

Spring term 2000

Using Linux at the ITU data bars

Department of Information Technology, Technical University of Denmark
Michael R. Hansen, Jørgen Arendt Jensen, Morten Olsen and Hans Rischel
The IT University of Copenhagen
Niels Hallenberg, Anders Steen Rasmussen and Peter Sestoft

These notes explain how to use Linux at the ITU data bars at ITU.

ITU has many data bars located in the buildings at Glentevej 65 and Glentevej 67. At Glentevej 65, we have a data bar in room 0.05. At Glentevej 67, we have data bars in the following rooms: 1.03, 1.30 and 2.47. All PC's have Windows NT and Linux (RedHat 6.1) installed. This note describes how to use Linux. **Linux** is *different* from the usual Windows system on PC's. Linux belongs to the UNIX family of operating systems which are widely used at universities.

Contents

1	Logging in	2
2	Executing commands via the button-bar	3
3	Windows and icons	4
4	Printing	5
5	Running Java programs	5
6	Running SML programs	5
6.1	SML system functions	7
7	File, directories and paths	8
8	Star Office	9
9	L ^A T _E X	9
10	Some useful UNIX commands	10
11	Desktop environments in the data bars	12
12	More about Linux	13

1 Logging in

The access to the computers in the ITU data bar system requires a user id (login) and a password. You will get your user id and password on a sheet of paper which is issued by the ITU administration. Your user id (login) and e-mail address, minus @itu.dk, are the same, that is, with login nh your e-mail address is nh@itu.dk.

You start a session by logging in at a PC and you terminate the session by logging out. Logging in comprises the following steps:

1. If the PC is running Windows, then restart the machine, that is, activate “Shut Down” in the “Start” menu. Type `linux` when the `Lilo:` prompt appears:
2. Wait for the log-in box on the screen. If the log-in box doesn't appear, then mail `sysadm@itu.dk` and report the problem.
3. Move the cursor to the log-in box and type your user id terminated by `return`. Then type your password terminated by `return`. Note that the system does echo your password using asterisks “*”.
4. The desktop appears on the screen and the log-in is completed.

<p>A session should always be terminated by logging out. You are NOT allowed to try to monopolize the PC by locking the screen. You do not need to turn off the PC.</p>
--

2 Executing commands via the button-bar

The desktop environment consists of a *button-bar* to the left and a *task-bar* at the right of the button-bar. The button-bar provides an easy way for executing some commonly used commands (programs). The picture at the right shows the button-bar. A button is activated by clicking on the button (moving the cursor to the button and pressing the left mouse key):

K opens a menu similar to the start button in Windows. From the menu you can start various applications. The applications are grouped in categories, for instance, to start the editor Emacs, you choose the category “Anvendelser”.

3×windows gives you a menu with the currently running applications and your six virtual desktops. A virtual desktop is a screen where you can have windows. The physical screen shows one virtual desktop at the time. You can for instance start Emacs in the desktop `Kilde` (“kildekode”) and Netscape in the desktop `Net` (“netværk”). You can then switch between the desktops `Kilde` and `Net`, and at any time you will either see Emacs or Netscape. Having more than one desktop is new to most Windows users and some people find them annoying, that is, only uses one desktop containing all windows – try it out and do what is best for you. The virtual desktop `Dok` is an abbreviation for “dokumentation” and `Test` for “afprøvning”.

Folder starts the application KFM Handbook, where you can manipulate folders and files – similar to the file manager in Windows.

Screen and circuit board opens the KDE Kontrolcenter, where you can set various system parameters.

Book! opens KDE hjælp. You can find information on how to use the KDE desktop environment and information on most GNU-related applications such as Emacs. To get information about GNU-related applications choose “System GNU info contents”.

Cross performs log out. Pressing the left mouse key creates the Exit dialog box where you click on **Log af**.

Padlock is used for locking the screen. **DO NOT USE IT – IT DOES NOT WORK AND WILL LOCK THE SCREEN FOREVER.**

Net, Dok, Kilde, Test, Div 1, Div 2 are used to shift the physical screen between the six virtual desktops.

Screen and shell creates a console window running a shell. The window can be used to type any UNIX command – similar to the DOS shell (window) in Windows.

Gnu^a starts the editor program Emacs in a new window. Emacs can be used for editing programs, and both Java and SML compilers can be invoked from inside Emacs.

N starts Netscape in a new window. Netscape is used to find and display information stored on the World Wide Web.

Book and ? opens a menu with various links to ITU documentation including printer status.



^aA Gnu is a large African antelope, see <http://www.geocities.com/RainForest/Canopy/9882/animalofthemoth.html>.



Figure 1: A window

3 Windows and icons

A window displays a separate dialogue with the operator. A window in a data bar has the general lay-out as shown in Figure 1, with a *system menu button*, a *sticky button*, a *title bar*, a *minimise button*, a *maximise button*, and a *close button* located at the top of the window.

A window is selected (and gets into the foreground) when the the left mouse key is click'ed in the window. A window can be moved around on the screen if the left mouse key is kept pressed down while the cursor is placed in the title bar (or `Ctrl+Alt+left mouse button` clicked). A window can be resized if the left mouse key is kept pressed down while the cursor is placed in one of the corners of the window.

Pressing the maximise button resizes the window to extend over all of the screen, while pressing it once more gets the window back to the previous size. Pressing the minimize button shrinks the window to a small icon which is placed in the task bar. The task bar is located at the top left corner, see Figure 2. A minimized window is restored by click'ing on the programs *icon* in the task bar.

Pressing the sticky button “sticks” the window to the desktop such that the window will stick on the desktop when you change virtual desktop (as selected by **Net**, **Dok**, **Kilde**, **Test**, **Div 1** and **Div 2**).

The close button is used to terminate the task in the window in a orderly fashion. A “dead” window can be killed by selecting **Luk** in the system menu (move cursor to the system menu button in the upper left corner, press down the left mouse key, move to **Luk** in the menu and release the key).

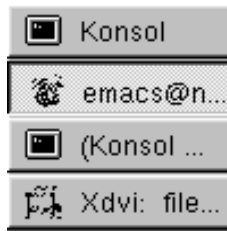


Figure 2: A task bar

4 Printing

Printers are located in various rooms; use the URL `http://localhost:631/printers` to see which printers are available and where they are located. You can also choose the item “Printere” in the menu found by pressing the **[Book and ?]** button on the button-bar, see Section 2. A printer is named `pXXX`, where `XXX` is the room-number where you find it (e.g., printer `p165` is located in room 1.65). We also have copy machines that can be used as printers. They are named `cXXX`. Currently the copy machines cannot be used for printing, but this feature will be enabled in the future.

5 Running Java programs

Editing and running Java programs can be done with Emacs. Given that your Java program resides in an Emacs buffer, typing `Ctrl+c Ctrl+c` will compile the program, and `Ctrl+c Ctrl+r` will execute the compiled program. A more detailed description on compiling and executing Java programs can be found in the document “Opsætning af PC” found at the URL: `http://www.itu.dk/intranet/sysadm/cdroms/sir/Documentation/opsaetWindows.pdf`.

6 Running SML programs

With Emacs, you can edit your SML program in one window and run an SML session in another window. Emacs provides simple commands for SML program layout, for loading the program into the SML session, and for repeating entries in the ML session.

Emacs is a powerful editor once you have learned to use it, but the commands may appear strange at first. The menus, however, make use of Emacs easier. An overview of Emacs commands is found in the *GNU Emacs Reference Card* (see `http://www.itu.dk/intranet/sysadm/cdroms/sir/Documentation/refcard.pdf`). The composed keys `Meta-x` (“meta-*x*”) and `Ctrl-x` (“control-*x*”) for arbitrary key *x* are made as follows:

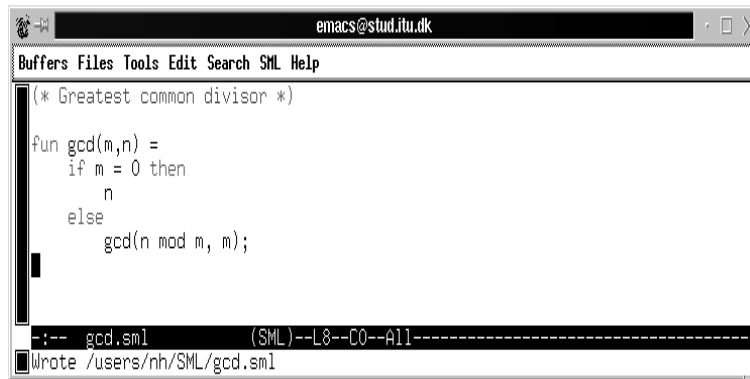


Figure 3: The Emacs editor with an SML program

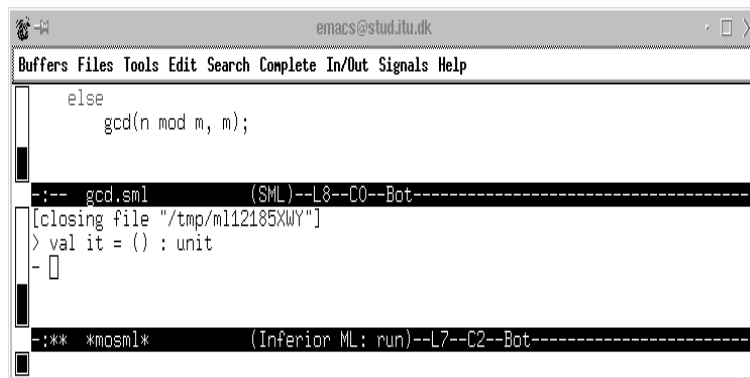



Figure 4: The SML session window

- Meta-*x* means: hold down the “Alt” or “Alt function” key and press the key *x*.
- Ctrl-*x* means: hold down the “Ctrl” key and press the key *x*.

This a brief step-by-step guide to editing programs.

1. Start Emacs by use of the emacs button in the button-bar. An empty Emacs window called **scratch** appears. Move the cursor to the Emacs window and click the left mouse button to select this window. Do not type your program in the **scratch** window.
2. To create an SML program buffer for a program called “gcd.sml”, type Ctrl-x Ctrl-f, then gcd.sml, then press return. (Alternatively, use the Files menu.)
If the file gcd.sml exists on disk already, it is loaded into Emacs. Otherwise a new empty buffer is created. The suffix “.sml” is important: it puts the buffer in Sml mode.
3. Type and edit your program in the SML program buffer. Editing commands are described in the reference card.
4. Type Ctrl-x Ctrl-s to save the program in a file on the disk. (Alternatively, use the Files menu.)
5. To load the program into an SML session, type Ctrl-c Ctrl-b in the SML program buffer. Press return twice to answer the questions M1 command: mosml and Any args:.

This creates an SML session, and loads the program into the SML session.

6. Type `Ctrl-c Ctrl-s` to see the output from the SML session. The SML session will be in a new window named `mosml`, as in Figure 4 on page 6.
7. Resize and move the Emacs window (as described in Section 3) to make the two Emacs buffers `gcd.sml` and `*mosml*` larger. You can left-click on the buffer you want active.
8. If the program has syntax errors or type errors, the SML session prints messages about them. Use `Ctrl-c `` to move the cursor to the position where an error has been located (on PC's with danish key layout the character ``` is the upper case of the key  just to the left of the backspace key). Edit the program to correct the errors. When you have finished correcting the errors, go back to step 4 above. Typing `Ctrl-c Ctrl-b` again will load the corrected program into the existing SML session, and SML prints the types of the functions and values defined.
9. Select the SML session buffer `*mosml*` window by moving the mouse to the buffer and click the left mouse button. Enter some expressions to test your program, for instance `gcd(34, 2);`. To go back to the SML program buffer (`gcd.sml`), move the mouse to the buffer and click the left mouse button.
10. When you have finished editing and testing, terminate the SML session by typing `Ctrl-c Ctrl-d` in the SML session buffer (`*mosml*`).
11. To print out the program or the output from the SML session, select the appropriate buffer and activate **Print buffer** or **Postscript Print buffer** in the **TOOLS** menu (move the mouse to **TOOLS** at the top of the window, press down the left mouse button, keep the button pressed down, move the cursor to the wanted item, and release the button). The default printer is `p165`. To choose another printer use the menu "Choose Printer" in the "Tools" menu.
12. To terminate Emacs, type `Ctrl-x Ctrl-c` or press the close button in the window (cf. Section 3). If you have not saved your program file, it will ask "Save file ...?", to which you should answer "y".

Key combinations like `Ctrl-c Ctrl-b` for sending a program to the SML compiler are specific for the *sml mode* of Emacs which is activated by the extension `.sml` of the file name. An overview of these special functions can be found in the `Sml` menu (click on `Sml` in the Emacs window).

If you get into a mess with Emacs, try `Ctrl-g` once or twice to interrupt it. If this does not help, type `Ctrl-x Ctrl-c` (or press the close button in the window) to terminate Emacs, and answer any questions it asks.

6.1 SML system functions

This section describes a selection of built-in functions for controlling the Moscow ML compiler in interactive mode.

<i>id</i>	type	value	exception
compile	string -> unit	compile unit (<i>U.sml</i> or <i>U.sig</i>)	Fail
load	string -> unit	Load unit and any units it needs	Fail
loadOne	string -> unit	Load unit only	Fail
printVal	'a -> 'a	Print value on stdout	
printDepth	int ref	Limit printed data depth	
printLength	int ref	Limit printed list and vector length	
quit	unit -> unit	Quit the interactive SML system	
system	string -> unit	Execute operating system command	
use	string -> unit	Include file in program	
verbose	bool ref	Permit feedback from compile	

The entities `printDepth`, `printLength`, and `verbose` are set by an *assignment*, that is, the assignment:

```
printLength := 100;
```

ensures that the response from SML will print the first 100 elements of a list.

7 File, directories and paths

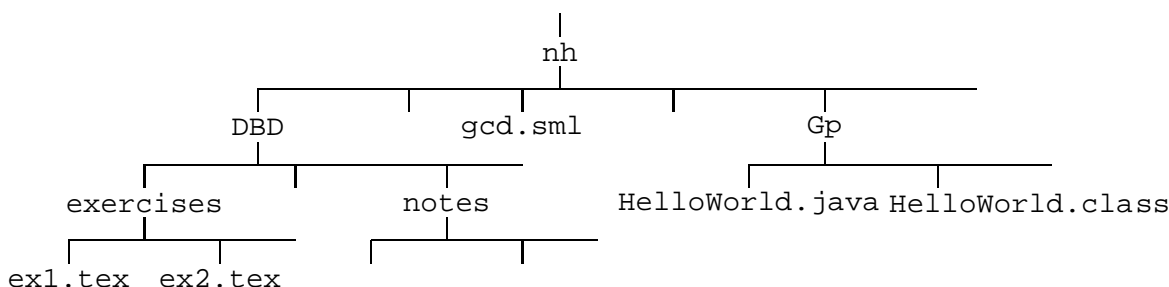


Figure 5: Possible file organisation for a user `nh`

Files on disk are organised in directories. A directory may contain subdirectories which gives a tree structure on the set of files. The private files for a user `nh` may be structured as shown in Figure 5. This user has a *home directory* `nh` containing subdirectories `Gp` for the course “Grundlæggende Programmering” and `DBD` for files related to the course “Design af brugergrænseflader og data”. The home directory contains furthermore a SML program `gcd.sml`, while the directory `DBD` contains a subdirectory `exercises` which contains two \LaTeX documents `ex1.tex` and `ex2.tex`, etc.

References to a file is made relative to the current directory. Hence, if `DBD` has been selected as the current directory, then the file `ex1.tex` in the `exercises` subdirectory is specified by the path:

```
exercises/ex1.tex
```


while the file `HelloWorld.java` is specified by a path using the parent directory symbol “`..`”:

```
../Gp/HelloWorld.java
```

Other users may read the file `ex1.tex` by using a path containing the user name preceded by a tilde character (unless reading by other users has been prohibited):

```
~nh/DBD/exercises/ex1.tex
```

Notice, that in Linux the file system is case sensitive, that is, the file name `DBD` and `dbd` refers to two different files.

To change current directory use `cd`. To get a file-listing in the current directory use `ls`.

8 Star Office

Star Office is a free office system similar to Microsoft Office. To use Star Office, you must first install it. You install it by choosing Star Office in the menu “Anvendelser” found by pressing the K button. This will start the installation. You are prompted to make a local or a network installation – you choose a local installation.

When Star Office is installed you start Star Office by choosing the same menu item, that is, Star Office in the menu “Anvendelser”.

Typing `soffice &` in an Xterm does also start Star Office.

9 L^AT_EX

You can also use L^AT_EX to type set documents. This document is type set using L^AT_EX. You use Emacs to edit your L^AT_EX document, and then the program `latex` to compile it into a “device independent file” (`.dvi`). As with Java and SML files, Emacs also has a mode for typing L^AT_EX documents.

More information on using L^AT_EX can be found in the note “The Not So Short Introduction to L^AT_EX2e” found on URL: <http://www.itu.dk/intranet/sysadm/cdroms/sir>

To try out L^AT_EX type in the following in an Emacs buffer named `text.tex`:

```
\documentclass[a4paper]{article}
\usepackage[english]{babel}
```

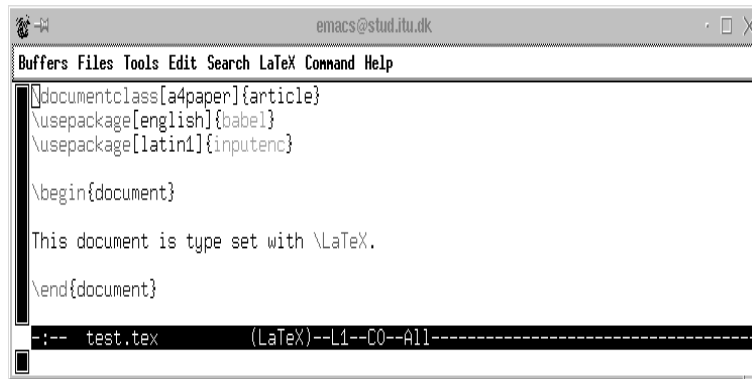


Figure 6: A \LaTeX buffer in Emacs

```
\usepackage[latin1]{inputenc}
\begin{document}
This document is type set with \LaTeX.
\end{document}
```

To compile the document type `Ctrl+c Ctrl+c` and answer the questions. If errors occur, then type `Ctrl+c Ctrl+l` to view the compiler messages. Correct the errors and type `Ctrl+c Ctrl+c` again. On successful compilation you can view the document typing `Ctrl+c Ctrl+c` again and press return.

10 Some useful UNIX commands

The commands described in this section are executed by typing text lines in a console window (see Sections 2 and 3). The descriptions are brief, just giving the name of the command and a one-line explanation, but you have easy access to more information by using the on-line documentation.

Online documentation

The online manual page system is a standard Unix feature, which you may use if you need information about a particular command. For example, to get a description of the Unix command `ls`, type “`man ls`” in an console window. Use the spacebar to move a page forward in the text, and use the arrow keys for scrolling. Type `q` to quit the manual page.

Changing password

At the beginning you get a password which has been generated automatically. This must be changed to one you invent yourself.

Creating and removing directories

<code>cd</code>	move to another directory.
<code>mkdir</code>	create new directory.
<code>rmdir</code>	remove directory.
<code>pwd</code>	show path to current directory

Using files

<code>cp</code>	copy file
<code>chmod</code>	change protection of file.
<code>find</code>	find files.
<code>locate</code>	find files based on a database – much faster than <code>find</code> .
<code>less</code>	display contents of file.
<code>ls</code>	list attributes of file(s).
<code>mv</code>	rename file.
<code>rm</code>	remove file.

Processes

<code>ps</code>	display current processes.
<code>kill</code>	kill process.
<code>free</code>	reports the amount of free memory.

Using floppy-disks

You may copy files to/from a floppy disk in DOS format using the below commands. The floppy disk unit on the PC is called “a :”.

<code>mcopy</code>	copy file to/from floppy disk.
<code>mdir</code>	display directory on floppy disk.
<code>mdel</code>	delete file on floppy disk.
<code>mformat</code>	format floppy disk.

Manipulation of text files

`diff` compares two files and reports the differences.
`grep` search file for pattern (regular expression).
`sort` sorts textfiles.
`head` prints first 10 lines of file.
`tail` prints (surprise) last 10 lines of file.
`a2ps` “pretty-prints” ASCII files on a printer – actually a PostScript file is generated and sent to a printer.
`wc` counts lines/words/letters in a file.

Redirection of input or output

To redirect the input or output for a program invoked by a command you use `<` and `>`. For example to redirect the output from a directory listing (command `ls`) to *outputfile*, you would do the following: `ls > outputfile`. To redirect the input from a file, you use `<`

Pipelines

Pipelines allow you to let the output of one program be used as input to another program. This is done in a command line containing several commands separated by bar characters “|”. Each bar denotes a pipeline which “pipes” the output of the command to the left of the bar to the input of the command to the right of the bar.

Example: let’s assume that you have created a program that outputs textlines, and you want to see the first 15 lines containing the word “helloworld”, you would execute the following: `myprogram | grep helloworld | head -n 15`

11 Desktop environments in the data bars

The graphics subsystem in Linux is an implementation of *The X Window System* or just *X* for short. It implements the basic graphic and windowing primitives and it is totally network transparent (unlike the graphics system of another well-known operating system). This means that you can run an entire session, or just a single application on any other computer in the data bar system from a PC in the data bar.

A *window manager* works on top of the graphics subsystem and it implements the lay-out of windows (borders, buttons, etc.), how to move windows around, how to use menus, and so on. The term *desktop environment* covers *X*, a windowmanager and a suite of utilities, configuration programs, games, editors and so on.

KDE - The K Desktop Environment

This is one of the most mature and full featured desktops available for Linux. The developers of KDE seem to have been inspired by MS Windows quite a bit, and KDE is nauseatingly alike it. Of course if you use MS Windows at home, you should feel quite at home in KDE. Using so called “themes” you can change the look of windows, borders, backgrounds ect., giving opportunity for countless hours of fun doing “window shopping”. KDE is the default window manager in the data bars.

WindowMaker

WindowMaker is a window manager with a twist. It is designed to be like OpenStep/NextStep, the famous operating system and computer that unfortunately failed back in the eighties. The “look and feel” is quite different from the other, but very effective.

GNOME - Gnu Object Model Environment

GNOME was originally created because of some licence problems concerning KDE. GNOME is much more configurable than KDE, and can be made to look like Star Trek on steroids. The “feel” can also be adjusted to your personal preference.

12 More about Linux

If you seek more information about Linux, searching the Internet is a good starting point. Another excellent source of information is the Linux User Groups (LUG's). Located in Copenhagen, we have a very large LUG: Skåne Sjælland Linux User Group (SSLUG), www.sslug.dk. We also have a LUG at ITU: (www2.itu.dk/linux). If you are interested in Linux, you should look at these two URL's. At www.sslug.dk, you also find an introductory text to Linux: “Linux - Friheden til at vælge” – mandatory reading if you are new to Linux.

You can get Linux free from many different companies. ITU also has a Linux distribution based on RedHat. Below we give a few URL's to some of the most popular suppliers of the Linux operating system:

Red Hat: www.redhat.com.

Debian: www.debian.org

SuSE: www.suse.com.

A lot of books exists on using Linux. All larger bookstores have them.

User-guides and HOWTO's can be found on <http://howto.linuxberg.com>

General computer and Linux related news on <http://www.slashdot.org>

Danish user groups

The Skåne-Sjællands Linux User Group (SSLUG) has a *lot* of material on their web-site. They also run mailing-lists/newsgroups where new-bies can get friendly help with all aspects of Linux. They also maintain a very nice user-guide, it is available on their website.

SSLUG - <http://www.sslug.dk> "Linux - Friheden til at vælge" <http://www.sslug.dk/linuxbog>

Getting Linux on your own machine

The data bar machines is running a customized version of Redhat 6.1. If you want to get Linux on your home PC, there are several options. It is possible to buy it at "Polyteknisk Boghandel", where you will get a nice manual and other goodies. Sometimes the computer magazines includes a CDROM with Linux on, and the IDG booklets also include a CDROM. It is also possible (and *legal*) just to burn a copy of someones CD. You can borrough a CD-ROM at ITU with an updated RedHat 6.1 Linux.

It's no problem if you want to use other distributions than Redhat, the only difference is that some files and scripts relating to the administration of the machine can be placed different.

Linux can coexist with MS Windows on the same machine, and can read both FAT32 and NTFS.