with slightly different fields.

5.3 Heading

The **SHOW/HEADING** command works in the same way as the **SHOW/SUN** and **SHOW/MOON** commands but outputs beam headings for a specified callsign or prefix. Reciprocal beam headings are also calculated.

Example

```
show/heading zl
```

The output from this command would look like this ..

```
sh/heading zl
ZL New-Zealand-ZL1-ZL: 7 degs - dist: 11238 mi, 18087 km Reciprocal heading: 355 degs
ZL New-Zealand-ZL2-ZL: 9 degs - dist: 11540 mi, 18574 km Reciprocal heading: 353 degs
ZL New-Zealand-ZL3-ZL: 19 degs - dist: 11634 mi, 18724 km Reciprocal heading: 345
degs
ZL New-Zealand-ZL4-ZL: 34 degs - dist: 11783 mi, 18963 km Reciprocal heading: 332
degs
```

[Next](#) [Previous](#) [Contents](#)

6. Announcements.

6.1 Making announcements.

Occasionally, you may wish to post something that does not fall into the normal parameters for a DX callout. You may wish to tell everybody connected that 10 FM is open for example, or ask if anyone knows the QSL manager for a certain callsign etc. You can do this using the **ANNOUNCE** command.

Example:

```
announce 10 FM is open in I084NB to europe.
```

That would let everyone know locally that this was the case, however it would not be forwarded to other nodes connected. To send announcements to other connected nodes as well, you would use the **FULL** extension.

Example:

```
announce full Anyone seen EA7WA today?
```

Be cautious in your use of announce full. There are many other tools available to get the information you require and the judicious use of this command is frowned upon as it creates a great deal of traffic across the network.

6.2 Listing announcements.

You can list previous announcements in the standard format with the **SHOW** command. As before you can list just the last 5 or as many as you wish.

Example:

```
show/announcements  
show/announcements 10
```

7. Nodes and users.

You can check which nodes are connected in the network, who is logged on locally, who is logged on at all the nodes or even just one node in particular. This is handy if you wish to see whether a friend is connected at the node they use. To see who is connected to the nodes, the **SHOW/CONFIGURATION** command is used.

Example:

```
show/configuration
show/configuration/nodes
show/configuration (node_call)
show/configuration (prefix)
```

The first of our three examples would output something like this,

```
sh/c
Node          Callsigns
EI5TCR        (7 users)
GB7ADX
GB7BAA        G4FPV          G8TIC
GB7BIG        (GD0TEP)       GD3UMW
GB7BPQ        (G0INA)        G0RCI          G3AKU          G3OCA
(GB7CDX)      G3JNB          G4ALR
GB7DJK        G0FYD          G0REK          G1TLH          G4PEL          G4S0Z
               G4TVR          G7SQW          K8AZ           M0CTQ-1        MM1CXE-10
               ON7WP
GB7DXA        G0RDI          G8SJP
GB7DXC        (G0HDB)        G0WFK          (G1FYC)        G3KWK          G3LME
               G30IL          G4BGW          G4FUJ          (G4PDQ)        GW7SMV
GB7DXE        G1NNB
(GB7DXG)      GU6EFB        GU7DHI
GB7DXK        G1NTW          G3IBI          G3NSM          G3XAQ          G4CUS
               G4XQY          G7GAN
GB7DXM        G1EUC          G3GAF          G3LAS          G4ZTR          G8WXU
               M0BCT          M1EMF
```

You will notice that EI5TCR is showing only that it has 7 users and not giving the actual callsigns. This means that this node is on a limited protocol of some kind, probably because of a contest situation where slow radio links can block up and fail if too much information is sent between nodes.

The second example would just show the nodes connected in the network, like this,

```
sh/c/n
Node          Callsigns
GB7BAA        GB7BAA        GB7BPQ        (GB7CDX)      GB7DJK        GB7DXA
               GB7DXC        GB7DXE        (GB7DXG)      GB7DXK        GB7DXL
               GB7DXM        GB7DXS        GB7IPT        GB7MRS        GB7UJS
               GB7YDX        KL7G          N2TLY        (ON0DXK)
GB7BIG        EI5TCR        GB7ADX        GB7BIG        GB7UDX
GB7MBC
```

PA4AB-14	PA4AB-14
PI4TUE-8	PI4TUE-8

If we insert the node_call pi4tue-8 into the third example, then this would be the output,

Node	Callsigns				
PI4TUE-8	9A1CMA-6	DF6PW	DL4FAY	DL4KAK	DL4WF-2
	F5NOD-2	F5PAC	IZ0CSR	N6CR	OH2BLD
	ON1LVL-13	ON4CBT	ON4CJP	ON5DXL-14	ON7NQ
	PA0RCT	PA3DYS	PA3FDO	PA5KW-4	PI4TUE-9
	YT1XX				

As you can see, only the users of the requested node are shown.

You can also use a prefix to show only a group of nodes. For example you may only wish to see who is connected to the nodes in the UK. To do this simply use a prefix like this ...

show/configuration gb7

To show the locally connected users, the **SHOW/USERS** command is used

Example:

show/users

The output of this command would look like this,

Callsigns connected to GB7MBC			
G0JHC	G0NEI	G0VGS	G0VGS-2
G3JAG	G3OWO	G3UEU	G0YLM

[Next](#) [Previous](#) [Contents](#)

8. Talk mode.

You can send a single comment or start a dedicated talk session to another user by using the **TALK** command.

Talk mode is used to send a one line comment or greeting to a specific user connected either at your node or another in the network. You can also enter into a dedicated talk session with another user. Talks will be automatically forwarded to other nodes if the station you are talking to is not connected locally. You can find out who is connected by using the **SHOW/CONFIGURATION** command, (described earlier).

Examples:

```
talk g0rdi Having a good day Iain?
```

This example would send the line "Having a good day Iain?" to the user g0rdi but would leave you in normal mode.

```
talk g0rdi
Entering Talkmode, /EX to end, /<cmd> to run a command
Talk (G0RDI)>
```

As you can see, you can still run commands whilst in talk mode. Simply prefix the command with a / like /sh/dx and you will get the expected output. If the user is connected to another node, you may have to use a slightly extended version of the **TALK** command.

```
talk g0rdi > gb7djk
```

To exit talk mode, you issue the command **/ex**.

Whilst in talk mode you will still receive DX spots. This means that you can chat to a friend whilst working DX.

9. Mail.

You can send and receive both personal mail and bulletins with DXSpider quite easily.

9.1 The "directory" command.

To list mail you would use the **DIRECTORY** command. On its own, this command will output the last ten messages received by the node, either to or from yourself or bulletins such as "DX" and "ALL". As with other commands you can display more by specifying a number with the command. You should be aware that Spider will accept these commands by separating with either a slash or a space, so *dir new* and *dir/new* work in the same way.

Example:

```
directory
directory/20
directory 20
```

Of course most of the time you will only want to list new mail sent to you personally since your last login. However you might also like to check for general mail received by the node. In fact if there is new mail for you, the cluster will tell you when you login. You will also be informed if new mail arrives for you during the time you are logged in. Mail is not only sent to callsigns though. Mail can also be sent to subjects like "all" or "local" or "dx" etc. You can treat these bulletins in the same way as personal mail with the directory command.

Examples:

```
directory/new
directory/own
directory/own/10
directory/all
directory/from <call>
```

The last option will only show mail from a callsign if it was sent to you personally or was sent as a bulletin. There are some additional commands and these can be found in the DXSpider Command Reference section.

9.2 Reading mail.

The output of the **DIRECTORY** command could be something like this.

```
dir
20735      2      ALL      GW7SMV 21-Feb 1204Z REC 9E1S QSL TDY 50Mhz....
20823     308      UK       G0HDB 22-Feb 2334Z Help - which district code?
20824     105      ALL       W9AE 23-Feb 0349Z S0NY QSL address?
20825      2      UK       G0LRJ 23-Feb 0806Z QSL REC LZ2CJ/1.CARD NO-750.
20858      2      ALL       GW7SMV 24-Feb 0905Z REC S92DX QSL CARD TDY 50Mhz
20921     200      ALL       GM4FDM 27-Feb 2203Z Trip to VP8
20949     375      ALL       K0MN 27-Feb 0428Z ST0P cards are gd @ ARRL
20950      2      UK       G0LRJ 28-Feb 0835Z QSL REC SV9/IZ0CKJ/P EU-187.
```

20987	569	ALL	GD0TEP	1-Mar	1733Z	Portable contests
21076	2	ALL	G4AFJ	3-Mar	1743Z	kh6nd/kh5 qsl received
21184-p	599	GW4HAT	G0VGS	4-Mar	1518Z	Re: Time

The first column is the actual message number. If the message is a personal one to you, you will see a letter 'p' after this number. If the message has been read, there will be a '-' between the message number and the 'p'. This only works for personal messages. The next column shows the file size of the message. The third column shows the address the message was sent to and the next column shows who sent it. The rest is fairly self-explanatory.

Reading a message is as simple as typing read, followed by the message number that you wish to read.

Example:

```
read 25
```

will read message number 25. However the mail will be displayed in it's entirety unless you specify a page length. You can set your page length to any number you like and when the message reaches that number of lines you will get a prompt giving you options.

Example:

```
set/page 20
```

9.3 Sending mail.

Sending mail is done in the time honoured way. First you specify a recipient for the message, then you will be prompted for a subject. Once you have done this you will be asked to type your message. Please remember that there is no automatic word wrap, so unless you are using a client that wraps lines automatically, remember to hit return at the end of each line.

Example:

```
send m0azm
Enter Subject (30 characters):
See you Thursday
Enter Message /EX to send or /ABORT to exit
Hi Ian,
Just a quick note to say that I can make Thursday after all. The
appointment I had has cancelled so we are go!
Cheers
Ian
```

At this point you can either hit return and enter /ex to send the message or use /abort at any time to abort it. You can now display the message in the normal way. There are several send options and these are listed in the Command Set section. These include sending mail to several recipients and asking for a return receipt.

9.4 Replying to mail.

If mail is addressed to you or to a bulletin address, you can use the **REPLY** command to reply to it. Using this command, the subject will be automatically set for you as "Re: subject", whatever the subject was.

Example:

reply 2500

9.5 Deleting mail

To delete a message, you use the *delete* command. You can only delete messages sent to or received by yourself. Bulletins are dealt with automatically or by the sysop.

[Next](#) [Previous](#) [Contents](#)

10. Filtering (From version 1.45)

10.1 General filter rules

Upto v1.44 it was not possible for the user to set their own filters. From v1.45 though that has all changed. It is now possible to set filters for just about anything you wish. If you have just updated from an older version of DXSpider you will need to update your new filters. You do not need to do anything with your old filters, they will be renamed as you update.

There are 3 basic commands involved in setting and manipulating filters. These are *accept*, *reject* and *clear*. First we will look generally at filtering. There are a number of things you can filter in the DXSpider system. They all use the same general mechanism.

In general terms you can create a 'reject' or an 'accept' filter which can have up to 10 lines in it. You do this using, for example ...

```
accept/spots .....
reject/spots .....
```

where are the specific commands for that type of filter. There are filters for spots, wwv, announce, wcy and (for sysops) connects. See each different accept or reject command reference for more details.

There is also a command to clear out one or more lines in a filter. They are ...

```
clear/spots 1
clear/spots all
```

There is clear/xxxx command for each type of filter.

and you can check that your filters have worked by the command ...

```
show/filter
```

For now we are going to use spots for the examples, but you can apply the same principles to all types of filter.

10.2 Types of filter

There are two main types of filter, *accept* or *reject*. You can use either to achieve the result you want dependent on your own preference and which is more simple to do. It is pointless writing 8 lines of reject filters when 1 accept filter would do the same thing! Each filter has 10 lines (of any length) which are tried in order. If a line matches then the action you have specified is taken (ie reject means ignore it and accept means take it)

If you specify reject filters, then any lines that arrive that match the filter will be dumped but all else will be accepted. If you use an accept filter, then **ONLY** the lines in the filter will be accepted and all else will be dumped. For example if you have a single line *accept* filter ...

```
accept/spots on vhf and (by_zone 14,15,16 or call_zone 14,15,16)
```

then you will **ONLY** get VHF spots *from* or *to* CQ zones 14, 15 and 16.

If you set a reject filter like this ...

```
reject/spots on hf/cw
```

Then you will get everything *EXCEPT* HF CW spots. You could make this single filter even more flexible. For example, if you are interested in IOTA and will work it even on CW even though normally you are not interested in CW, then you could say ...

```
reject/spots on hf/cw and not info iota
```

But in that case you might only be interested in iota and say:-

```
accept/spots not on hf/cw or info iota
```

which achieves exactly the same thing. You should choose one or the other until you are comfortable with the way it works. You can mix them if you wish (actually you can have an accept AND a reject on the same line) but don't attempt this until you are sure you know what you are doing!

You can arrange your filter lines into logical units, either for your own understanding or simply convenience. Here is an example ...

```
reject/spots 1 on hf/cw  
reject/spots 2 on 50000/1400000 not (by_zone 14,15,16 or call_zone 14,15,16)
```

What this does is to ignore all HF CW spots and also rejects any spots on VHF which don't either originate or spot someone in Europe.

This is an example where you would use a line number (1 and 2 in this case), if you leave the digit out, the system assumes '1'. Digits '0'-'9' are available. This makes it easier to see just what filters you have set. It also makes it more simple to remove individual filters, during a contest for example.

You will notice in the above example that the second line has brackets. Look at the line logically. You can see there are 2 separate sections to it. We are saying reject spots that are VHF or above *APART* from those in zones 14, 15 and 16 (either spotted there or originated there). If you did not have the brackets to separate the 2 sections, then Spider would read it logically from the front and see a different expression entirely ...

```
(on 50000/1400000 and by_zone 14,15,16) or call_zone 14,15,16
```

The simple way to remember this is, if you use OR - use brackets. Whilst we are here CASE is not important. 'And BY_Zone' is just the same as 'and by_zone'.

As mentioned earlier, setting several filters can be more flexible than simply setting one complex one. Doing it in this way means that if you want to alter your filter you can just redefine or remove one or more lines of it or one line. For example ...

```
reject/spots 1 on hf/ssb
```

would redefine our earlier example, or

```
clear/spots 1
```

To remove all the filter lines in the spot filter ...

```
clear/spots all
```

10.3 Filter options

You can filter in several different ways. The options are listed in the various helpfiles for accept, reject and filter.

10.4 Advanced filtering

Once you are happy with the results you get, you may like to experiment.

The previous example that filters hf/cw spots and accepts vhf/uhf spots from EU can be written with a mixed filter, for example ...

```
rej/spot on hf/cw
acc/spot on 0/30000
acc/spot 2 on 50000/1400000 and (by_zone 14,15,16 or call_zone 14,15,16)
```

Note that the first filter has not been specified with a number. This will automatically be assumed to be number 1. In this case, we have said *reject all HF spots in the CW section of the bands but accept all others at HF. Also accept anything in VHF and above spotted in or by operators in the zones 14, 15 and 16*. Each filter slot actually has a 'reject' slot and an 'accept' slot. The reject slot is executed BEFORE the accept slot.

It was mentioned earlier that after a reject test that doesn't match, the default for following tests is 'accept', the reverse is true for 'accept'. In the example what happens is that the reject is executed first, any non hf/cw spot is passed to the accept line, which lets through everything else on HF. The next filter line lets through just VHF/UHF spots from EU.

[Next](#) [Previous](#) [Contents](#)

11. Hints, tips and common questions.

Q.These commands seem very long! Can I shorten them?

A.Of course you can and the usual abbreviations work. If you are unsure, try it.

Q.I am not sure if this command is correct. Can I cause any harm if I try it?

A.Do not be afraid to try a command to see if it will work, at the worst you will get an error message. If you require any help on a command, just type help followed by the command you want help on. Look at the "DXSpider Command Reference" section to see what help can be found.

Q.How should I use the announce command?

A.With respect. Use the command by all means, but please only use the "full" extension if absolutely necessary. It can create a LOT of messages passing between clusters.

Q.I like to be working in the shack while logged into the cluster but I can't be looking at the screen all the time. How can I be alerted when anything happens?

A.Use the **SET/BEEP** command. You can find information on this in the "DXSpider Command Reference" section.

Q.I got disconnected from the cluster and now I can't log back in again. What is wrong?

A.Probably the cluster thinks you are still logged on and will not let you reconnect using the same call to prevent loops. Try logging on again adding an ssid to your callsign as DXSpider treats G0YLM and G0YLM-1 as different users.

Q.How do I know if I have got the latest version of this user manual?

A.The latest and greatest will always be on the Website. It will also be included with every release of DXSpider. As always, if unsure, ask your sysop what version number is the latest.

12. The DXSpider command reference

Below is a complete list of commands available from the cluster prompt. These are listed below in alphabetical order.

12.1 accept/announce

accept/announce [0-9] <pattern> Set an accept filter line for announce

Create an 'accept this announce' line for a filter.

An accept filter line means that if the announce matches this filter it is passed onto the user. See **HELP FILTERS** for more info. Please read this to understand how filters work - it will save a lot of grief later on.

You can use any of the following things in this line:-

info <string>	eg: iota or qsl
by <prefixes>	eg: G,M,2
origin <prefixes>	
origin_dxcc <numbers>	eg: 61,62 (from eg: sh/pre G)
origin_itu <numbers>	
origin_zone <numbers>	
by_dxcc <numbers>	
by_itu <numbers>	
by_zone <numbers>	
channel <prefixes>	
wx 1	filter WX announces
dest <prefixes>	eg: 6MUK,WDX (distros)

some examples:-

```
acc/ann dest 6MUK
acc/ann 2 by_zone 14,15,16
(this could be all on one line: acc/ann dest 6MUK or by_zone 14,15,16)
```

or

```
acc/ann by G,M,2
```

This filter would only allow announces that were posted buy UK stations. You can use the tag 'all' to accept everything eg:

```
acc/ann all
```

but this probably for advanced users...

12.2 accept/spots

accept/spots [0-9] <pattern> Set an accept filter line for spots

Create an 'accept this spot' line for a filter.

An accept filter line means that if the spot matches this filter it is passed onto the user. See **HELP FILTERS** for more info. Please read this to understand how filters work - it will save a lot of grief later on.

You can use any of the following things in this line:-

freq <range>	eg: 0/30000 or hf or hf/cw or 6m,4m,2m
on <range>	same as 'freq'
call <prefixes>	eg: G,PA,HB9
info <string>	eg: iota or qsl
by <prefixes>	
call_dxcc <numbers>	eg: 61,62 (from eg: sh/pre G)
call_itu <numbers>	
call_zone <numbers>	
by_dxcc <numbers>	
by_itu <numbers>	
by_zone <numbers>	
origin <prefixes>	
channel <prefixes>	

For frequencies, you can use any of the band names defined in **SHOW/BANDS** and you can use a subband name like: cw, rtty, data, ssb - thus: hf/ssb. You can also just have a simple range like: 0/30000 - this is more efficient than saying simply: freq HF (but don't get too hung up about that)

some examples:-

```
acc/spot 1 on hf/cw
acc/spot 2 on vhf and (by_zone 14,15,16 or call_zone 14,15,16)
```

You can use the tag 'all' to accept everything, eg:

```
acc/spot 3 all
```

but this probably for advanced users...

12.3 accept/wcy

accept/wcy [0-9] <pattern> set an accept WCY filter

It is unlikely that you will want to do this, but if you do then you can filter on the following fields:-

by <prefixes>	eg: G,M,2
origin <prefixes>	
origin_dxcc <numbers>	eg: 61,62 (from eg: sh/pre G)
origin_itu <numbers>	
origin_zone <numbers>	
by_dxcc <numbers>	
by_itu <numbers>	
by_zone <numbers>	
channel <prefixes>	

There are no examples because WCY Broadcasts only come from one place and you either want them or not (see **UNSET/WCY** if you don't want them).

This command is really provided for future use.

See `HELP FILTER` for information.

12.4 accept/wwv

accept/wwv [**0-9**] **<pattern>** Set an accept WWV filter

It is unlikely that you will want to do this, but if you do then you can filter on the following fields:-

by <prefixes>	eg: G,M,2
origin <prefixes>	
origin_dxcc <numbers>	eg: 61,62 (from eg: sh/pre G)
origin_itu <numbers>	
origin_zone <numbers>	
by_dxcc <numbers>	
by_itu <numbers>	
by_zone <numbers>	
channel <prefixes>	

for example

```
accept/wwv by_zone 4
```

is probably the only useful thing to do (which will only show WWV broadcasts by stations in the US).

See `HELP FILTER` for information.

12.5 announce

announce **<text>** Send an announcement to local users

Send an announcement to LOCAL users only, where **<text>** is the text of the announcement you wish to broadcast

12.6 announce full

announce full **<text>** Send an announcement cluster wide

This command will send your announcement across the whole cluster network.

12.7 apropos

apropos **<string>** Search the help database

Search the help database for **<string>** (it isn't case sensitive), and print the names of all the commands that may be relevant.

12.8 bye

bye Exit from the cluster

This will disconnect you from the cluster

12.9 clear/spots

clear/spots [1|all] Clear a spot filter line

This command allows you to clear (remove) a line in a spot filter or to remove the whole filter.

If you have a filter:-

```
acc/spot 1 on hf/cw
acc/spot 2 on vhf and (by_zone 14,15,16 or call_zone 14,15,16)
```

and you say:-

```
clear/spot 1
```

you will be left with:-

```
acc/spot 2 on vhf and (by_zone 14,15,16 or call_zone 14,15,16)
```

If you do:

```
clear/spot all
```

the filter will be completely removed.

12.10 dbavail

dbavail Show a list of all the databases in the system

The title says it all really, this command lists all the databases defined in the system. It is also aliased to SHOW/COMMAND.

12.11 dbshow

dbshow <dbname> <key> Display an entry, if it exists, in a database

This is the generic user interface to the database to the database system. It is expected that the sysop will add an entry to the local Aliases file so that users can use the more familiar AK1A style of enquiry such as:

```
SH/BUCK G1TLH
```

but if he hasn't and the database really does exist (use DBAVAIL or SHOW/COMMAND to find out) you can do the same thing with:

```
DBSHOW buck G1TLH
```

12.12 directory

directory List messages

directory all List all messages

directory own List your own messages

directory new List all new messages

directory to <call> List all messages to <call>
directory from <call> List all messages from <call>
directory subject <string> List all messages with <string> in subject
directory <nn> List last <nn> messages
directory <from>-<to> List messages <from> message <to> message

List the messages in the messages directory.

If there is a 'p' one space after the message number then it is a personal message. If there is a '-' between the message number and the 'p' then this indicates that the message has been read.

You can use shell escape characters such as '*' and '?' in the <call> fields.

You can combine some of the various directory commands together eg:-

```
DIR TO G1TLH 5
or
DIR SUBJECT IOTA 200-250
```

You can abbreviate all the commands to one letter and use ak1a syntax:-

```
DIR/T G1* 10
DIR/S QSL 10-100 5
```

12.13 dx

dx [by <call>] <freq> <call> <remarks> Send a DX spot

This is how you send a DX Spot to other users. You can, in fact, now enter the <freq> and the <call> either way round.

```
DX FR0G 144.600
DX 144.600 FR0G
DX 144600 FR0G
```

will all give the same result. You can add some remarks to the end of the command and they will be added to the spot.

```
DX FR0G 144600 this is a test
```

You can credit someone else by saying:-

```
DX by G1TLH FR0G 144.600 he isn't on the cluster
```

The <freq> is compared against the available bands set up in the cluster. See SHOW/BANDS for more information.

12.14 help

help <cmd> Get help on a command

All commands can be abbreviated, so SHOW/DX can be abbreviated to SH/DX, ANNOUNCE can be shortened to AN and so on.

Look at the APROPOS <string> command which will search the help database for the <string> you specify and give you a list of likely commands to look at with HELP.

12.15 kill

kill <msgno> [<msgno> ..] Delete a message from the local system

Delete a message from the local system. You will only be able to delete messages that you have originated or been sent (unless you are the sysop).

12.16 links

links Show which nodes are physically connected

This is a quick listing that shows which links are connected and some information about them. See WHO for a list of all connections.

12.17 read

read Read the next unread personal message addressed to you

read <msgno> Read the specified message

You can read any messages that are sent as 'non-personal' and also any message either sent by or sent to your callsign.

12.18 reject/announce

reject/announce [0-9] <pattern> Set a reject filter for announce

Create an 'reject this announce' line for a filter.

An reject filter line means that if the announce matches this filter it is passed onto the user. See HELP FILTERS for more info. Please read this to understand how filters work - it will save a lot of grief later on.

You can use any of the following things in this line:-

info <string>	eg: iota or qsl
by <prefixes>	eg: G,M,2
origin <prefixes>	
origin_dxcc <numbers>	eg: 61,62 (from eg: sh/pre G)
origin_itu <numbers>	
origin_zone <numbers>	
by_dxcc <numbers>	
by_itu <numbers>	
by_zone <numbers>	
channel <prefixes>	
wx 1	filter WX announces
dest <prefixes>	eg: 6MUK,WDX (distros)

some examples:-

```
rej/ann by_zone 14,15,16 and not by G,M,2
```

You can use the tag 'all' to reject everything eg:

```
rej/ann all
```

but this probably for advanced users...

12.19 reject/spots

reject/spots [0-9] <pattern> Set a reject filter line for spots

Create a 'reject this spot' line for a filter.

A reject filter line means that if the spot matches this filter it is dumped (not passed on). See HELP FILTERS for more info. Please read this to understand how filters work - it will save a lot of grief later on.

You can use any of the following things in this line:-

freq <range>	eg: 0/30000 or hf or hf/cw or 6m,4m,2m
on <range>	same as 'freq'
call <prefixes>	eg: G,PA,HB9
info <string>	eg: iota or qsl
by <prefixes>	
call_dxcc <numbers>	eg: 61,62 (from eg: sh/pre G)
call_itu <numbers>	
call_zone <numbers>	
by_dxcc <numbers>	
by_itu <numbers>	
by_zone <numbers>	
origin <prefixes>	
channel <prefixes>	

For frequencies, you can use any of the band names defined in SHOW/BANDS and you can use a subband name like: cw, rtty, data, ssb - thus: hf/ssb. You can also just have a simple range like: 0/30000 - this is more efficient than saying simply: on HF (but don't get too hung up about that)

some examples:-

```
rej/spot 1 on hf
rej/spot 2 on vhf and not (by_zone 14,15,16 or call_zone 14,15,16)
```

You can use the tag 'all' to reject everything eg:

```
rej/spot 3 all
```

but this probably for advanced users...

12.20 reject/wcy

reject/wcy [0-9] <pattern> Set a reject WCY filter

It is unlikely that you will want to do this, but if you do then you can filter on the following fields:-

by <prefixes>	eg: G,M,2
origin <prefixes>	
origin_dxcc <numbers>	eg: 61,62 (from eg: sh/pre G)
origin_itu <numbers>	
origin_zone <numbers>	

```
by_dxcc <numbers>
by_itu <numbers>
by_zone <numbers>
channel <prefixes>
```

There are no examples because WCY Broadcasts only come from one place and you either want them or not (see UNSET/WCY if you don't want them).

This command is really provided for future use.

See HELP FILTER for information.

12.21 reject/wwv

reject/wwv [0-9] <pattern> Set a reject WWV filter

It is unlikely that you will want to do this, but if you do then you can filter on the following fields:-

```
by <prefixes>           eg: G,M,2
origin <prefixes>
origin_dxcc <numbers>    eg: 61,62 (from eg: sh/pre G)
origin_itu <numbers>
origin_zone <numbers>
by_dxcc <numbers>
by_itu <numbers>
by_zone <numbers>
channel <prefixes>
```

for example

```
reject/wwv by_zone 14,15,16
```

is probably the only useful thing to do (which will only show WWV broadcasts by stations in the US).

See HELP FILTER for information.

12.22 reply

reply Reply (privately) to the last message that you have read

reply <msgno> Reply (privately) to the specified message

reply B <msgno> Reply as a Bulletin to the specified message

reply NOPRivate <msgno> Reply as a Bulletin to the specified message

reply RR <msgno> Reply to the specified message with read receipt

You can reply to a message and the subject will automatically have "Re:" inserted in front of it, if it isn't already present.

You can also use all the extra qualifiers such as RR, PRIVATE, NOPRIVATE, B that you can use with the SEND command (see SEND for further details)

12.23 send

send <call> [<call> ...] Send a message to one or more callsigns

send RR <call> Send a message and ask for a read receipt

send COPY <msgno> <call> Send a copy of a message to someone

send PRIVATE <call> Send a personal message

send NOPRIVATE <call> Send a message to all stations

All the SEND commands will create a message which will be sent either to an individual callsign or to one of the 'bulletin' addresses.

SEND <call> on its own acts as though you had typed SEND PRIVATE, that is it will mark the message as personal and send it to the cluster node that that callsign is connected to.

You can have more than one callsign in all of the SEND commands.

You can have multiple qualifiers so that you can have for example:-

```
SEND RR COPY 123 PRIVATE G1TLH G0RDI
```

which should send a copy of message 123 to G1TLH and G0RDI and you will receive a read receipt when they have read the message.

SB is an alias for SEND NOPRIVATE (or send a bulletin in BBS speak) SP is an alias for SEND PRIVATE

12.24 set/address

set/address <your_address> Record your postal address

Literally, record your address details on the cluster.

12.25 set/announce

set/announce Allow announce messages

Allow announce messages to arrive at your terminal.

12.26 set/beep

set/beep Add beeps to terminal messages

Add a beep to DX and other terminal messages.

12.27 set/dx

set/dx Allow DX messages to arrive at your terminal

You can stop DX messages with the *unset/dx* command

12.28 set/dxgrid

set/dxgrid Allow grid squares on the end of DX messages

Some logging programs do not like the additional information at the end of a DX spot. If this is the case, use the *unset/dxgrid* command to remove the grid squares.

12.29 set/echo

set/echo Make the cluster echo your input

If you are connected via a telnet session, different implementations of telnet handle echo differently depending on whether you are connected via port 23 or some other port. You can use this command to change the setting appropriately.

You can remove the echo with the *unset/echo* command

The setting is stored in your user profile.

YOU DO NOT NEED TO USE THIS COMMAND IF YOU ARE CONNECTED VIA AX25.

12.30 set/here

set/here Set the here flag

Let others on the cluster know you are here by only displaying your callsign. If you are away from your terminal you can use the *unset/here* command to let people know you are away. This simply puts brackets around your callsign to indicate you are not available.

12.31 set/homenode

set/homenode <node_call> Set your home cluster

Tell the cluster system where you normally connect to. Any Messages sent to you will normally find their way there should you not be connected. eg:-

```
SET/HOMENODE gb7djK
```

12.32 set/language

set/language <language> Set the language you wish to use

You can select the language that you want the cluster to use. Currently the languages available are *en* (English) *sp* (Spanish) and *nl* (Dutch).

12.33 set/location

set/location <lat and long> Set your latitude and longitude

You can set your latitude and longitude manually or alternatively use the *set/qra* command which will do the conversion for you.

```
set/location 54 04 N 2 02 E
```

12.34 set/logininfo

set/logininfo Show logins and logouts of nodes and users

Show users and nodes when they log in and out of the local cluster. You can stop these messages by using the *unset/logininfo* command.

12.35 set/name

set/name <your_name> Set your name

Tell the cluster what your name is, eg:-

```
set/name Dirk
```

12.36 set/page

set/page <n> Set the number of lines per page

Tell the system how many lines you wish on a page when the number of lines of output from a command is more than this. The default is 20. Setting it explicitly to 0 will disable paging.

```
SET/PAGE 30
SET/PAGE 0
```

The setting is stored in your user profile.

12.37 set/qra

set/qra <locator> Set your QRA locator

Tell the system what your QRA (or Maidenhead) locator is. If you have not done a SET/LOCATION then your latitude and longitude will be set roughly correctly (assuming your locator is correct ;-). For example:-

```
SET/QRA J002LQ
```

12.38 set/qth

set/qth <your QTH> Set your QTH

Tell the system where you are. For example:-

```
set/qth East Dereham, Norfolk
```

12.39 set/talk

set/talk Allow talk messages to be seen at your console

Allow talk messages to arrive at your console. You can switch off talks with the *unset/talk* command.

12.40 set/wcy

set/wcy Allow WCY messages to be seen at your console

Allow WCY information to be seen at your console. You can switch off WCY messages with the *unset/wcy* command.

12.41 set/wwv

set/wwv Allow WWV messages to be seen at your console

Allow WWV information to be seen at your console. You can switch off WWV messages with the *unset/wwv* command.

12.42 set/wx

set/wx Allow WX messages to be seen at your console

Allow WX information to be seen at your console. You can switch off WX messages with the *unset/wx* command.

12.43 show/configuration

show/configuration [<node>] Show all visible nodes and their users

This command allows you to see all the users that can be seen and the nodes to which they are connected. With the optional *node*, you can specify a particular node to look at.

This command is normally abbreviated to: sh/c

BE WARNED: the list that is returned can be VERY long

12.44 show/configuration/node

show/configuration/node Show all the nodes connected

Show all the nodes connected locally and the nodes they have connected.

12.45 show/date

show/date [<prefix>|<callsign>] Show the local time

This is very nearly the same as SHOW/TIME, the only difference the format of the date string if no arguments are given.

If no prefixes or callsigns are given then this command returns the local time and UTC as the computer has it right now. If you give some prefixes then it will show UTC and UTC + the local offset (not including DST) at the prefixes or callsigns that you specify.

12.46 show/dx

show/dx [options] interrogate the spot database

If you just type SHOW/DX you will get the last so many spots (sysop configurable, but usually 10).

In addition you can add any number of these options in very nearly any order to the basic SHOW/DX command, they are:-

on <band>	- eg 160m 20m 2m 23cm 6mm	
on <region>	- eg hf vhf uhf shf	(see SHOW/BANDS)

- `<number>` - the number of spots you want
- `<from>-<to>` - `<from>` spot no `<to>` spot no in the selected list

- `<prefix>` - for a spotted callsign beginning with `<prefix>`
- `*<suffix>` - for a spotted callsign ending in `<suffix>`
- `*<string>*` - for a spotted callsign containing `<string>`

- `day <number>` - starting `<number>` days ago
- `day <from>-<to>` - `<from>` days `<to>` days ago

- `info <text>` - any spots containing `<text>` in the info or remarks

- `by <call>` - any spots spotted by `<call>` (spotter `<call>` is the same).

- `qsl` - this automatically looks for any qsl info on the call held in the spot database.

- `iota [<iota>]` - If the iota island number is missing it will look for the string `iota` and anything which looks like an iota island number. If you specify then it will look for that island.

- `gra [<locator>]` - this will look for the specific locator if you specify one or else anything that looks like a locator.

e.g.

```
SH/DX 9m0
SH/DX on 20m info iota
SH/DX 9a on vhf day 30
SH/DX rf1p qsl
SH/DX iota
SH/DX iota eu-064
SH/DX gra jn86
```

12.47 show/dxcc

show/dxcc <prefix> Interrogate the spot database by country

This command takes the `<prefix>` (which can be a full or partial callsign if desired), looks up which internal country number it is and then displays all the spots as per SH/DX for that country.

The options for SHOW/DX also apply to this command. e.g.

```
SH/DXCC G
SH/DXCC W on 20m info iota
```


12.48 show/files

show/files [**<filearea>** [**<string>**]] List the contents of a filearea

SHOW/FILES on its own will show you a list of the various fileareas available on the system. To see the contents of a particular file area type:-

```
SH/FILES <filearea>
```

where <filearea> is the name of the filearea you want to see the contents of.

You can also use shell globbing characters like '*' and '?' in a string to see a selection of files in a filearea eg:-

```
SH/FILES bulletins arld*
```

See also TYPE - to see the contents of a file.

12.49 show/filter

show/filter Show the filters you have set

Show the contents of all the filters that are set by you. This command displays all the filters set - for all the various categories.

12.50 show/moon

show/moon [**<prefix>** | **<callsign>**] Show moon rise and set times

Show the Moon rise and set times for a (list of) prefixes or callsigns, together with the azimuth and elevation of the sun currently at those locations.

If you don't specify any prefixes or callsigns, it will show the times for your QTH (assuming you have set it with either SET/LOCATION or SET/QRA), together with the current azimuth and elevation.

In addition, it will show the gain or loss dB relative to the nominal distance of 385,000Km due to the ellipsoidal nature of the orbit.

If all else fails it will show the Moonrise and set times for the node that you are connected to.

For example:-

```
SH/MOON
SH/MOON G1TLH W5UN
```

12.51 show/muf

show/muf **<prefix>** [**<hours>**][**long**] Show the likely propagation to <prefix>

This command allow you to estimate the likelihood of you contacting a station with the prefix you have specified. The output assumes a modest power of 20dBW and receiver sensitivity of -123dBm (about 0.15µV/10dB SINAD)

The result predicts the most likely operating frequencies and signal levels for high frequency (shortwave) radio propagation paths on specified days of the year and hours of the day. It is most useful for paths between 250 km and 6000 km, but can be used with reduced accuracy for paths shorter or longer than this.

The command uses a routine MINIMUF 3.5 developed by the U.S. Navy and used to predict the MUF given the predicted flux, day of the year, hour of the day and geographic coordinates of the transmitter and receiver. This routine is reasonably accurate for the purposes here, with a claimed RMS error of 3.8 MHz, but much smaller and less complex than the programs used by major shortwave broadcasting organizations, such as the Voice of America.

The command will display some header information detailing its assumptions, together with the locations, latitude and longitudes and bearings. It will then show UTC (UT), local time at the other end (LT), calculate the MUFs, Sun zenith angle at the midpoint of the path (Zen) and the likely signal strengths. Then for each frequency for which the system thinks there is a likelihood of a circuit it prints a value.

The value is currently a likely S meter reading based on the conventional 6dB / S point scale. If the value has a '+' appended it means that it is 1/2 an S point stronger. If the value is preceded by an 'm' it means that there is likely to be much fading and by an 's' that the signal is likely to be noisy.

By default SHOW/MUF will show the next two hours worth of data. You can specify anything up to 24 hours worth of data by appending the no of hours required after the prefix. For example:-

```
SH/MUF W
```

produces:

```
RxSens: -123 dBm SFI: 159   R: 193   Month: 10   Day: 21
Power :   20 dBW   Distance: 6283 km   Delay: 22.4 ms
Location                               Lat / Long   Azim
East Dereham, Norfolk                 52 41 N 0 57 E   47
United-States-W                       43 0 N 87 54 W   299
UT LT  MUF Zen  1.8  3.5  7.0 10.1 14.0 18.1 21.0 24.9 28.0 50.0
18 23 11.5 -35  mS0+ mS2   S3
19  0 11.2 -41  mS0+ mS2   S3
```

indicating that you will have weak, fading circuits on top band and 80m but usable signals on 40m (about S3).

inputting:-

```
SH/MUF W 24
```

will get you the above display, but with the next 24 hours worth of propagation data.

```
SH/MUF W L 24
SH/MUF W 24 Long
```

Gives you an estimate of the long path propagation characteristics. It should be noted that the figures will probably not be very useful, nor terrible accurate, but it is included for completeness.

12.52 show/prefix

show/prefix <callsign> Interrogate the prefix database

This command takes the <callsign> (which can be a full or partial callsign or a prefix), looks up which internal country number it is and then displays all the relevant prefixes for that country together with the internal country no, the CQ and ITU regions.

See also SHOW/DXCC

12.53 show/qra

show/qra <locator> [<locator>] Show the distance between locators

show/qra <lat> <long> Convert latitude and longitude to a locator

This is a multipurpose command that allows you either to calculate the distance and bearing between two locators or (if only one locator is given on the command line) the distance and bearing from your station to the locator. For example:-

```
SH/QRA I092QL
SH/QRA JN06 IN73
```

The first example will show the distance and bearing to the locator from yourself, the second example will calculate the distance and bearing from the first locator to the second. You can use 4 or 6 character locators.

It is also possible to convert a latitude and longitude to a locator by using this command with a latitude and longitude as an argument, for example:-

```
SH/QRA 52 41 N 0 58 E
```

12.54 show/qrz

show/qrz <callsign> Show any callbook details on a callsign

This command queries the QRZ callbook server on the internet and returns any information available for that callsign. This service is provided for users of this software by <http://www.qrz.com>

12.55 show/route

show/route <callsign> Show the route to <callsign>

This command allows you to see to which node the callsigns specified are connected. It is a sort of inverse sh/config.

```
sh/route n2tly
```

12.56 show/satellite

show/satellite <name> [<hours> <interval>] Show satellite tracking data

Show the tracking data from your location to the satellite of your choice from now on for the next few hours.

If you use this command without a satellite name it will display a list of all the satellites known currently to the system.

If you give a name then you can obtain tracking data of all the passes that start and finish 5 degrees below the horizon. As default it will give information for the next three hours for every five minute period.

You can alter the number of hours and the step size, within certain limits.

Each pass in a period is separated with a row of '-----' characters

So for example:-

```
SH/SAT AO-10
SH/SAT FENGYUN1 12 2
```

12.57 show/sun

show/sun [<prefix>|<callsign>] Show sun rise and set times

Show the sun rise and set times for a (list of) prefixes or callsigns, together with the azimuth and elevation of the sun currently at those locations.

If you don't specify any prefixes or callsigns, it will show the times for your QTH (assuming you have set it with either SET/LOCATION or SET/QRA), together with the current azimuth and elevation.

If all else fails it will show the sunrise and set times for the node that you are connected to.

For example:-

```
SH/SUN
SH/SUN G1TLH K9CW ZS
```

12.58 show/time

show/time [<prefix>|<callsign>] Show the local time

If no prefixes or callsigns are given then this command returns the local time and UTC as the computer has it right now.

If you give some prefixes then it will show UTC and UTC + the local offset (not including DST) at the prefixes or callsigns that you specify.

12.59 show/wcy

show/wcy Show the last 10 WCY broadcasts

show/wcy <n> Show the last <n> WCY broadcasts

Display the most recent WCY information that has been received by the system

12.60 show/wwv

show/wwv Show the last 10 WWV broadcasts

show/wwv <n> Show the last <n> WWV broadcasts

Display the most recent WWV information that has been received by the system

12.61 sysop

sysop Regain your privileges if you login remotely

The system automatically reduces your privilege level to that of a normal user if you login in remotely. This command allows you to regain your normal privilege level. It uses the normal system: five numbers are returned that are indexes into the character array that is your assigned password (see SET/PASSWORD). The indexes start from zero.

You are expected to return a string which contains the characters required in the correct order. You may intersperse those characters with others to obscure your reply for any watchers. For example (and these values are for explanation :-):

```
password = 012345678901234567890123456789
> sysop
```

22 10 15 17 3

you type:-

```
aa2bbbb0ccc5ddd7xxx3n
or 2 0 5 7 3
or 20573
```

They will all match. If there is no password you will still be offered numbers but nothing will happen when you input a string. Any match is case sensitive.

12.62 talk

```
talk <callsign> Enter talk mode with <callsign>
talk <callsign> <text> Send a text message to <callsign>
talk <callsign> > <node_call> [<text>] Send a text message to <callsign> via
<node_call>
```

Send a short message to any other station that is visible on the cluster system. You can send it to anyone you can see with a SHOW/CONFIGURATION command, they don't have to be connected locally.

The second form of TALK is used when other cluster nodes are connected with restricted information. This usually means that they don't send the user information usually associated with logging on and off the cluster.

If you know that G3JNB is likely to be present on GB7TLH, but you can only see GB7TLH in the SH/C list but with no users, then you would use the second form of the talk message.

If you want to have a ragchew with someone you can leave the text message out and the system will go into 'Talk' mode. What this means is that a short message is sent to the recipient telling them that you are in a 'Talking' frame of mind and then you just type - everything you send will go to the station that you asked for.

All the usual announcements, spots and so on will still come out on your terminal.

If you want to do something (such as send a spot) you precede the normal command with a '/' character, eg:-

```
/DX 14001 G1TLH What's a B class licensee doing on 20m CW?
/HELP talk
```

To leave talk mode type:

```
/EX
```

12.63 type

```
type <filearea>/<name> Look at a file in one of the fileareas
```

Type out the contents of a file in a filearea. So, for example, in filearea 'bulletins' you want to look at file 'arld051' you would enter:-

```
TYPE bulletins/arld051
```

See also SHOW/FILES to see what fileareas are available and a list of content.

12.64 who

who Show who is physically connected locally

This is a quick listing that shows which callsigns are connected and what sort of connection they have

12.65 wx

wx <text> Send a weather message to local users

wx full <text> Send a weather message to all cluster users

Weather messages can sometimes be useful if you are experiencing an extreme that may indicate enhanced conditions

Next [Previous](#) [Contents](#)