UNIVERSITÀ
DEGLI STUDI DI TRIESTE

dipartimento di matematica e geoscienze

The DMG Manuals

The Local DSTX Network

A brief description of computers, storage, shell configuration etc…

Document Name: DMGBrief.pdf
Document Version: 1.2
Date Issued: 25 September 2017 14:36
Author: Franco Vaccari

University of Trieste
Department of Mathematic and Geosciences
Group of Seismology
via Weiss, 4
34128 Trieste, Italy

Table of Contents

The DSTX local network

A local network of computers exists, dedicated to the people working with the Group of Seismology at the Department of Mathematic and Geosciences of the University of Trieste. The DST acronym comes from the time when the department was named “Dipartimento di Scienze della Terra” (Dept. of Earth Sciences)

Web site

A web site has been set up for the users of the DSTX network, where news, manuals (including this one) and other relevant information can be found. Its address is

http://dstx02.units.it/dstxpub

Accounts

Network accounts are provided upon request. Users can login on any client computer connected to the main server, and work in the exact same environment (home directory, shell configuration, etc).

Home directory

Each user has a home directory (~), where to keep his own Word Processing documents and the like. ⚠️➡️ Users are NOT supposed to run computational jobs inside that directory. There are another volumes dedicated to that, namely */XDST, /bkXDST, /tmpXDST* ⬅️⚠️.

/bkXDST and /tmpXDST directories

You have two dedicated storage areas on a network RAID disk, where you are expected to run your computational jobs and to store the input and output data .

The path to those directories are /bkXDST/$USER and */tmpXDST/$USER*, where *$USER* is the user’s login name that has been assigned to you (e.g. */tmpXDST/joe*).

There is an alias to reach that area, so you can just type

```
cdk     to reach /bkXDST/$USER
cdt     to reach /tmpXDST/$USER
```

to get there. You can start adding your own directories for the computational jobs you will be going to run.

⚠️➡️**Those directories have to be considered as a temporary storage area, where preliminary computations are performed. Only the files associated with the final results should be moved to your directory on the /XDST disk, described below, and all preliminary and obsolete files should be deleted, to avoid filling the /bkXDST and the /tmpXDST volumes with garbage**⬅️⚠️

/XDST directory

⚠️➡️ You have another dedicated storage area on a different network RAID disk, where you are expected to store **only** the final results of your computational jobs and the corresponding input and output data ⬅️⚠️.

The path to this directory is */XDST/$USER*, where *$USER* is the user's login name that has been assigned to you (e.g. */XDST/joe*).

There is an alias to reach that area, so you can just type

```
cdd
```

to get there. By default, two subdirectories have been automatically set up for you in /XDST/$USER:

- *src* (where you should put the source code you'll eventually develop)

- *bin* (where your executable files should be placed).

You can start adding your own directories for the computational jobs you will be going to run.

Connecting to the system

Connection from the outside world

The DSTX network can be reached only passing through the dedicated server dstx01.units.it (IP address 140.105.54.5). If you are connecting from a Unix/Linux computer, just ssh from a terminal session using this command:

```
ssh -p3897 user@dstx01.units.it
```

where *user* is the short name associated with your account. If you are connecting using a Windows computer, you need to install a ssh client software, like for instance the free Putty application, and configure it to open an ssh session to host dstx01.units.it using port 3897.

Connection sitting in front of a local client computer

Mac OS X Client computers are available in the computer room. Enter your username and password at the login screen. Once logged in you may navigate the file system using the Finder. Not familiar with the Mac OS? Should not be difficult. See Appendix 1 for some basic notions. Please try exploring the Help menu before asking too obvious questions…

Running computational jobs

⚠️➡ Computational jobs must be run on the dedicated server *is01* ⬅⚠️. *If you are sitting in front of a client computer in the computer room, launch the Terminal application by clicking on its icon in the Dock, at the bottom of the screen:*

and then type:

```
ssh is01
```

You are now connected to is01, and should select the volume on which to perform the computations. There are two volumes available that you can choose from, and two aliases will select your personal directory on each of them:

```
cdk ——>     /bkXDST/$USER cdt  ——>  /tmpXDST/$USER
cdt     ——>  /tmpXDST/$USER
```

You are strongly advised to create new subdirectories for each run you are going to execute, as the amount of files that will be generated might be huge and you can easily loose track of what is what.

Running jobs interactively

Single programs can be executed in the Terminal shell by simply typing their name, followed by the parameters eventually required. Two examples:

```
ray
```

will run the ray program to generate the Rayleigh modes for the structure given in input.

```
hazgmt *.amx
```

will run the hazgmt program passing to it all the files with extension *.amx* present in the working directory.

Running jobs in the background

In the example above, if you close your Terminal session while the *ray* program is running, the *ray* process will be automatically killed, and you'll have to run it again from the beginning. To avoid this problem, ⚠➡ programs can be run "in the background", so that they will continue running even if the Terminal session from where they have been launched is closed. To do so, simply follow the program name with the "&" character ⬅⚠:

```
ray &
```

After doing so, you can safely close the Terminal session, or quit the Terminal application, or even logout from the system. The *ray* process will continue running till the end of its job.

For the submission of more complicated shell scripts, like for instance the *jobfd* shell script for the computation of synthetic seismograms along 2D profiles with the hybrid technique, it is safer to submit the job via the at command:

```
echo jobfd | at now
```

No need for the & character in this case.

Checking the status of jobs

Some programs can take long time to finish. Hours, if not days. Therefore they have to be run in the background, so that you can logout from the system and let the client computer be available to other users.

You can then check the status of the running jobs looking at the active processes on the machine they have been launched on, hopefully *is01*. ⚠➡ Some aliases has been set up for this purpose ⬅⚠. You can connect via ssh to the machine you are interested in, and you can type any of the following commands:

j	(**j**obs)	list the top 15 CPU-consuming jobs on the machine, belonging to any user
mj	(**m**y **j**obs)	list the top 10 CPU-consuming jobs on the machine, belonging to me
amj	(**a**ll **m**y **j**obs)	list all my jobs on the machine, sorted by %CPU usage

to check if the process you're interested in is still running.

Input data and execution examples

Examples of executions of the most relevant packages developed by the group, with input data and results, can be found in the */XDST/Examples* directory. Look into the manual relative to the computational job you are going to run for the list of files you need.

Shell environment

The default shell

⚠ Default shell for user accounts is tcsh ⚠, properly configured to have access to the XDST seismological software and to MacPorts-installed open source applications. Other shells can be chosen (csh, bash, zsh) but are not properly pre-configured, so their use is discouraged, unless you volunteer to prepare a proper configuration to share with the administrator.

Path for executables

⚠ Executables compiled by the user on *is01* should be put in */XDST/$USER/bin/10.6* in order to appear in the user's PATH, and will have precedence over the official DSTX programs eventually located in */XDST/bin/10.6* ⚠.

The use of PPC executables is discouraged, as all computational jobs should be run on *is01*, with Intel architecture.

In case a program is present in more than one place specified in your PATH, typing

```
which <program_name>
```

will tell you the path of the one that will be executed, and typing

```
where <program_name>
```

will show you all occurrences available in your PATH.

When you add a new executable to */XDST/$USER/bin/10.6*, type

```
rehash
```

to make it discoverable in your PATH.

Compiling programs

⚠ Your source code must be compiled on *is01* server ⚠.

The Apple-installed GCC compiler (the GNU Compiler Collection, installed in */usr/bin*) does not include the FORTRAN language. Therefore another, more complete and usually more recent version of GCC is installed in */opt/local* using the MacPorts package manager.

Apple-installed compilers

For C programs, gcc-4.0 and gcc-4.2 are available, both located in */usr/bin*. *gcc* is a symbolic link pointing to */usr/bin/gcc-4.2*.

For C++ programs, g++-4.0 and g++-4.2 are available, both located in */usr/bin*. *g++* is a symbolic link pointing to */usr/bin/g++-4.2*.

MacPorts installed compilers

The GCC compiler collection currently installed is at version 4.9.2, and is located in */opt/local*.

For C programs, use *gcc-mp-4.9* compiler.

For C++ programs, use *g++-mp-4.9* compiler.

For FORTRAN program, *gfortran* is a symbolic link pointing to */opt/local/bin/gfortran-mp-4.9.*

The g95 compiler is another FORTRAN compiler that is also installed and can be tried when a gfortran bug prevents your code from compiling. Executables produced by g95 are usually faster than the equivalent produced by gfortran by a 20% factor. Official DSTX programs located in /XDST/bin/10.6 were produced mostly with g95 compiler.

Executables

⚠➡ Executables you produce for *is01* processor are better saved into */XDST/$USER/bin/10.6* directory ⬅⚠, that is part of your $PATH environmental variable. In this way, you will be able to call them from any working directory you will be in.

Pre-installed Unix executables

Seismological software developed at the department is located in */XDST/bin/10.6* and */XDST/Scripts/*.

MacPorts-installed open source executables are installed in */opt/local/bin* and are also included in the PATH.The SAC seismic analysis code is installed in */usr/local*.

Among the software installed with the Fink package manager, we mention:

-GMT Generic Mapping Tool, useful for mapping and scientific plots

-gnuplot Scientific plotting

-Ghostscript PostScript rendering and conversion

-ImageMagick Graphic manipulation software

but much more is installed. To see the complete list of the MacPorts installed packages, give the command

```
port installed
```

⚠➡ All the above software is already included in the PATH environmental variable, so is ready to be called on *is01* from any working directory ⬅⚠.

Default aliases

Aliases are useful shortcuts associated with more complex or lengthy commands. Some aliases are already set by default for your account, and are listed in Table 1. You can override the predefined aliases, and add your own if you wish, by editing your ~/.tcshrc file.

Table 1. Default aliases defined for tcsh.

Alias	Command	Purpose
`cdk`	`cd /bkXDST/$USER`	go to the user's working directory on /bkXDST

cdt	*cd /tmpXDST/$USER*	go to the user's working directory on /tmpXDST
cdd	*cd /XDST/$USER*	go to the user's working directory on /XDST
cde	cd /XDST/Examples	go to the directory where example files and example executions of the main seismological software produced by the group are stored
cdg	*cd /XDST/$USER/Git*	go to the directory where git repositories should be stored
l	*ls -l*	detailed list
ll	*ls -l !* | more*	detailed list, paged
lt	*ls -lt !* | head -10*	show the 10 most recent files
l1	*ls -1*	list files in a single column
la	*ls -la*	list all files, hidden files included
lbig	*ls -lSh !* | more*	list all files, sorted by size
gf2	*gfortran -fno-automatic -fno-second-underscore -O2 !$ -o $DST_USER_BIN\/!$:s/.f/.out/*	expects a file <filename>.f, produces the corresponding executable <filename>.out, and place it in /XDST/$USER/bin/10.6
aj	*ps -ewwwAo %cpu,user,start,pid,command | sed s/\ %CPU/~%CPU/ | sort -r | sed s/~%CPU/\ %CPU/*	list all the processes
amj	*ps -eAo %cpu,start,user,pid,command | grep $user | sort -r*	list all the processing belonging to me, sorted by CPU% used (descending)
mj	*ps -eAo %cpu,start,user,pid,command | grep $user | sort -r | head -11*	list the 10 processes belonging to me consuming more CPU resources, sorted by CPU% used (descending)
uj	*set tmp=$1;ps -eAo%cpu,user,start,pid,command | grep $tmp*	list the jobs matching the passed argument (will give a grep error if called without an argument)
compDstUser	*source $DST_INIT/Scripts/compDstUser*	set the environment so that executables produced by Makefiles of Git repositories are placed in /XDST/$USER/bin/10.6

Appendix 1 - Mac OS X applications

If you are not used to Mac OS X, here you have a list of the applications you'll find yourself working with. For your convenience, their icons are placed in the dock, visible at the bottom of the screen. You should get familiar with them if you plan working in front of the client computers. You don't need to know about them if you just connect remotely to DSTX via ssh.

Finder

It's the main interface between you and the file system. Use it to navigate through folders, open files etc.

Terminal

It's the application you will use to run the computational jobs, taking advantage of the Unix underpinnings of Mac OS X. You better get familiar with the Unix system: spending some time learning the basics will vastly improve your working experience with the computational software installed on the DSTX computers.

TextWrangler

It's the application you will use to edit text files, typically to prepare input data for your computational jobs (if you really don't want to use vi or the like in Terminal…)

Preview

This application can display most of the graphic files you might need to visualize (PostScript, PDF, jpeg, png etc).

Safari

The Mac OS X default internet browser.

Appendix 2 - DSTX Hardware

Servers

A rack with 7 Xserve computers constitutes the core of the DSTX local network. The characteristics are listed in Table A1. An Xserve RAID is connected to is01, with 6+1 spare disks in RAID 5 configuration for a total of 1.8 TB of space dedicated to */XDST* mount point. An external RAID disk provides the */tmpXDST* (1.8 TB) and */bkXDST* (3.6 TB) mount points. A Gigabit switch is providing the network infrastructure.

Table A1. Rack-mounted server computers

Name	CPU	OS	Cores	GHz (CPU/ Bus)	RAM (GB)	Services
dstx01/ ps01	PPC G5	Mac OS X Server 10.5.8	2	2.0/1.0	7	External access, VPN
dstx02/ is02	Intel Xeon	Mac OS X Server 10.7.5	4	2.0/1.33	32	External access, Web services
dstx03/ sismats	Intel	Mac OS X 10.9.4	8	2.8/1.6	32	External access, Computations
is01	Intel Xeon	Mac OS X Server 10.6.8	8	2.26/1.33	96	OD master, Computations
pn01(*)	PPC G5		2	2.0/1.0	3	Computational jobs
pn02(*)	PPC G5		2	2.3/1.15	5	Computational jobs
pn03(*)	PPC G5		2	2.3/1.15	5	Computational jobs
pn04(*)	PPC G5		2	2.3/1.15		Computational jobs

(*) currently switched off

Clients

Several Mac clients are available, configured with network accounts, to grant access to the DSTX network for the people working at the Department. They are listed in Table A2.

Table A2. Client computers

Name	CPU	OS	Cores	GHz (CPU/ Bus)	RAM (GB)	Location
im01	Intel Core 2 Duo	Mac OS X 10.6.8	2	2.16/0.8	3	Server room
im02	Intel Core 2 Duo	Mac OS X 10.6.8	2	2.4/0.8	2	Computer room
im03	Intel Core 2 Duo	Mac OS X 10.6.8	2	2.0/0.8	4	Computer room
im04	Intel Core 2 Duo	Mac OS X 10.6.8	2	2.0/1.0	2	Computer room
im05	Intel Core 2 Duo	Mac OS X 10.6.8	2	2.16/0.67	2	Computer room
im06	Intel Core i7	Mac OS X 10.7.5	4	2.16/0.67	16	Computer room
it01	Intel Core Duo	Mac OS X 10.6.8	2	2.0/0.67	2	Computer room
it02	Intel Core Duo	Mac OS X 10.6.8	2	2.0/0.67	2	Computer room
pm01	PPC G3	Mac OS X 10.4.11	1	0.4/0.1	1	Room S5
pm04	PPC G4	Mac OS X 10.5.8	1	1.0/0.13	1	Computer room
pm05	PPC G4	Mac OS X 10.4.11	2	1.0/?	1	Room T26
pm07	PPC G4	Mac OS X 10.5.8	1	1.0/0.13	1	Computer room