

K-TOOLS: Kdict NOW TRANSLATE

If you don't feel like poring over a paper dictionary, don't worry. As Stefanie Teufel explains, with Kdict at hand, an online alternative may not be far off

However good your knowledge of English, French or jargon may be, you're still likely to forget the meaning of a word every now and then. For cases like these, there's Kdict – the dictionary for KDE. Formerly available as an individual application, from version 2.2 Kdict is now a permanent part of the kdenetwork packet so you don't even have to install it separately.

This virtual translation aid uses the DICT protocol (<http://www.dict.org>) in order to search for definitions in databases on diverse Internet servers. Anyone who has previously taken a look at the program will be in for a surprise because in addition to porting onto version 2.2 of KDE, the developers have also installed features in the new edition that make working with this tool even better. The opportunity to embed Kdict in the panel is particularly useful.

A reference work for you alone

If the words "on diverse Internet servers" have already made you think anxiously about your phone bill, don't worry: you don't need to be permanently online. With the DICT daemon it's possible to run a

Webster Webster's Revised Unabridged Dictionary, 1913 Edition is a purely English-language dictionary with refined search options (wildcards, etc.), which now encompasses over 100,000 words.

WordNet Work began on WordNet in 1985 and is still going on today. Those involved in its development are mainly psychologists and linguists from the University of Princeton, USA. The structure of this lexicon is oriented towards psycholinguistic theories about human memory, unlike the more usual dictionaries, which are organised on the basis of alphabetical order or in classes of synonyms. Therefore, words are shown with the distinction between the form of the word and the meaning of the word. The meaning of the word is thereby represented by a number of synonymous word forms. WordNet is not limited to any specific field and currently comprises over 95,600 English word forms. This thesaurus is especially recommended as an aid to writing English texts.

Jargon File Eric Raymond's The New Hacker's Dictionary (which is the official name) is the best reference work for the latest Internet and hacker slang. It lists and defines – in often ironic fashion – terms to do with the Net and the computer.

K-tools

In this column we present tools, month by month, which have proven to be especially useful when working under KDE, solve a problem which otherwise is deliberately ignored, or are just some of the nicer things in life, which – once discovered – you wouldn't want to do without.

DICT server on your own computer. As well as providing faster access speeds and cost saving (for those without a true flat rate Internet service, at least), this kind of local server also offers the option to install extra databases.

Before you start up your own virtual dictionary, you'll first need to install the dictd server and the associated utility programs. The simplest way to do this is to use the deb or rpm packets appropriate for your distribution. Both Debian and Mandrake contain the appropriate packets, and you can easily track down those for other rpm-based distributions at <http://rpmfind.net>.

Alternatively, you can also compile dictd yourself. To do this you'll need the packet dictd-1.5.5.tar.gz, which you can download from <ftp://ftp.dict.org/pub/dict>. Unpack this and install it with our old friends `.configure`, `make` and `make install`.

We now ought to download a couple more dictionary databases, without which the server makes little sense. The usual suspects such as **Webster** (dict-web1913), **WordNet** (dict-wn) or the **Jargon File** (jargon) can be found at <http://www.dict.org/links.html>. It's also worth a visit to <http://www.freedict.de/>, which provides you with bilingual glossaries in more than 15 languages.

Although <http://www.dict.org> hosts many other databases, most of these are in a format which dictd cannot deal with. Every DICTD database must consist of two files: `.index` contains the index and `.dict` the actual data or words. Unpack the database archive into a directory which you have created specifically for this purpose, such as `/usr/local/db`. If the data is

present, the local DICT access should be configured. If, in addition to Kdict, you would still like to use the command line program dict to hunt through your local databases, the file dict.conf should contain the following simple but effective line:

```
server localhost
```

We will now use the file dictd.conf to configure the local server. Firstly, use the access command to define who will have access to this in the future. The best way to do this is to decide something like

```
access {allow localhost deny *}
```

This means that you are only permitting local access.

All that you need to do now is specify which databases you would like dictd to use. There were the files we carefully hunted out and placed in the `/usr/local/db` directory. Our dictd.conf looks as follows:

```
database Latin { data "/usr/local/db/2
lat-eng.dict.dz"
                index "/usr/local/2
db/lat-eng.index" }
database English{ data "/usr/local/db/2
eng-lat.dict.dz"
                 index "/usr/local/2
db/eng-lat.index" }
database Jargon { data2
"/usr/local/db/jargon.dict.dz"
                 index "/usr/local/db/2
jargon.index" }
```

Should you wish to use additional databases (such as Webster), you can expand dictd.conf at any time by adding a command for your new database. Our example would look something like this:

```
database web1913 { data "/usr/local/db/2
web1913.dict.dz"
                 index "/usr/local/db/2
web1913.index" }
```

Now start your dictionary server by entering a simple

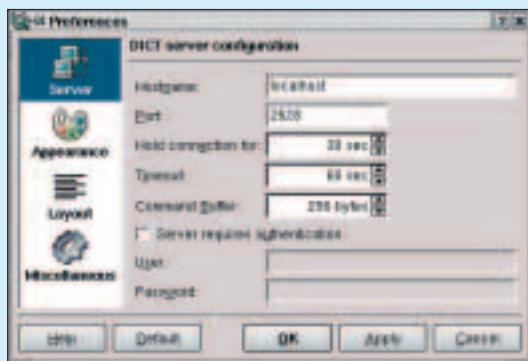


Figure 2: This server is known as the local host

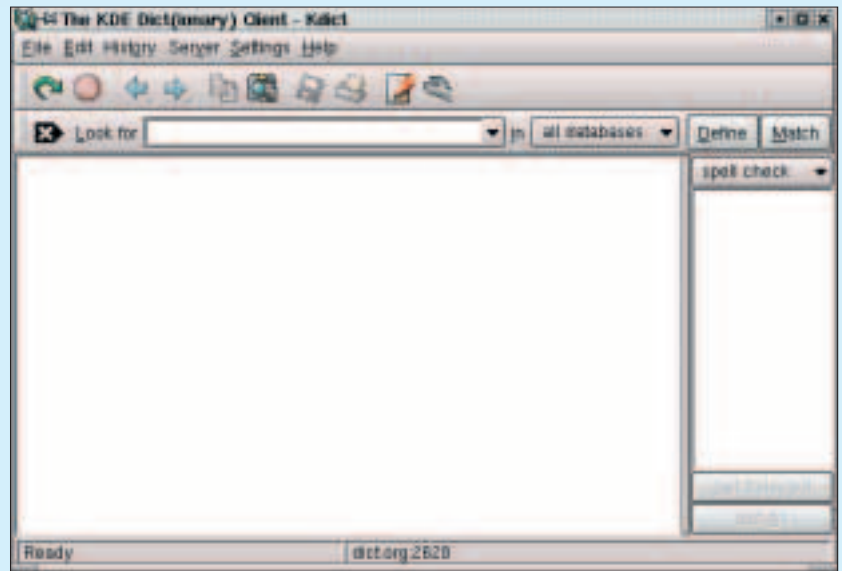


Figure 1: Welcome

`dictd` command in any terminal. Using this, access with `dict`, or in our case `Kdict`, ought to work.

Verbose

After all this installing and configuring, you must by now be dying to finally see Kdict in action. Nothing could be easier; simply start your dictionary with the `kdict &` command in any terminal emulation or alternatively via `K-Menu/Utilities/Kdict`. Don't be too shocked by the rather empty window (Figure 1) that will then confront you – this will fill up faster than you think.

Before you start assiduously looking up missing words in Kdict, you will need to adjust with the settings of your personal translator. To do this, open the settings dialogue via `Settings/Install Kdict` (Figure 2). If you've decided on a local server, change the default from `dict.org` to `localhost` and then click on the Apply button. You can happily accept all the other settings under this item, as specifications for

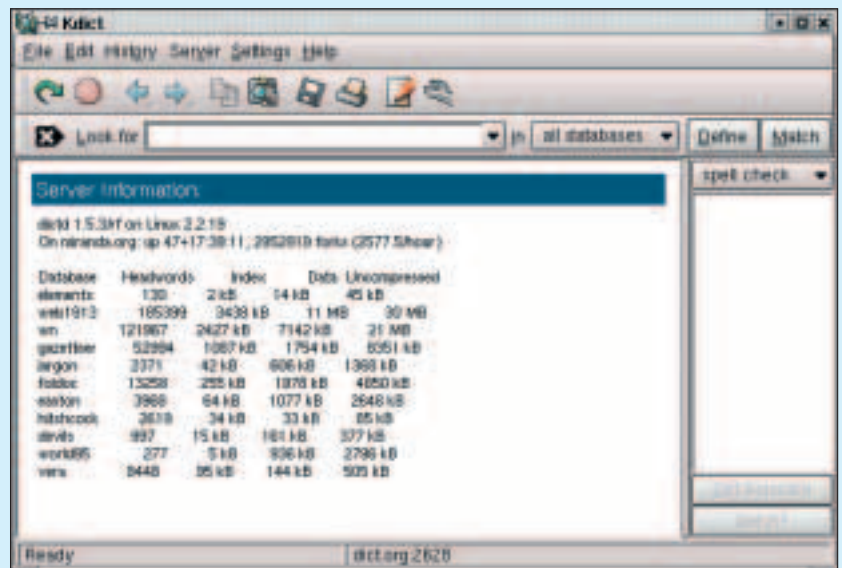


Figure 3: This is what your server has to offer

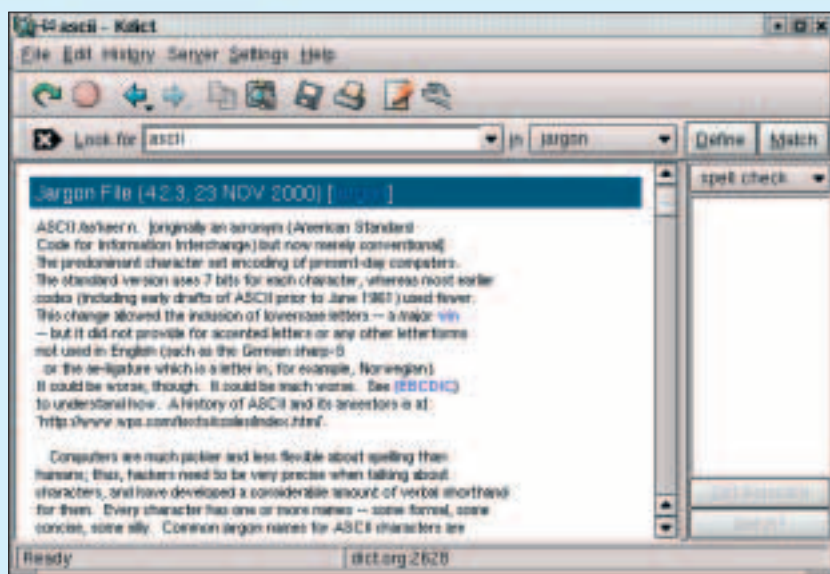


Figure 4: A database as you like it

the Timeout and so on only make sense for online queries. Incidentally, Timeout defines how long Kdict should wait for a reply from the server.

In order to play safe, it's wise to test your configuration by invoking *Server/Server Information*. Kdict will then provide you with a few bits of information regarding the status of your server (Figure 3). Next, start *Server/Get Capabilities* to provide Kdict with a list of available databases and search strategies from the server set (in our case *localhost*). In future the dictionary front-end will show you all the available databases and strategies in its selection menus.

As you can imagine, the Kdict electronic reference library is very easy to use. If you want to look for a word or definition, enter it in the Look for: field and then press the Enter key, or click on the magnifying glass icon on the toolbar. In the pull-down list next to the Look for: field you can also define the available databases in which Kdict should search.

As a fully-fledged KDE application, Kdict is adept at finding its way around the KDE clipboard and can look up its contents should you wish. To do this, select the menu item *Edit/Match Clipboard content*. If you regularly use this feature it's advisable to create your own keyboard shortcut via *Settings/Configure key bindings*. Alternatively, you can mark a word with the left mouse and then click on the Kdict window with the middle mouse button to start the search.

As it often makes sense to conduct a search in just one part of the databases, Kdict gives you the option to define your own set of databases. These are free compilations of the existing databases, such as all English-Latin dictionaries, which appear as virtual databases in the database selection list. To create this type of set, select *Server/Edit Database sets* and assemble your own database combination using the

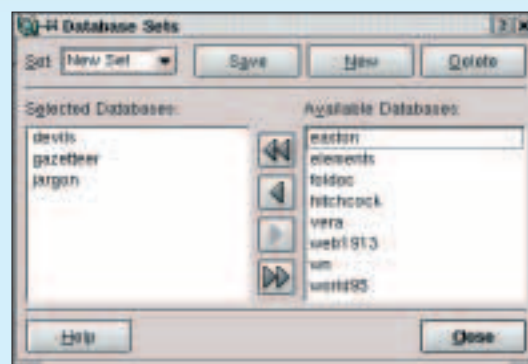


Figure 5: Any amount of information

arrow keys (Figure 4).

Whether you use a pre-defined dictionary or one of your own concoction, Kdict will display the result of your search in a window, as in Figure 5. The program also defends itself against the error gremlins that will always try to slip in. If Kdict finds no suitable term in its databases it offers you a list of similar-sounding words on the right-hand side of the window, from which you may find your term. If so, look it up immediately by marking it and choosing the Load selection button. If you want to look up all the alternatives in one go, the Load all button will satisfy your needs.

Don't be surprised if some words appear in blue in the definition. This is due to another of Kdict's useful features. If you click on one of these highlighted words, Kdict immediately shows you the associated definition of this term – hyperlinks à la Kdict.

Panel & Co.

One of the new options offered by Kdict, lets you embed the program in the panel. In order to do this, select *Configure Panel/Add/Applet/Dictionary* from the K menu. You will then immediately see an applet as in Figure 6. If in future you are studying a Web site, manpage or readme file and there are a number of terms you are unfamiliar with, you can simply enter them into this box; Kdict looks up the term and presents you with the results of its efforts in the usual way.

If, on the other hand, you just want to quickly look up a word, you can make use of the command line options of your little helper. In this case, start Kdict in the following way:

```
kdict word_you_want_to_find
```

If you want to look up several terms in one go, you must place them in quotation marks. With a *kdict -c* or *kdict -clipboard*, Kdict looks up the content of the clipboard. A *kdict -h* gives you an exact overview of the additional command line options.

Figure 6: Swallowed

