
Arabic HOWTO

Mohammed Elzubeir, Arabeyes Project <contact (at) arabeyes dot org>

\$Revision: 1.3 \$

This document is meant to serve as a starting point for anyone looking into adding Arabic support (from Alef to Yeh) to their Linux/Unix environment.

Table of Contents

License	1
Introduction	1
Acknowledgement	2
Translations	2
Feedback	2
Setup Essentials	2
Configure Kernel	2
Set Locales	2
Install Libraries	3
Patch 'less-378'	3
Use Arabic Filenames	4
Configure Console for Arabic	5
Setup Akka (deprecated)	5
Configure X Windows for Arabic	6
Install Fonts	7
Setup X Terminals	9
Configure Keymaps	10
Print in Arabic	11
Configure Applications for Arabic	11
Setup Editors	11
Setup Mail Clients	13
Setup Word Processors	13
Configure Desktop Environments for Arabic	14
Setup FVWM	14
Setup Gnome	15
Setup KDE	16

License

Copyright (c) 2002, Arabeyes Project, Mohammed Elzubeir.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Introduction

Arabic support has been non-existent in Linux until recently. What this means for you as the user is that enabling Arabic is not straight-forward or out of the box. What it also means is that this document may at some point be outdated as things are changing at a very fast rate. It is the intentions of the author to keep this HOWTO up-to-date at all times, but this does not guarantee it.

It is advised that you follow this document from the beginning till the end. This is mainly because it is organized as building blocks. This is particularly true to the fonts.

Acknowledgement

One must give credit to those who have made this document possible. Many thanks go to Nadim Shaikli for the wealth of information he has posted throughout the months. Thanks also go to Isam Bayazidi for his Gnome and KDE instructions, as well as Mohammed Sameer for his reading Arabic filenames information.

There are also countless people who have contributed indirectly to this HOWTO by posts they have made on the various mailing-lists who are too numerous to list here.

Translations

Mohammed Sameer has volunteered to translate this document to Arabic (if no one else sticks their neck out).

Feedback

The sources for this document vary from different mailing-lists, personal experiences and feedback from others on what needs to be addressed.

If you have any questions or suggestion, please do make them known on the 'doc' mailing list here: <http://lists.arabeyes.org/mailman/listinfo/doc> [http://lists.arabeyes.org/mailman/listinfo/doc]

Setup Essentials

Before we go any further, let us first make sure we have all the very essentials for Arabic support. The remainder of this document will assume you have read and followed the instructions of the following sections.

Configure Kernel

We will not get into the details of compiling your kernel, as it is not within the scope of this document. You can either find out if your pre-compiled kernel is compiled with the following options (check with your distribution's documentation) or compile your own kernel.

```
# Partition Types
CONFIG_NLS=y
# Native Language Support
CONFIG_NLS_DEFAULT="UTF8"
CONFIG_NLS_CODEPAGE_864=y
CONFIG_NLS_ISO8859_6=y
CONFIG_NLS_UTF8=y
```

Set Locales

NOTE: This section is incomplete and may contain errors. Please do NOT use it just yet.

There are several environments that need to be defined in order for certain applications to function the way you expect them to, with regard to Arabic.

```
$ LC_CTYPE=ar_EG.UTF-8
$ CHARSET=ISO_8859-6
$ OUTPUT_CHARSET=UTF-8
$ LESSCHARSET='UTF-8'
$ LANG=ar_US
$ export LC_CTYPE CHARSET OUTPUT_CHARSET LESSCHARSET LANG
```

Please note that you can change the 'US' for the country code of wherever you are. So, if you are in Egypt you would put in 'ar_EG' (this applies to all of the above).

There are also other locale environment variables like LANGUAGE and LC_ALL. LC_ALL overrides all other LC_* variables. You can simply set it to 'UTF-8'.

```
$ export LC_ALL=ar_US.UTF-81
```

Install Libraries

There are three main element in bringing Arabic support:

- *Bidirectional support*: the ability for text to render in both directions (left-to-right) and (right-to-left) when intermixing an RTL language (e.g. Arabic) with a LTR language (e.g. English).
- *Shaping/Joining support*: the ability to shape Arabic script accordingly (up to four possible shapes per letter depending on position in word).
- *UTF-8 support*: the ability to support the Unicode Transformational Format.

Install FriBiDi

<http://fribidi.sourceforge.net/> [<http://fribidi.sourceforge.net/>]

Perhaps the most popular and most important library to have in your arsenal. This library currently supports re-ordering in compliance with the Unicode TR#9 [<http://www.unicode.org/unicode/reports/tr9/>]. Applications such as mlterm and Pango either use it or parts of it.

Unfortunately as of the writing of this document, fribidi is yet to have *shaping* as a part of the entire library.

Patch 'less-378'

<http://www.arabeyes.org/project.php?proj=patches> [<http://www.arabeyes.org/project.php?proj=patches>]

A patch was submitted to the author of less to incorporate it onto the main source code. In the meantime,

¹This is not recommended as it may have some unpredictable effects.

this patch will fix the size an Arabic line takes on your terminal.

Download the patch from here: http://www.arabeyes.org/download/download/external/less/less_composing.patch.tgz
[http://www.arabeyes.org/download/download/external/less/less_composing.patch.tgz].

And the source for 'less' from here: <http://www.greenwoodsoftware.com/less/less-378.tar.gz>
[<http://www.greenwoodsoftware.com/less/less-378.tar.gz>]

```
$ tar zxvf less-378.tar.gz
$ cd less-378
$ tar zxvf less_composing.patch.tgz
$ patch -b -p0 < less_composing.patch
$ ./configure
$ make && make install
```

Now you should have a fully functional 'less' which displays Arabic text with the proper screen width!

Use Arabic Filenames

Although the capability to have Arabic filenames are there, it is generally not advised. That is because currently, most applications will not know how to deal with it.

Read Arabic Filenames

In order for your file manager to read Arabic filenames properly, you need to specify the character set to be used. There are two environment settings you need to have, which you can either export (using bash) or add to your ~/.profile or ~/.bash_profile file.

```
$ export G_BROKEN_FILENAMES=1 2
```

Write Arabic Filenames

There are two possibly means (at least) to name files using Arabic characters. The first is via an application's GUI filemanager (e.g. dired within emacs). The second is via the more common ubiquitous command-line in cooperation with a UTF-8 enabled shell. The command-line method, of course, will have to be used in conjunction with an Arabic-enabled Xterminal or terminal emulator (such as mlterm or PuTTY). The two shells that have been tested and used extensively are bash and tcsh.

bash version 3.0+ requires no special setup or instructions as it works flawlessly with Arabic UTF-8 filename. Simply compile, install and use.

tcsh is somewhat more picky and its Arabic (and UTF-8) support depends on its compile options and environment. In order to find out what options tcsh was compiled we'll need to probe its special 'version' variable while under its shell.

```
$ tcsh
$ set | grep version
```

2

Glib assumes that the filenames are in the locale encoding rather than in UTF-8

If 'wide' is listed as part of the options, then UTF-8 support is available and Arabic filenames are possible pending the use of a UTF-8 locale (e.g. ar_IQ.UTF-8). If 'dspm' is listed, then the following special variable setting is required.

```
$ set dspmbyte=utf8
```

If neither option is listed, tcsh is unusable for this function and bash should be seriously considered instead.

Mount Windows Partition

In order for you to be able to read Arabic filenames from a Windows partition, you need to tell the **mount** command what character set to use. This is done by the following (assuming your Windows partition resides on /dev/hda3:

```
# mount -t auto /dev/hda3 /mnt/win/ -oiocharset=utf8 3
```

You can also make this permanent by adding it to your /etc/fstab file.

```
/dev/hda3 /mnt/win vfat defaults,iocharset=utf8 0 0
```

Configure Console for Arabic

The console is the land where GUI is non-existent. Fortunately, we can have Arabic under the Linux console with the use of some utilities.

Setup Akka (deprecated)

<http://www.arabeyes.org/project.php?proj=akka> [<http://www.arabeyes.org/project.php?proj=akka>]

Akka is NO longer used and/or maintained. Please look into using BiCon. This section needs to be removed and updated with BiCon's info.

<http://www.arabeyes.org/project.php?proj=bicon> [<http://www.arabeyes.org/project.php?proj=bicon>]

Akka intercepts all input and output to and from the terminal to give the user the ability to read Arabic text. This means that any application that can support the Arabic character set (or UTF-8) can and should be able to work under Akka.

Akka has several dependencies that must be first satisfied:

3

Please note that using the 'oiocharset' option is reported to cause inconsistencies. Use at your own risk!

- fribidi => 0.10 (<http://fribidi.sourceforge.net> [<http://fribidi.sourceforge.net>])
- glib (<ftp://ftp.gtk.org/pub/gtk> [<ftp://ftp.gtk.org/pub/gtk>])
- loadkeys
- orbit (<http://www.labs.redhat.com/orbit/> [<http://www.labs.redhat.com/orbit/>])
- SWIG (<http://www.swig.org/> [<http://www.swig.org/>])

Assuming that you have all the above installed in your system, and you have downloaded the latest akka version. If you have a Debian system you can download the Debian package from the Akka homepage. Otherwise, you can start compiling:

```
$ ./configure4
$ make
# cp keymaps/us- latin1.map /etc/console-tools/
# cp keymaps/arabic.kmap /usr/share/keymaps/
# mkdir /usr/share/fonts/akka && mkdir /usr/share/fonts/akka/glyph
# cp fonts/* /usr/share/fonts/akka/
# cp glyph/* /usr/share/fonts/akka/glyph/
# cp conf/akka-conf.pl /usr/bin/
# cp src/Akka.pm /usr/lib/perl5/
# cp src/akka /usr/bin
```

Now you can simply run the akka daemon first, followed by the perl driver. The perl script is what tells the daemon what mode you want your terminal to be in (Latin, Arabic, shaped, squared, etc.)

```
# akka &
# akka-conf.pl
```

There are three keyboard sequences you need to know, to make the best out of Akka.

- **Shift-F10** - Insert mode (ltr,rtl -- cursor doesn't move)
- **Shift-F11** - Switch Language (e.g. Arabic/English)
- **Shift-F12** - Mirror Screen (ltr,rtl)

Configure X Windows for Arabic

XFree86 is an X Window implementation that is freely available and is one of the most popular. XFree86 is written under the assumption that only Latin-based languages will use it. This means that adapting the application to the new world of internationalization is additionally cumbersome and complex. It would be nice if we could scrap it all together and start from scratch, but we will work with what we

⁴Please note that if you have Akka version 1.0 or earlier, you will need to replace the 'configure' with 'chconfigure'.

have for now ;)

Install Fonts

<http://www.arabeyes.org/project.php?proj=khotot> [<http://www.arabeyes.org/project.php?proj=khotot>]

Unfortunately, XFree86 does not come with full Arabic fonts. In fact, the XFree86 repository does include a complete Arabic font but it is truncated during the installation for *memory optimization* reasons. This is only true to bitmap fonts. XFree86 does not have any complete Arabic TrueType fonts.

To find out what fonts you have installed in your system, do:

```
$ xlsfonts | more
```

Install Bitmap Fonts

You can download the complete font here: http://prdownloads.sourceforge.net/arabeyes/ae_fonts_mono.tar.bz2?download [http://prdownloads.sourceforge.net/arabeyes/ae_fonts_mono.tar.bz2?download]

To install it, simply copy it over to one of your fonts directories (e.g. /usr/X11R6/lib/X11/fonts/misc/):

```
# cd /usr/X11R6/lib/X11/fonts/misc
# mkfontdir && xset fp+ `pwd` && xset fp rehash
```

This should make the font available to you. To test if your XFree86 system can see the font, you can do:

```
$ xlsfonts | grep arabeyes
-misc-fixed-medium-r-normal--20-200-75-75-c-100-arabeyes-1
```

Install TrueType Fonts

In order for you to be able to use TTF (TrueType Fonts) you must have an X Font Server that supports TrueType fonts. As of XFree86 4.x, it can natively support TrueType fonts. There are two more popular servers: xfs and xfstt. The main difference between the two is that xfs also supports Type 1 fonts (which are Adobe's PostScript fonts).

You can get TTF's from the Fonts section here: <http://www.arabeyes.org/resources.php> [<http://www.arabeyes.org/resources.php>]

xfs

1. To check if you are running xfs do:

```
$ ps -waux | grep xfs
```

If it is not running, make sure you start it before restarting XFree86.

```
$ xfs -droppriv -daemon
```

2. You need to add the following line to your `/etc/X11/XF86Config-4` file (where you see 'FontPath' entries in the file):

```
FontPath  "unix/:7100"
```

3. Create the font directory by executing the following commands while inside the directory:

```
$ ttmkfdir -o fonts.scale ; mkfontdir
```

4. Check to see if your `xfs /etc/X11/fs/config` file knows about your new truetype fonts directory. They should be listed under the 'catalogue' line, with the font directories are separated by commas.

xfstt

The location where `xfstt` keeps truetype fonts apparently varies from one distribution to another. For example, it is found in `/usr/share/fonts/truetype/` in Debian. You must put your TTF's in that directory in order for it to know about your fonts.

You can update the list of loaded fonts by doing:

```
$ xfstt --sync
$ xfstt &
```

In order for XFree86 to know about `xfstt` you need to add the following line:

```
FontPath  "unix/:7101"5
```

Install Type 1 Fonts

Type 1 fonts are what ghostscript uses. This is especially important because some big applications (like StarOffice until recently) only supported Type 1 fonts and not TrueType fonts. It is also useful if you use TeX for your typesetting.

Installing Type 1 fonts is straight-forward. While in the directory containing your fonts:

5
Please note that the default port number will also vary from one distribution to another.

```
# type1inst
# cat Fontmap >> /PATH/TO/YOUR/SYSTEM-WIDE/Fontmap
```

The path to your system-wide Fontmap file also varies. For instance, it is `/usr/share/gs/6.53/Fontmap.GS` in my Debian distribution.

You can also convert TrueType fonts to Type 1 fonts (which is what you will probably need to do).

```
$ ttf2pt1 -b fontname.ttf fontname
```

You can get ttf2pt1 here: <http://quadrant.netspace.net.au/ttf2pt1/>
[<http://quadrant.netspace.net.au/ttf2pt1/>]

Configure Anti-aliasing

Anti-aliasing is what makes your fonts look so beautiful. It is almost as if someone took an eraser and smudged the letters with it so the angles are so smooth. This is all achieved via the Xft library, which is designed to interface the FreeType rasterizer with the X Rendering Extension. It is often called 'gdkxft' or similar. Check with your distribution.

To configure Xft to work for you, either check the system-wide configuration (`/etc/X11/XftConfig`) to find out where your personalized configuration (`~/.xftconfig`) should be. Put the following in it:

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
  <dir>/usr/X11R6/lib/X11/fonts/truetype</dir>
</fontconfig>
```

The majority of applications currently don't come with Xft enabled, even if the source code links to it the binary distributions are often left without Xft support. A good example of this is Mozilla. As of version 1.2 you would have to compile the source yourself with `--enable-xft` option.

Setup X Terminals

Setup mlterm

<http://mlterm.sourceforge.net/> [<http://mlterm.sourceforge.net/>]

Mlterm was the first X terminal to support Arabic and bidi in a satisfactory fashion. Most binary distributions have utf-8 and bidi support. If yours doesn't, you can download and compile mlterm like this:

```
$ ./configure --enable-fribidi && make && make install
```

Create a directory named `~/.mlterm` in your home-directory with two files: `font` and `main`. `font` should have:

```
ISO10646_UCS2_1 = 20, -arabeyes-fixed-medium-r-normal--20-200-75-75-c-100-misc-1;
```

and `main` should have the following:

```
ENCODING = utf8
fontsize = 20
```

Now once you start `mlterm`, you should be able to read Arabic under any text-based application that supports UTF-8. Also, note that less may not work unless you set the `LESSCHARSET` environment to 'UTF-8'.

Setup item

To be completed.

Configure Keymaps

There are two key programs you will need to know about:

1. `xmodmap`: modifies keymaps and pointer button mappings in X. This is the old deprecated way.
2. `setxkbmap`: sets the keyboard using the X Keyboard Extension. This is how the new way of setting the keyboard is done (as of XFree86 4.x)

As of XFree86 4.2.0 there is already a symbol map for the Arabic keyboard, which should be found in `$X11DIR/lib/X11/xkb/symbols/ar`. If this file is missing or you have an older version of XFree86 (especially 3.3.6) you can use this keymap: <http://www.arabeyes.org/download/download/3rd/arabic.xkb> [<http://www.arabeyes.org/download/download/3rd/arabic.xkb>] in conjunction with the `xmodmap` utility (read the manpages for more information).

However, if you do have XFree86 4.2 or higher, you can simply switch your keyboard by doing:

```
$ setxkbmap -symbols "us(pc101)+ar+group(ctrl_shift_toggle)"
```

Now you can simply switch between the Arabic keyboard layout and the English one by pressing the `Ctrl` and `Shift` buttons. Or you can simply add this to your `/etc/X11/XF86Config-4` file like so:

```
Section "InputDevice"
    Identifier "Keyboard0"
    Driver "keyboard"
    Option "XkbRules" "xfree86"
    Option "XkbModel" "pc101"
    Option "XkbLayout" "ar"
    Option "XkbOptions" "grp:ctrl_shift_toggle"
EndSection
```

Print in Arabic

Printing Arabic documents can be a tricky thing. To print plain text documents you can use `text-bdf2ps.pl`. Currently, the latest test version of the script includes the Arabic patch. It can be downloaded here: <http://oldrus-ispell.sourceforge.net/txtbdf2ps.html> [<http://oldrus-ispell.sourceforge.net/txtbdf2ps.html>]

```
$ txtbdf2ps.pl -UTF-8 \  
               -bidi \  
               -bdf=/PATH/PATH/PATH/10x21.bdf \  
               -text=arabic_file > output.ps  
$ lpr output.ps
```

You can also use TrueType fonts, by replacing the `-bdf` option with `-font` followed by the path to the TrueType font.

```
$ txtbdf2ps.pl -UTF-8 \  
               -bidi \  
               -font=/PATH/PATH/PATH/font.ttf \  
               -text=arabic_file > output.ps  
$ lpr output.ps
```

Configure Applications for Arabic

Setup Editors

Install VIM

<http://www.arabeyes.org/project.php?proj=vim> [<http://www.arabeyes.org/project.php?proj=vim>]

VIM supports Arabic as of version 6.2 out of the box. However, if you want to run an older version (for your own reasons), then there is the patch which has been submitted to VIM's author that is incorporated into 6.2 release. Although the VIM Arabic patch does not yet support bidirectionality, using it under `ml-term` will provide such support.

Download VIM-6.1 source

Go to: <http://vim.sourceforge.net/download.php> [<http://vim.sourceforge.net/download.php>] and click on the 'unix' section, download the following files:

```
vim-6.1.tar.bz2  
vim-6.1-lang.tar.gz
```

6

The `\`'s are there to tell the shell prompt that we will continue our command in the next line (essentially to ignore the newline). You can simply omit them and continue typing the command in one line. It is only here for readability.

Uncompress files:

```
$ tar jxvf vim-6.1.tar.bz2
$ tar zxvf vim-6.1-lang.tar.gz
```

Download the Arabic Shaping patch

You can download the patch here: http://prdownloads.sourceforge.net/arabeyes/arabic_shape61.tar.gz?download
[http://prdownloads.sourceforge.net/arabeyes/arabic_shape61.tar.gz?download]

```
$ cd YOUR_VIM_ROOT_DIR
$ patch -b -p0 < arabic_shape61.patch
```

Compile VIM

```
$ configure --enable-multibyte --with-features=big
$ make && make install
```

Start VIM

You can either start VIM in console mode or graphical mode. While the graphical mode has a nicer interface (to some people), it does not support bidirectional text. Running vim under mlterm will automatically give you this support.

```
$ ./vim -g
```

Enable Arabic within vim

```
:set guifont=-misc-fixed-medium-r-normal--20-200-75-75-c
:set encoding=utf-8
:set keymap=arabic
:set arabic
```

Note you can include the above 4 commands as-is in your `~/.vimrc` file

Some important VIM commands

```
:set norightleft
:set rightleft
:help
:q!
```

Use **Ctrl-^** to switch between Arabic and English.

Install Emacs-Bidi

<http://www.m17n.org/emacs-bidi/index.html> [<http://www.m17n.org/emacs-bidi/index.html>]

The instructions to compile and install 'emacs-bidi' are all available on the above referenced website. After downloading the emacs-bidi.tar.gz package, you do:

```
$ tar zxvf emacs-bidi.tar.gz
$ cd emacs-bidi
$ ./configure
$ make && make install
```

You can skip the fonts part since it is the same font file that is referenced above in the 'Bitmap Fonts'. However, Emacs may not be able to automatically recognize the font in the "Font Menu". Instead you can place this in your ~/.Xdefaults

```
Emacs*font: -m17n-mule-medium-r-normal--20-140-100-100-p-90-iso10646-1
#Emacs*font: -microsoft-tahoma-medium-r-normal--0-0-0-0-p-0-iso8859-6
```

You can uncomment the second line in the above example and comment the first one, if you want to use the MS Tahoma font instead. This is simply to demonstrate that you can use any font available to your system. However, I wouldn't recommend it since it slows Emacs-Bidi considerably.

Setup Mail Clients

Install mutt

<http://www.mutt.org/>

'mutt' supports UTF-8 properly as of version 1.4. If the binary distribution you got does not seem to work like you expect it to, grab the source and compile with the following:

```
$ ./configure --enable-locales-fix --without-wc-funcs 7
$ make && make install
```

Also, note that mutt does not have bidi support, which means that it is best used under an x terminal such as mlterm or akka (for the console).

If you are unable to read certain messages, it will probably be because the person who sent you the email is using a non-standard encoding or message header. Unfortunately, those cases are very common.

Setup Word Processors

Setup LyX

Although LyX is not your typical word processor, it is classified as WYSIWYM (What You See Is What

⁷--enable-locales-fix' and '--without-wc-funcs' are NOT required if you are under Linux. These solve problems for non-Linux systems only.

You Mean). There are two main packages you will need to have in order to function with Arabic. Namely LyX and ArabTeX.

Whether ArabTeX is bundled with teTeX packages may vary and depend on your distribution. For instance, Mandrake calls it 'tetex-latex-arab' whereas Debian simply calls it 'arabtex'. Check with your distribution.

Once you have both installed, create ~/.lyxrc with the following contents:

```
\rtl true
\kbmap true
\kbmap_primary null
\kbmap_secondary arabic

\bind "F12" "language Arabic"

\language_auto_begin false
\language_auto_end false
\language_command_begin "\begin{arabtext}"
\language_command_end "\end{arabtext}"
\language_package "\usepackage{arabtex, iso88596}\setcode{iso8859-6}"

\screen_font_encoding iso8859-6
\screen_font_encoding_menu iso8859-1
\screen_font_roman "-*-tahoma" 8
```

Note that you can switch between English and Arabic using the **F12** key.

Install StarOffice/OpenOffice

To be completed.

Install AbiWord

To be completed.

Configure Desktop Environments for Arabic

Setup FVWM

<http://www.fvwm.org/> [<http://www.fvwm.org/>]

Recently (Dec. 18, 2002), a patch was added to the FVWM CVS repository, allowing for full Arabic support. This means that FVWM supports both bidi and Arabic shaping. Since this is yet to be in a release, you will have to download the source code from cvs and compile it yourself.

Download From CVS

8

The fontname is taken from the full name of the font (e.g. `-microsoft-tahoma-medium-r-normal--0-0-0-0-p-0-iso8859-6`).

In order to login to cvs and download the source code, you need to do the following:

```
$ cvs -d:pserver:anonymous@cvs.fvwm.org:/home/cvs/fvwm login
(Enter Password: guest)
$ cvs -d:pserver:anonymous@cvs.fvwm.org:/home/cvs/fvwm co fvwm
```

Compile and Install FVWM

This will download the source code to a subdirectory `fvwm/`.

```
$ cd fvwm
$ utils/configure_dev.sh
$ ./configure
$ make && make install
```

Now you have your new FVWM installed. Run it and make sure everything is working. This can be done by including it in your `~/.xinitrc`. For example:

```
exec fvwm
```

Then you can run `xinit`

```
$ xinit &
```

Configure FVWM

Now that we have FVWM running we need to configure it to do what we want. In order for FVWM to handle Arabic text, you will need to tell it what fonts to use. This can be done by modifying your `~/.fvwm/.fvwm2rc` to include the following:

```
Style * Font -misc-fixed-medium-r-normal--20-200-75-75-c-100-arabeyes-1/iso10646-1
Style * IconFont -misc-fixed-medium-r-normal--20-200-75-75-c-100-arabeyes-1/iso106
```

Setup Gnome

<http://www.arabeyes.org/project.php?proj=gnome-i18n>
[<http://www.arabeyes.org/project.php?proj=gnome-i18n>]

Localize Interface

To be completed.

Set Arabic Keyboard Layout

Gnome 2.0.2 and later will give you the proper keyboard based on your locale settings. See setlocales for more information.

Setup KDE

<http://www.arabeyes.org/project.php?proj=kde-i18n>
[<http://www.arabeyes.org/project.php?proj=kde-i18n>]

Localize Interface

<http://i18n.kde.org/teams/index.php?action=info&team=ar>
[<http://i18n.kde.org/teams/index.php?action=info&team=ar>]

Download the latest Arabic translation from the above link and then:

```
$ tar jxvf kde-i18n-ar.tar.bz2
$ cd kde-i18n-ar
$ ./configure
$ make install
```

KDE's default fonts are not full Unicode fonts. In other words, they do not include Arabic. Now that we have the actual Arabic interface translations installed, we need to make sure that the default KDE font is set to a Unicode font or one that at least is a full Arabic font. If you are using one of the MS fonts Arial (`arialuni.ttf`) and Courier New (`cour.ttf`) are both full Unicode fonts you can use.

If you are using KDE < 3.1 then: Control Center->Look & Feel->Fonts. If you are using a later version then: Control Center->Appearance and Theme->Fonts

So far all we have done is prepare everything for a full Arabic interface. If you are not interested in a complete Arabic interface you can skip this and move on to `kdearabickeyboard`.

To configure the interface, if you are using KDE < 3.1 then: Control Center. If you have a later version of KDE then: Control Center->Regional & Accessibility->Country/Region & Language->Add Language->Choose Arabic

Set Arabic Keyboard Layout

To be completed.

Force Arabic Fonts on Konqueror

Some sites use stylesheets that ask for fonts that you do not have. Konqueror often replaces it with a font that does not support Arabic, so you end up with *squared*.

To force your Style Sheet in Konqueror you can:

1. Settings->Configure Konqueror->Stylesheets
2. Under the *General Tab* and select *Customize*.
3. Choose a font under *Base Family*, then check the "User same family for all text" box.

⁹You can add a '--prefix' option to tell it to install it relative to where your other KDE files are

4. Press *OK* and re-start Konqueror.

Please do note that doing this will make all sites use the one font you have selected (and may not look very pretty on every page). Others have reportedly changed fontnames and aliases to fool the Browser into thinking certain fonts do exist in the system.