Linux Access HOWTO

Table of Contents

Table of Contents

Table of Contents

Linux Access HOWTO

Michael De La Rue, <access–howto@ed.ac.uk>

v2.11, 28 March 1997

The Linux Access HOWTO covers the use of adaptive technology with Linux, In particular, using adaptive technology to make Linux accessible to those who could not use it otherwise. It also covers areas where Linux can be used within more general adaptive technology solutions.

5.Physical Problems

6.General Programming Issues

7.Other Information

8.Software Packages

9.Hardware

10.Acknowledgements

1.Introduction

The aim of this document is to serve as an introduction to the technologies which are available to make Linux usable by people who, through some disability would otherwise have problems with it. In other words the target groups of the technologies are, the blind, the partially sighted, deaf and the physically disabled. As any other technologies or pieces of information are discovered they will be added.

The information here not just for these people (although that is probably the main aim) but also to allow developers of Linux to become aware of the difficulties involved here. Possibly the biggest problem is that, right now, very few of the developers of Linux are aware of the issues and various simple ways to make life simpler for implementors of these systems. This has, however, changed noticeably since the introduction of this document, and at least to a small extent because of this document, but also to a large extent due to the work of some dedicated developers, many of whom are mentioned in the document's Acknowledgements.

Please send any comments or extra information or offers of assistance to <access−howto@ed.ac.uk> This address might become a mailing list in future, or be automatically handed over to a future maintainer of the HOWTO, so please don't use it for personal email.

I don't have time to follow developments in all areas. I probably won't even read a mail until I have time to update this document. It's still gratefully received. If a mail is sent to the blind−list or the access−list, I *will* eventually read it and put any useful information into the document. Otherwise, please send a copy of anything interesting to the above email address.

Normal mail can be sent to

```
Linux Access HOWTO
23 Kingsborough Gardens
Glasgow G12 9NH
Scotland
U.K.
```

And will gradually make its way round the world to me. Email will be faster by weeks.

I can be personally contacted using <miked@ed.ac.uk>. Since I use mail filtering on all mail I receive, please use the other address except for personal email. This is most likely to lead to an appropriate response.

1.1 Distribution Policy

2.Comparing Linux with other Operating Systems

2.1 General Comparison

The best place to find out about this is in such documents as the `Linux Info Sheet', `Linux Meta FAQ' and `Linux FAQ' (see Linux Documentation). Major reasons for a visually impaired person to use Linux would include it's inbuilt networking which gives full access to the Internet. More generally, users are attracted by the full development environment included. Also, unlike most other modern GUI environments, the graphical front end to Linux (X Windows) is clearly separated from the underlying environment and there is a complete set of modern programs such as World Wide Web browsers and fax software which work directly in the non graphical environment. This opens up the way to provide alternative access paths to the systems functionality; Emacspeak is a good example.

For other users, the comparison is probably less favourable and less clear. People with very specific and complex needs will find that the full development system included allows properly customised solutions. However, much of the software which exists on other systems is only just beginning to become available. More development is being done however in almost all directions.

2.2 Availability of Adaptive Technology

There is almost nothing commercial available *specifically* for Linux. There is a noticeable amount of free software which would be helpful in adaptation, for example, a free speech synthesiser and some free voice control software. There are also a number of free packages which provide good support for certain Braille terminals, for example.

2.3 Inherent Usability

Linux has the vast advantage over Windows that most of it's software is command line oriented. This is now changing and almost everything is now available with a graphical front end. However, because it is in origin a programmers operating system, line oriented programs are still being written covering almost all new areas of interest. For the physically disabled, this means that it is easy to build custom programs to suit their needs. For the visually impaired, this should make use with a speech synthesiser or Braille terminal easy and useful for the foreseeable future.

Linux's multiple virtual consoles system make it practical to use as a multi−tasking operating system by a visually impaired person working directly through Braille.

The windowing system used by Linux (X11) comes with many programming tools, and should be adaptable. However, in practice, the adaptive programs available up till now have been more primitive than those on the Macintosh or Windows. They are, however, completely free (as opposed to hundreds of pounds) and the quality is definitely improving.

In principle it should be possible to put together a complete, usable Linux system for a visually impaired person for about $500 (cheap & nasty PC + sound card). This compares with many thousands of dollars for other operating systems (screen reader software/ speech synthesiser hardware). I have yet to see this. I doubt it would work in practice because the software speech synthesisers available for Linux aren't yet sufficiently good. For a physically disabled person, the limitation will still be the expense of input hardware.

3.Visually Impaired

I'll use two general categories here. People who are partially sighted and need help seeing / deciphering / following the text and those who are unable to use any visual interface whatsoever.

3.1 Seeing the Screen with Low Vision

There are many different problems here. Often magnification can be helpful, but that's not the full story. Sometimes people can't track motion, sometimes people can't find the cursor unless it moves. This calls for a range of techniques, the majority of which are only just being added to X.

SVGATextMode

This program is useful for improving the visibility of the normal text screen that Linux provides. The normal screen that Linux provides shows 80 characters across by 25 vertically. This can be changed (and the quality of those characters improved) using SVGATextMode. The program allows full access to the possible modes of an SVGA graphics card. For example, the text can be made larger so that only 50 by 15 characters appear on the screen. There isn't any easy way to zoom in on sections of a screen, but you can resize when needed.

X Window System

For people who can see the screen there are a large number of ways of improving X. They don't add up to a coherent set of features yet, but if set up correctly could solve many problems.

Different Screen Resolutions

The X server can be set up with many different resolutions. A single key press can then change between them allowing difficult to read text to be seen.

In the file `/etc/XF86Config`, you have an entry in the Screen section with a line beginning with modes. If, for example, you set this to

```
Modes       "1280x1024" "1024x768" "800x600" "640x480" "320x240"
```

with each mode set up correctly (which requires a reasonably good monitor for the highest resolution mode), you will be able to have four times screen magnification, switching between the different levels using

`Ctrl+Alt+Keypad-Plus` and `Ctrl+Alt+Keypad-Minus`

Moving the mouse around the screen will scroll you to different parts of the screen. For more details on how to set this up you should see the documentation which comes with the **XFree86** X server.

Screen Magnification

There are several known screen magnification programs, `xmag` which will magnify a portion of the screen as much as needed but is very primitive. Another one is `xzoom`. Previously I said that there had to be something better than `xmag`, well this is it. See section xzoom.

Another program which is available is `puff`. This is specifically oriented towards visually impaired users. It provides such features as a box around the pointer which makes it easier to locate. Other interesting features of `puff` are that, if correctly set up, it is able to select and magnify portions of the screen as they are updated. However, there seem to be interacations between `xpuff` and the window manager which could make it difficult to use. When used with my `fvwm` setup, it didn't respond at all to key presses. However using `twm` improved the situation.

The final program which I have seen working is `dynamag`. This again has some specific advantages such as the ability to select a specific area of the screen and monitor it, refreshing the magnified display at regular intervals between a few tenths of a second at twenty seconds. `dynamag` is part of the UnWindows distribution. See UnWindows for more details.

Change Screen Font

The screen fonts all properly written X software should be changeable. You can simply make it big enough for you to read. This is generally accomplished by putting a line the file `.Xdefaults` which should be in your home directory. By putting the correct lines in this you can change the fonts of your programs, for example

```
Emacs.font: -sony-fixed-medium-r-normal--16-150-75-75-c-80-iso8859-*
```

To see what fonts are available, use the program `xfontsel` under X.

There should be some way of changing things at a more fundamental level so that everything comes out with a magnified font. This could be done by renaming fonts, and by telling telling font generating programs to use a different level of scaling. If someone gets this to work properly, please send me the details of how you did it.

Cross Hair Cursors etc..

For people that have problems following cursors there are many things which can help;

- cross−hair cursors (horizontal and vertical lines from the edge of the screen)
- flashing cursors (flashes when you press a key)

No software I know of specifically provides a cross hair cursor. `puff`, mentioned in the previous section does however provide a flashing box around the cursor which can make it considerably easier to locate.

For now the best that can be done is to change the cursor bitmap. Make a bitmap file as you want it, and another one which is the same size, but completely black. Convert them to the XBM format and run

```
        xsetroot -cursor cursorfile.xbm black-file.xbm
```

actually, if you understand masks, then the black−file doesn't have to be completely black, but start with it like that. The `.Xdefaults` file controls cursors used by actual applications. For much more information, please see the X Big Cursor mini−HOWTO, by Joerg Schneider <schneid@ira.uka.de>.

Audio

Provided that the user can hear, audio input can be very useful for making a more friendly and communicative computing environment. For a person with low vision, audio clues can be used to help locate the pointer (see UnWindows). For a console mode user using Emacspeak (see Emacspeak), the audio icons available will provide very many useful facilities.

Setting up Linux audio is covered in the Linux Sound HOWTO (see Linux Documentation). Once sound is set up, sounds can be played with the `play` command which is included with most versions of Linux. This is the way to use my version of UnWindows.

Producing Large Print

Using large print with Linux is quite easy. There are several techniques.

LaTeX / TeX

LaTeX is an extremely powerful document preparation system. It may be used to produce large print documents of almost any nature. Though somewhat complicated to learn, many documents are produced using LaTeX or the underlying typesetting program, TeX.

this will produce some reasonably large text

```
\font\magnifiedtenrm=cmr10 at 20pt  % setup a big font
\magnifiedtenrm
this is some large text
\bye
```

For more details, see the LaTeX book which is available in any computer book shop. There are also a large number of introductions available on the internet.

Outputting Large Text

Almost all Linux printing uses postscript, and Linux can drive almost any printer using it. I output large text teaching materials using a standard Epson dot matrix printer.

For users of X, there are various tools available which can produce large Text. These include `LyX`, and many commercial word processors.

3.2 Aids for Those Who Can't Use Visual Output

For someone who is completely unable to use a normal screen there are two alternatives Braille and Speech. Obviously for people who also have hearing loss, speech isn't always useful, so Braille will always be important.

If you can choose, which should you choose? This is a matter of `vigorous' debate. Speech is rapid to use, reasonably cheap and especially good for textual applications (e.g. reading a long document like this one).

Problems include needing a quiet environment, possibly needing headphones to work without disturbing others and avoid being listened in on by them (not available for all speech synthesisers).

Braille is better for applications where precise layout is important (e.g. spreadsheets). Also can be somewhat more convenient if you want to check the beginning of a sentence when you get to the end. Braille is, however, much more expensive and slower for reading text. Obviously, the more you use Braille, the faster you get. Grade II Braille is difficult to learn, but is almost certainly worth it since it is much faster. This means that if you don't use Braille for a fair while you can never discover its full potential and decide for yourself. Anyway, enough said on this somewhat controversial topic.

based on original by James Bowden <jrbowden@bcs.org.uk>

Braille Terminals

Braille terminals are normally a line or two of Braille. Since these are at most 80 characters wide and normally 40 wide, they are somewhat limited. I know of two kinds

- Hardware driven Braille terminals.
- Software driven Braille terminals.

The first kind works only when the computer is in text mode and reads the screen memory directly. See section hardware driven Braille terminals.

The second kind of Braille terminal is similar, in many ways, to a normal terminal screen of the kind Linux supports automatically. Unfortunately, they need special software to make them usable.

There are two packages which help with these. The first, `BRLTTY`, works with several Braille display types and the authors are keen to support more as information becomes available. Currently `BRLTTY` supports Tieman B.V.'s CombiBraille series, Alva B.V.'s ABT3 series and Telesensory Systems Inc.'s PowerBraille and Navigator series displays. The use of Blazie Engineering's Braille Lite as a Braille display is discouraged, but support may be renewed on demand. See section Software Braille Terminals.

The other package I am aware of is Braille Enhanced Screen. This is designed to work on other UNIX systems as well as Linux. This should allow user access to a Braille terminal with many useful features such as the ability to run different programs in different `virtual terminals' at the same time.

Speech Synthesis

Speech Synthesisers take (normally) ASCII text and convert it into actual spoken output. It is possible to have these implemented as either hardware or software. Unfortunately, the free Linux speech synthesisers are, reportedly, not good enough to use as a sole means of output.

Hardware speech synthesisers are the alternative. The main one that I know of that works is DECtalk from Digital, driven by `emacspeak`. However, at this time (March 1997) a driver for the Doubletalk synthesiser has been announced. Using `emacspeak` full access to all of the facilities of Linux is fairly easy. This includes the normal use of the shell, a world wide web browser and many other similar features, such as email. Although, it only acts as a plain text reader (similar to IBM's one for the PC) when controling programs it doesn't understand, with those that it does, it can provide much more sophisticated control. See section Emacspeak for more information about `emacspeak`.

Handling Console Output

When it starts up, Linux at present puts all of its messages straight to the normal (visual) screen. This could be changed if anyone with a basic level of kernel programming ability wants to do it. This means that it is impossible for most Braille devices to get information about what Linux is doing before the operating system is completely working.

It is only at that stage that you can start the program that you need for access. If the `BRLTTY` program is used and run very early in the boot process, then from this stage on the messages on the screen can be read. Most hardware and software will still have to wait until the system is completely ready. This makes administering a Linux system difficult, but not impossible for a visually impaired person. Once the system is ready however, you can scroll back by pressing (on the default keyboard layout) Shift−PageUP.

There is one Braille system that can use the console directly, called the Braillex. This is designed to read directly from the screen memory. Unfortunately the normal scrolling of the terminal gets in the way of this. If you are using a Kernel newer than 1.3.75, just type `linux no-scroll` at the LILO prompt or configure LILO to do this automatically. If you have an earlier version of Linux, see section Screen Memory Braille Terminals

The other known useful thing to do is to use sounds to say when each stage of the boot process has been reached. (T.V. Raman suggestion)

Optical Character Recognition

There is a free Optical Character Recognition (OCR) program for Linux called `xocr`. In principle, if it is good enough, this program would allow visually impaired people to read normal books to some extent (accuracy of OCR is never high enough..). However, according to the documentation, this program needs training to recognise the particular font that it is going to use and I have no idea how good it is since I don't have the hardware to test it.

3.3 Beginning to Learn Linux

Beginning to learn Linux can seem difficult and daunting for someone who is either coming from no computing background or from a pure DOS background. Doing the following things may help:

- Learn to use Linux (or UNIX) on someone else's system before setting up your own.
- Initially control Linux from your own known speaking/Braille terminal. If you plan to use speech, you may want to learn `emacs` now. You can learn it as you go along though. See below
- If you come from an MS−DOS background, read the DOS2Linux Mini HOWTO for help with converting (see The Linux HOWTO Documents).

The Emacspeak HOWTO written by Jim Van Zandt (<jrv@vanzandt.mv.com>) covers this in much more detail (see The Linux HOWTO Documents).

If you are planning to use Emacspeak, you should know that Emacspeak does not attempt to teach Emacs, so in this sense, prior knowledge of Emacs would always be useful. This said, you certainly do not need to know much about Emacs to start using Emacspeak. In fact, once Emacspeak is installed and running, it provides a fluent interface to the rich set of online documentation including the info pages, and makes learning what you need a lot easier.

"In summary: starting to use Emacspeak takes little learning. Getting the full mileage out of Emacs and Emacspeak, especially if you intend using it as a replacement for X Windows as I do does involve eventually becoming familiar with a lot of the Emacs extensions; but this is an incremental process and does not need to be done in a day." − *T.V.Raman*

One other option which may be interesting are the RNIB training tapes which include one covering UNIX. These can be got from

```
RNIB
Customer Services
PO Box 173
Peterborough
Cambridgeshire PE2 6WS
Tel: 01345 023153 (probably only works in UK)
```

3.4 Braille Embossing

Linux should be the perfect platform to drive a Braille embosser from. There are many formatting tools which are aimed specifically at the fixed width device. A Braille embosser can just be connected to the serial port using the standard Linux printing mechanisms. For more info see the Linux Printing HOWTO.

There is a free software package which acts as a multi−lingual grade two translator available for Linux from the American ``National Federation for the Blind''. This is called NFBtrans. See section NFB translator for more details.

4.Hearing Problems

For the most part there is little problem using a computer for people with hearing problems. Almost all of the output is visual. There are some situations where sound output is used though. For these, the problem can sometimes be worked round by using visual output instead.

4.1 Visual Bells

By tradition, computers go `beep' when some program sends them a special code. This is generally used to get attention to the program and for little else. Most of the time, it's possible to replace this by making the entire screen (or terminal emulator) flash. How to do this is very variable though.

xterm (under X)

for xterm, you can either change the setting by pressing the middle mouse button while holding down the control key, or by putting a line with just `XTerm*visualBell: true' (not the quotes of course) in the file `.Xdefaults` in your home directory.

the console (otherwise)

The console is slightly more complex. Please see Alessandro Rubini's Visual Bell mini HOWTO for details on this. Available along with all the other Linux documentation (see section other Linux documents). Mostly the configuration has to be done on a per application basis, or by changing the Linux Kernel its self.

5.Physical Problems

Many of these problems have to be handled individually. The needs of the individual, the ways that they can generate input and other factors vary so much that all that this HOWTO can provide is a general set of pointers to useful software and expertise.

5.1 Unable to Use a Mouse/Pointer

Limited mobility can make it difficult to use a mouse. For some people a tracker ball can be a very good solution, but for others the only possible input device is a keyboard (or even something which simulates a keyboard). For normal use of Linux this shouldn't be a problem (but see the section Making the keyboard behave), but for users of X, this may cause major problems under some circumstances.

Fortunately, the `fvwm` window manager has been designed for use without a pointer and most things can be done using this. I actually do this myself when I lose my mouse (don't ask) or want to just keep typing. `fvwm` is included with all distributions of Linux that I know of. Actually using other programs will depend on their ability to accept key presses. Many X programs do this for all functions. Many don't. I sticky mouse keys, which are supposedly present in the current release of X should make this easier.

Unable to Use a Keyboard

People who are unable to use a keyboard normally can sometimes use one through a headstick or a mouthstick. This calls for special setup of the keyboard. Please see also the section Making the keyboard behave.

Other Input Hardware (X Windows System only)

For others, the keyboard cannot be used at all and only pointing devices are available. In this case, no solution is available under the standard Linux Console and X will have to be used. If the X–Input extension can be taught to use the device and the correct software for converting pointer input to characters can be found (I haven't seen it yet) then any pointing should be usable without a keyboard.

There are a number of devices worth considering for such input such as touch screens and eye pointers. Many of these will need a `device driver' written for them. This is not terribly difficult if the documentation is available, but requires someone with good C programming skills. Please see the *Linux Kernel Hackers guide* and other kernel reference materials for more information. Once this is set up, it should be possible to use these devices like a normal mouse.

Controlling Physical Hardware From Linux

The main group of interest here are the Linux Lab Project. Generally, much GPIB (a standard interface to scientific equipment, also known as the IEEE bus) hardware can be controlled. This potentially gives much potential for very ambitious accessibility projects. As far as I know none have yet been attempted.

5.2 Speech Recognition

Speech recognition is a very powerful tool for enabling computer use. There are two recognition systems that I know of for Linux, the first is `ears` which is described as ``recognition is not optimal. But it is fine for playing and will be improved'', the second is `AbbotDemo` ``A speaker independent continuous speech recognition system'' which may well be more interesting, though isn't available for commercial use without prior arrangement. See the Linux software map for details (see section other Linux documents).

5.3 Making the Keyboard Behave

X Window System.

The latest X server which comes with Linux can include many features which assist in input. This includes such features as StickKeys, MouseKeys, RepeatKeys, BounceKeys, SlowKeys, and TimeOut. These allow customisation of the keyboard to the needs of the user. These are provided as part of the `XKB`> extension in versions of X after version 6.1. To find out your version and see whether you have the extension installed, you can try.

```
xdpyinfo -queryExtensions
```

Getting Rid of Auto Repeat

To turn off key repeat on the Linux console run this command (I think it has to be run once per console; a good place to run it would be in your login files, `.profile` or `.login` in your home directory).

```
setterm -repeat off
```

To get rid of auto repeat on any X server, you can use the command

```
xset -r
```

which you could put into the file which get runs when you start using X (often `.xsession` or `.xinit` under some setups)

Both of these commands are worth looking at for more ways of changing behaviour of the console.

Macros / Much input, few key presses

Often in situations such as this, the biggest problem is speed of input. Here the most important thing to aim for is the most number of commands with the fewest key presses. For users of the shell (`bash` / `tcsh`) you should look at the manual page, in particular command and filename completion (press the tab key and bash tries to guess what should come next). For information on macros which provide sequences of commands for just one key press, have a look at the Keystroke HOWTO.

Sticky Keys

Sticky keys are a feature that allow someone who can only reliably press one button at a time to use a keyboard with all of the various modifier keys such as shift and control. These keys, instead of having to be held on at the same time as the other key instead become like the caps lock key and stay on while the other key is pressed. They may then either switch off or stay on for the next key depending on what is needed. For information about how to set this up please see the Linux Keyboard HOWTO, especially section `I can use only one finger to type with' (section 15 in the version I have) for more information on this. – Information from Toby Reed.

6.General Programming Issues

Many of the issues worth taking into account are the same when writing software which is designed to be helpful for access as when trying to follow good design.

6.1 Try to Make it Easy to Provide Multiple Interfaces

If your software is only usable through a graphical interface then it can be very hard to make it usable for someone who can't see. If it's only usable through a line oriented interface, then someone who can't type will have difficulties.

Provide keyboard shortcuts as well as the use of the normal X pointer (generally the mouse). You can almost certainly rely on the user being able to generate key presses to your application.

6.2 Make software configurable.

If it's easy to change fonts then people will be able to change to one they can read. If the colour scheme can be changed then people who are colour blind will be more likely to be able to use it. If fonts can be changed easily then the visually impaired will find your software more useful.

6.3 Test the Software on Users.

If you have a number of people use your software, each with different access problems then they will be more likely to point up specific problems. Obviously, this won't be practical for everybody, but you can always ask for feedback.

6.4 Make Output Distinct

Where possible, make it clear what different parts of your program are what. Format error messages in a specific way to identify them. Under X, make sure each pane of your window has a name so that any screen reader software can identify it.

6.5 Licenses

Some software for Linux (though none of the key programs) has license like `not for commercial use'. This could be quite bad for a person who starts using the software for their personal work and then possibly begins to be able to do work they otherwise couldn't with it. This could be something which frees them from financial and other dependence on others people. Even if the author of the software is willing to make exceptions, it makes the user vulnerable both to changes of commercial conditions (some company buys up the rights) and to refusal from people they could work for (many companies are overly paranoid about licenses). It is much better to avoid this kind of licensing where possible. Protection from commercial abuse of software can be obtained through more specific licenses like the GNU Public License or Artistic License where needed.

7.Other Information

7.1 Linux Documentation

The Linux documentation is critical to the use of Linux and most of the documents mentioned here should be included in recent versions of Linux, from any source I know of.

If you want to get the documentation on the Internet, here are some example sites. These should be mirrored at most of the major FTP sites in the world.

- ftp.funet.fi (128.214.6.100) : /pub/OS/Linux/doc/
- tsx−11.mit.edu (18.172.1.2) : /pub/linux/docs/
- sunsite.unc.edu (152.2.22.81) : /pub/Linux/docs/

The Linux Info Sheet

A simple and effective explanation of what Linux is. This is one of the things that you should hand over when you want to explain why you want Linux and what it is good for.

The Linux Info Sheet is available on the World Wide Web from http://sunsite.unc.edu/mdw/HOWTO/INFO−SHEET.html and other mirrors.

The Linux Meta FAQ

A list of other information resources, much more complete than this one. The meta FAQ is available on the World Wide Web from http://sunsite.unc.edu/mdw/HOWTO/META−FAQ.html and other mirrors

The Linux Software Map

The list of software available for Linux on the Internet. Many of the packages listed here were found through this. The LSM is available in a searchable form from http://www.boutell.com/lsm/. It is also available in a single text file in all of the FTP sites mentioned in section Linux Documentation.

The Linux HOWTO documents

The HOWTO documents are the main documentation of Linux. This Access HOWTO is an example of one.

The home site for the Linux Documentation Project which produces this information is http://sunsite.unc.edu/mdw/linux.html. There are also many companies producing these in book form. Contact a local Linux supplier for more details.

The Linux HOWTO documents will be in the directory `HOWTO` in all of the FTP sites mentioned in section Linux Documentation.

The Linux FAQ

A list of `Frequently Asked Questions' with answers which should solve many common questions. The FAQ list is available from http://www.cl.cam.ac.uk/users/iwj10/linux−faq/ as well as all of the FTP sites mentioned in section Linux Documentation.

7.2 Mailing Lists

There are two lists that I know of covering these issues specifically for Linux. There are also others which it is worth researching which cover computer use more generally. Incidentally, if a mail is sent to these lists I *will* read it eventually and include any important information in the Access−HOWTO, so you don't need to send me a separate copy unless it's urgent in some way.

The Linux Access List

This is a general list covering Linux access issues. It is designed `to service the needs of users and developers of the Linux OS and software who are either disabled or want to help make Linux more accessible'. To subscribe send email to <majordomo@ssv1.union.utah.edu> and in the BODY (not the subject) of the email message put:

```
subscribe linux-access <your-email-address>
```

The Linux Blind List

This is a mailing list covering Linux use for blind users. There is also a list of important and useful software being gathered in the list's archive. To subscribe send mail to <blinux–list–request@redhat.com> with the `subject: help`. This list is now moderated.

7.3 WWW References

The World Wide Web is, by it's nature, very rapidly changing. If you are reading this document in an old version then some of these are likely to be out of date. The original version that I maintain on the WWW shouldn't go more than a month or two out of date, so refer to that please.

Linux Documentation is available from http://sunsite.unc.edu/mdw/linux.html

Linux Access On the Web http://www.tardis.ed.ac.uk/~mikedlr/access/ with all of the versions of the HOWTO in http://www.tardis.ed.ac.uk/~mikedlr/access/HOWTO/. Preferably, however, download from one of the main Linux FTP sites. If I get a vast amount of traffic I'll have to close down these pages and move them elsewhere.

The BLINUX Documentation and Development Project http://leb.net/blinux/. "The purpose of The BLINUX Documentation and Development Project is to serve as a catalyst which will both spur and speed the development of software and documentation which will enable the blind user to run his or her own Linux workstation."

Emacspeak WWW page http://cs.cornell.edu/home/raman/emacspeak/emacspeak.html

BRLTTY unofficial WWW page http://www.sf.co.kr/t.linux/new/brltty.html

Yahoo (one of the most major Internet catalogues) http://www.yahoo.com/Society_and_Culture/Disabilities/Adaptive_Technology/

The Linux Lab Project http://www.fu–berlin.de/~clausi/.

The BLYNX pages: Lynx Support Files Tailored For Blind and Visually Handicapped Users
http://leb.net/blinux/blynx/.

7.4 Suppliers

This is a UK supplier for the Braillex.

```
Alphavision Limited
```

7.5 Manufacturers

Alphavision

I think that they are a manufacturer? RNIB only lists them as a supplier, but others say they make the Braillex.

```
Alphavision Ltd
Seymour House
Copyground Lane
High Wycombe
Bucks HP12 3HE
England
U.K.
```

Phone

+44 1494−530 555

Linux Supported Alphavision AT Products

- Braillex

Blazie Engineering

The Braille Lite was supported in the original version of `BRLTTY`. That support has now been discontinued. If you have one and want to use it with Linux then that may be possible by using this version of the software.

```
Blazie Engineering
105 East Jarrettsville Rd.
Forest Hill, MD 21050
```

```
U.S.A.
```

Phone

+1 (410) 893−9333

FAX

+1 (410) 836−5040

BBS

+1 (410) 893−8944

E−Mail

<info@blazie.com>

WWW

http://www.blazie.com/

Blazie AT Products

- Braille Lite (support discontinued)

Digital Equipment Corporation

```
Digital Equipment Corporation
P.O. Box CS2008
Nashua
NH 03061-2008
U.S.A
```

Order

+1 800−722−9332

Tech info

+1 800−722−9332

FAX

+1 603−884−5597

WWW

http://www.digital.com/

Linux Supported DEC AT Products

- DECTalk Express

Kommunikations−Technik Stolper GmbH


```
KTS Stolper GmbH
Herzenhaldenweg 10
73095 Albershausen
Germany
```

Phone

+49 7161 37023

Fax

+49 7161 32632

Linux Supported KTG AT Products

- Brailloterm

8.Software Packages

References in this section are taken directly from the Linux Software map which can be found in all standard places for Linux documentation and which lists almost all of the software available for Linux.

8.1 Emacspeak

Emacspeak is the software side of a speech interface to Linux. Any other character based program, such as a WWW browser, or `telnet` or another editor can potentially be used within `emacspeak`. The main difference between it and normal screen reader software for such operating systems as DOS is that it also has a load more extra features. It is based in the emacs text editor.

A text editor is generally just a program which allows you to change the contents of a file, for example, adding new information to a letter. Emacs is in fact far beyond a normal text editor, and so this package is much more useful than you might imagine. You can run any other program from within emacs, getting any output it generates to appear in the emacs terminal emulator.

The reason that emacs is a better environment for Emacspeak is that it can can understand the layout of the screen and can intelligently interpret the meaning of, for example, a calendar, which would just be a messy array of numbers otherwise. The originator of the package manages to look after his own Linux machine entirely, doing all of the administration from within emacs. He also uses it to control a wide variety of other machines and software directly from that machine.

Emacspeak is included within the Debian Linux distribution and is included as contributed software within the Slakware distribution. This means that it is available on many of the CDROM distributions of Linux. By the time this is published, the version included should be 5 or better, but at present I only have version 4 available for examination.

```
Begin3
Title:          emacspeak - a speech output interface to Emacs
Version:        4.0
Entered-date:   30MAY96
Description:    Emacspeak is the first full-fledged speech output
                system that will allow someone who cannot see to work
                directly on a UNIX system. (Until now, the only option
                available to visually impaired users has been to use a
                talking PC as a terminal.) Emacspeak is built on top
                of Emacs. Once you start emacs with emacspeak loaded,
                you get spoken feedback for everything you do. Your
                mileage will vary depending on how well you can use
                Emacs.  There is nothing that you cannot do inside
                Emacs:-)
Keywords:       handicap access visually impaired blind speech emacs
Author:         raman@adobe.com (T. V. Raman)
Maintained-by:  jrv@vanzandt.mv.com (Jim Van Zandt)
Primary-site:   sunsite.unc.edu apps/sound/speech
                124kB   emacspeak-4.0.tgz
Alternate-site:
Original-site:  http://www.cs.cornell.edu /pub/raman/emacspeak
                123kB   emacspeak.tar.gz/Info/People/raman/emacspeak/emacspeak.tar.gz
Platforms:      DECtalk Express or DEC Multivoice speech synthesizer,
                GNU FSF Emacs 19 (version 19.23 or later) and TCLX
                7.3B (Extended TCL).
Copying-policy: GPL
End
```

8.2 BRLTTY

This is a program for running a serial port Braille terminal. It has been widely tested and used, and supports a number of different kinds of hardware (see the Linux Software Map entry below).

The maintainer is, Nikhil Nair <nn201@cus.cam.ac.uk>. The other people working on it are Nicolas Pitre <nico@cam.org> and Stephane Doyon <doyons@jsp.umontreal.ca>. Send any comments to all of them.

The authors seem keen to get support in for more different devices, so if you have one you should consider contacting them. They will almost certainly need programming information for the device, so if you can contact your manufacturer and get that they are much more likely to be able to help you.

A brief feature list (from their README file) to get you interested

- Full implementation of the standard screen review facilities.
- A wide range of additional optional features, including blinking cursor and capital letters, screen freezing for leisurely review, attribute display to locate highlighted text, hypertext links, etc.
- `Intelligent' cursor routing. This allows easy movement of the cursor in text editors etc. without moving the hands from the Braille display.
- A cut & paste function. This is particularly useful for copying long filenames, complicated commands etc.
- An on−line help facility.
- Support for multiple Braille codes.
- Modular design allows relatively easy addition of drivers for other Braille displays, or even (hopefully) porting to other Unix−like platforms.

```
Begin3
Title:          BRLTTY - Access software for Unix for a blind person
                         using a soft Braille terminal
Version:        1.0.2, 17SEP96
Entered-date:   17SEP96
Description:    BRLTTY is a daemon which provides access to a Unix console
                for a blind person using a soft Braille display (see the
                README file for a full explanation).

                BRLTTY only works with text-mode applications.

                We hope that this system will be expanded to support
                other soft Braille displays, and possibly even other
                Unix-like platforms.
Keywords:       Braille console access visually impaired blind
Author:         nn201@cus.cam.ac.uk (Nikhil Nair)
                nico@cam.org (Nicolas Pitre)
                doyons@jsp.umontreal.ca (Stephane Doyon)
                jrbowden@bcs.org.uk (James Bowden)
Maintained-by:  nn201@cus.cam.ac.uk (Nikhil Nair)
Primary-site:   sunsite.unc.edu /pub/Linux/system/Access
                110kb brltty-1.0.2.tar.gz (includes the README file)
                  6kb brltty-1.0.2.README
                  1kb brltty-1.0.2.lsm
Platforms:      Linux (kernel 1.1.92 or later) running on a PC or DEC Alpha.
                Not X/graphics.
```

```
                Supported Braille displays (serial communication only):
                  - Tieman B.V.: CombiBraille 25/45/85;
                  - Alva B.V.: ABT3xx series;
                  - Telesensory Systems Inc.: PowerBraille 40 (not 65/80),
                    Navigator 20/40/80 (latest firmware version only?).
Copying-policy: GPL
End
```

8.3 Screen

Screen is a standard piece of software to allow many different programs to run at the same time on one terminal. It has been enhanced to support some Braille terminals (those from Telesensory) directly.

8.4 Rsynth

This is a speech synthesiser listed in the Linux Software Map. It doesn't apparently work well enough for use by a visually impaired person. Use hardware instead, or improve it.. a free speech synthesiser would be really really useful.

8.5 xocr

`xocr` is a package which implements optical character recognition for Linux. As with `Rsynth`, I don't think that this will be acceptable as a package for use as a sole means of input by a visually impaired person. I suspect that the algorithm used means that it will need to be watched over by someone who can check that it is reading correctly. I would love to be proved wrong.

8.6 xzoom

`xzoom` is a screen magnifier, in the same vein as `xmag`, but sufficiently better to be very useful to a visually impaired person. The main disadvantages of `xzoom` are that it can't magnify under itself, that some of the key controls aren't compatible with `fvwm`, the normal Linux window manager and that it's default configuration doesn't run over a network (this can be fixed at some expense to speed). Apart from that though, it's excellent. It does continuous magnification which allows you to, for example, scroll a document up and down, whilst keeping the section you are reading magnified. Alternatively, you can move a little box around the screen, magnifying the contents and letting you search for the area you want to see. `xzoom` is also available as an rpm from the normal RedHat sites, making it very easy to install for people using the rpm system (such as Redhat users).

```
Begin3
Title:          xzoom
Version:        0.1
Entered-date:   Mar 30 1996
```

```
Description:    xzoom can magnify (by integer value) rotate
                (by a multiple if 90 degrees) and mirror about
                the X or Y axes areas on X11 screen
                and display them in it's window.
Keywords:       X11 zoom magnify xmag
Author:         Itai Nahshon <nahshon@best.com>
Maintained-by:  Itai Nahshon <nahshon@best.com>
Primary-site:   sunsite.unc.edu
                probably in /pub/Linux/X11/xutils/xzoom-0.1.tgz
Platforms:      Linux+11. Support only for 8-bit depth.
                Tested only in Linux 1.3.* with the XSVGA 3.1.2
driver.
                                Needs the XSHM extension.
Copying-policy: Free
End
```

8.7 NFBtrans

`nfbtrans` is a multi−grade Braille translation program distributed by the National Federation for the Blind in the U.S.A. It is released for free in the hope that someone will improve it. Languages covered are USA English, UK English, Spanish, Russian, Esperanto, German, Biblical Hebrew and Biblical Greek, though others could be added just by writing a translation table. Also covered are some computer and math forms. I have managed to get it to compile under Linux, though, not having a Braille embosser available at the present moment I have not been able to test it.

NFBtrans is available from ftp://nfb.org/ftp/nfb/braille/nfbtrans/. After downloading it, you will have to compile it.

Compiling NFBtrans on Linux

I have returned this patch to the maintainer of NFBtrans and he says that he has included it, so if you get a version later than 740, you probably won't have to do anything special. Just follow the instructions included in the package.

```
unzip −L NFBTR740.ZIP   #or whatever filename you have
mv makefile Makefile
```

Next save the following to a file (e.g. `patch−file`)

```
*** nfbpatch.c.orig     Tue Mar 12 11:37:28 1996
--- nfbpatch.c  Tue Mar 12 11:37:06 1996
***************
*** 185,190 ****
--- 185,193 ----
    return (finfo.st_size);
```

```
  }                /* filelength */

+ #ifndef linux
+ /* pretty safe to assume all linux has usleep I think ?? this should be
+ done properly anyway */
  #ifdef SYSVR4
  void usleep(usec)
    int usec;
***************
*** 195,200 ****
--- 198,204 ----
UKP  }                /* usleep */

  #endif
+ #endif

  void beep(count)
    int count;
```

and run

```
patch < patch-file
```

then type

```
make
```

and the program should compile.

8.8 UnWindows

UnWindows is a package of access utilities for X which provides many useful facilities for the visually impaired (not blind). It includes a screen magnifier and other customised utilities to help locate the pointer. UnWindows can be downloaded from ftp://ftp.cs.rpi.edu/pub/unwindows.

As it comes by default, the package will not work on Linux because it relies on special features of Suns. However, some of the utilities do work and I have managed to port most of the rest so this package may be interesting to some people. My port will either be incorporated back into the original or will be available in the BLINUX archives (see WWW references). The remaining utility which doesn't yet work is the configuration utility.

In my version the programs, instead of generating sounds themselves, just call another program. The other program could for example be

```
play /usr/lib/games/xboing/sounds/ouch.au
```

Which would make the `xboing` ouch noise, for example it could do this as the pointer hit the left edge of the screen.

dynamag

`dynamag` is a screen magnification program. please see the section on Screen magnification (magnification). This program worked in the default distribution.

coloreyes

`coloreyes` makes it easy to find the pointer (mouse) location. It consists of a pair of eyes which always look in the direction of the pointer (like xeyes) and change color depending on how far away the mouse is (unlike xeyes). This doesn't work in the default distribution, but the test version, at the same location, seems to work.

border

`border` is a program which detects when the pointer (mouse) has moved to the edge of the screen and makes a sound according to which edge of the screen has been approached. The version which is available uses a SUN specific sound system. I have now changed this so that instead of that, it just runs a command, which could be any Linux sound program.

un−twm

The window manager is a special program which controls the location of all of the other windows (programs) displayed on the X screen. `un−twm` is a special version which will make a sound as the pointer enters different windows. The sound will depend on what window has been entered. The distributed version doesn't work on linux because, like `border` it relies on SUN audio facilities. Again I already have a special version which will be avaliable by the time you read this.

9.Hardware

9.1 Braille terminals driven from Screen Memory

These are Braille terminals that can read the screen memory directly in a normal text mode. It is possible to use it to work with Linux for almost all of the things that a seeing user can do on the console, including installation. However, it has a problem with the scrolling of the normal Linux kernel, so a kernel patch needs to be applied. See Patching the Kernel for Braillex and Brailloterm.

Braillex

The Braillex is a terminal which is designed to read directly from the Screen memory, thus getting round any problems with MS−DOS programs which don't behave strangely. If you could see it on screen, then this terminal should be able to display it in Braille. In Linux, unfortunately, screen handling is done differently from MS−DOS, so this has to be changed somewhat.

To get this terminal to work, you have to apply the patch given below in section Patching the Kernel. Once this is done, the Braillex becomes one of the most convenient ways to use Linux as it allows all of the information normally available to a seeing person to be read. Other terminals don't start working until the operating system has completely booted.

The Braillex is available with two arrangements of Braille cells (80x1 or 40x2) and there is a model, called the IB 2−D which also has a vertical bar to show information about all of the lines of the screen (using 4 programmable dots per screen line)

```
Price: 8,995  (pounds sterling) or 11495 UKP for 2-D
Manufacturer: Alphavision Limited (UK)
Suppliers: ????
```

Brailloterm

``What is Brailloterm?

It's a refreshable display Braille, made by KTS Kommunikations−Technik Stolper GmbH. It has 80 Braille cells in an unique line. Each cell has 8 dots that are combined (up/down) to represent a character. By default, Brailloterm shows me the line in which the screen cursor is. I can use some functions in Brailloterm to see any line in the screen.'' – *Jose Vilmar Estacio de Souza*<jvilmar@embratel.net.br>

Jose then goes on to say that the terminal can also use the serial port under DOS but that it needs a special program. I don't know if any of the ones for Linux would work.

As with Braillex, this needs a special patch to the kernel work properly. See section Patching the Kernel.

```
Price: about 23.000,- DM /  $ 15.000,
Manufacturer: Kommunikations-Technik Stolper GmbH
Suppliers: ????
```

Patching the Kernel for Braillex and Brailloterm

This probably also applies to any other terminals which read directly from screen memory to work under MS−DOS. Mail me to confirm any terminals that you find work. This does not apply and will actually lose some features for terminals driven using the BRLTTY software.

I am told this patch applies to all Kernels version 1.2.X. It should also work on all Kernel versions from 1.1.X to 1.3.72, with just a warning from patch (I've tested that the patch applies to 1.3.68 at least). **From 1.3.75 the patch is no longer needed** because the Kernel can be configured not to scroll using `` `linux no−scroll' `` at the LILO prompt. See the Boot Prompt HOWTO for more details.

```
*** drivers/char/console.c~     Fri Mar 17 07:31:40 1995
--- drivers/char/console.c      Tue Mar  5 04:34:47 1996
***************
*** 601,605 ****
  static void scrup(int currcons, unsigned int t, unsigned int b)
  {
!       int hardscroll = 1;

        if (b > video_num_lines || t >= b)
--- 601,605 ----
  static void scrup(int currcons, unsigned int t, unsigned int b)
  {
!       int hardscroll = 0;

        if (b > video_num_lines || t >= b)
```

To apply it:

1. Save the above text to a file (say patch−file)
2. change to the drivers/char directory of your kernel sources
3. run

```
patch < patch-file
```

4. Compile your kernel as normal

Apply those patches and you should be able to use the Braille terminal as normal to read the Linux Console.

Put in words, the patch just means `change the 1 to a 0 in the first line of the function `scrup` which should be near line 603 in the file drivers/char/console.c'. The main thing about `patch` is that program understands this, and that it knows how to guess what to do when the Linux developers change things in that file.

If you want to use a more modern kernel with completely disabled scrolling, (instead of the boot prompt solution I already mentioned), please use the following patch. **This does not apply to kernels earlier than 1.3.75**.

```
*** console.c~  Fri Mar 15 04:01:45 1996
--- console.c   Thu Apr  4 13:29:48 1996
***************
*** 516,520 ****
  unsigned char has_wrapped;          /* all of videomem is data of fg_console */
  static unsigned char hardscroll_enabled;
! static unsigned char hardscroll_disabled_by_init = 0;

  void no_scroll(char *str, int *ints)
--- 516,520 ----
  unsigned char has_wrapped;          /* all of videomem is data of fg_console */
  static unsigned char hardscroll_enabled;
! static unsigned char hardscroll_disabled_by_init = 1;

  void no_scroll(char *str, int *ints)
```

9.2 Software Driven Braille Terminals

The principle of operation of these terminal is very close to that of a CRT terminal such as the vt100. They connect to the serial port and the computer has to run a program which sends them output. At present there are two known programs for Linux. `BRLTTY`, see section BRLTTY) and Braille enhanced screen.

Tieman B.V.

CombiBraille

This Braille terminal is supported by the `BRLTTY` software. It comes in three versions with 25, 45 or 85 Braille cells. The extra five cells over a standard display are used for status information.

```
Price: around 4600 UKP for the 45 cell model ...
Manufacturer: Tieman B.V.
Suppliers: Concept Systems, Nottingham, England (voice +44 115 925 5988)
```

Alva B.V.

The ABT3xx series is supported in `BRLTTY`. Only the ABT340 has been confirmed to work at this time. Please pass back information to the `BRLTTY` authors on other models.

```
Price: 20 cell - 2200 UKP; 40 cell 4500 UKP; 80 cell 8000 UKP
Manufacturer: Alva
Suppliers: Professional Vision Services LTD, Hertshire, England
           (+44 1462 677331)
```

Telesensory Systems Inc. displays

Because they have provided programming information to the developers, the Telesensory displays are supported both by `BRLTTY` and `screen`.

Powerbraille

There are three models the 40, the 65 and the 80. Only the 40 is known to be supported by `BRLTTY`.

```
Price: 20 cell - 2200 UKP; 40 cell 4500 UKP; 80 cell 8000 UKP
Manufacturer: Alva
Suppliers: Professional Vision Services LTD, Hertshire, England
           (+44 1462 677331)
```

Navigator

Again there are three models the 20, the 60 and the 80. Recent versions are all known to work with `BRLTTY` but whether earlier ones (with earlier firmware) also work has not been confirmed.

```
Price: 80 cell 7800 UKP
Manufacturer: Alva
Suppliers: Professional Vision Services LTD, Hertshire, England
           (+44 1462 677331)
```

Braille Lite

This is more a portable computer than a terminal. It could, however, be used with BRLTTY version 0.22 (but not newer versions) as if it was a normal Braille terminal. Unfortunately, many of the features available with the CombiBraille cannot be used with the Braille Lite. This means that it should be avoided for Linux use where possible.

```
Price: $3,395.00
Manufacturer: Blazie Engineering
```

9.3 Speech Synthesisers

Speech synthesisers normally connect to the serial port of a PC. Useful features include

- Braille labels on parts
- Many voices to allow different parts of document to be spoken differently
- Use with headphones (not available on all models)

The critical problem is that the quality of the speech. This is much more important to someone who is using the speech synthesiser as their main source of information than to someone who is just getting neat sounds out of a game. For this reason T.V. Raman seems to only recommend the DECTalk. Acceptable alternatives would be good.

DECTalk Express

This is a hardware speech synthesiser. It is recommended for use with Emacspeak and in fact the DECTalk range are the only speech synthesisers which work with that package at present. This synthesiser has every useful feature that I know about. The only disadvantage that I know of at present is price.

```
Price: $1195.00
Manufacturer: Digital Equipment Corporation

Suppliers: Many.  I'd like details of those with Specific Linux
        support / delivering international or otherwise of note only
        please.  Otherwise refer to local organisations.
        Digital themselves or the Emacspeak WWW pages.
```

Accent SA

This is a synthesiser made by Aicom Corporation. An effort has begun to write a driver for it however help is needed. Please see http://www.cyberspc.mb.ca/~astrope/speak.html if you think you can help.

SPO256−AL2 Speak and Spell chip.

Some interest has been expressed in using this chip in self built talking circuits. I'd be interested to know if anyone has found this useful. A software package `speak-0.2pl1.tar.gz` was produced by David Sugar <dyfet@tycho.com>. My suspicion, though, is that the quality of the output wouldn't be good enough for regular use.

10.Acknowledgements

Much of this document was created from various information sources on the Internet, many found from Yahoo and DEC's Alta Vista Search engine. Included in this was the documentation of most of the software packages mentioned in the text. Some information was also gleaned from the Royal National Institute for the Blind's helpsheets.

T.V. Raman, the author of Emacspeak has reliably contributed comments, information and text as well as putting me in touch with other people who he knew on the Internet.

Kenneth Albanowski <kjahds@kjahds.com> provided the patch needed for the Brailloterm and information about it.

Roland Dyroff of S.u.S.E. GmbH (Linux distributors and makers of S.u.S.E. Linux (English/German)) looked up KTS Stolper GmbH at my request and got some hardware details and information on the Brailloterm.

The most major and careful checks over of this document were done by James Bowden, <jrbowden@bcs.org.uk> and Nikhil Nair <nn201@cus.cam.ac.uk>, the `BRLTTY` authors who suggested a large number of corrections as well as extra information for some topics.

The contributors to the blinux and linux−access mailing lists have contributed to this document by providng information for me to read.

Mark E. Novak of the Trace R&D centre http://trace.wisc.edu/ pointed me in the direction of several packages of software and information which I had not seen before. He also made some comments on the structure of the document which I have partially taken into account and should probably do more about.

Other contributors include Nicolas Pitrie and Stephane Doyon.

A number of other people have contributed comments and information. Specific contributions are acknowledged within the document.

This version was specifically produced for RedHat's Dr. Linux book. This is because they provided warning of it's impending release to myself and other LDP authors. Their doing this is strongly appreciated since wrong or old information sits around much longer in a book than on the Internet.

No doubt you made a contribution and I haven't mentioned it. Don't worry, it was an accident. I'm sorry. Just tell me and I will add you to the next version.
