Managing Recipes with Krecipes

Easy Cooking

No need to juggle with cookery books, just to keep those friends that you invited round for a meal happy. Krecipes allows amateur chefs de cuisine to manage their recipes and create shopping lists. It can even help you with a diet if needed. BY FRAUKE OSTER

It is just a year old, but it has lots of useful ingredients: the Krecipes [1] recipe management program makes your Linux PC a useful tool in the kitchen. The last program package was a pre-release from just before Christmas 2003, version 0.4.1. The three developers working on the project have really gotten things cooking since then, so you can look forward to version 0.5.

The ***kdenonbeta*** **CVS** module, which is stored on the KDE servers, has a downloadable snapshot. Krecipes is quite unusual for a *kdenonbeta* program in that it has been capable of speaking six languages ever since version 0.4.1. Some languages are not quite complete as the online help is missing, so if you feel like helping out, don't hesitate to contact the developers.

Choose Your Recipe Repository!

Krecipes stores recipes in a database; you have a choice between **MySQL** [2] and **SQLite** [3] at present. PostgreSQL is due to be added later, probably in line with the version 1.0 release. MySQL and SQLite support works fine, but which of these databases is the better option?

MySQL is extremely complex and has its own database server that needs to be running when you launch Krecipes. If you already have a MySQL server set up on your computer, you can assign recipe management to this server. If not, amateur cooks might prefer to use SQLite.

To use MySQL you need to install the mysql package, the Qt dev(el) package, and *qt3-mysql* from your distribution disks. If the *qt3-mysql* package is missing, your only option is to compile Qt, which will create a MySQL plug-in for the library (see Box 1). You will not be able to use Krecipes with MySQL without completing these steps.

Things are a lot easier if you opt for SQLite, as Krecipes implements the required bridging functionality itself. Download the SQLite package off the Internet [3]. Then unpack the SQLite package, change to the source directory, and launch the configuration routine:

```
tar xvzf sqlite-2.8.12.tar.gz
cd sqlite
./configure
```

If this works out okay, you can carry on with the compilation and installation steps. You can install SQLite system wide by entering *make install*. This requires system administrator privileges:

```
make
su -c "make install"
```

Enter the Recipe Manager

After setting up the database, less adventurous users can download the Krecipes source code from the project homepage. Every else can boldly go to the developer branch of KDE via the CVS.

No matter what version of Krecipes you opt for, make sure that the *kdelibs* dev(el) packages are installed on your system. If not, install the appropriate package (*kdelibs3-devel-version.rpm* or similar) for your distribution. This is the only way to build KDE programs from the sources.

If you are using the Krecipes sources, you first need to unpack the tar archive:

```
tar xvjf krecipes.tar.bz2
```

Change to the *krecipes* directory created by this step, and launch the configuration tool. The tool needs the path to the Qt root directory:

GLOSSARY

CVS: *The "Concurrent Versioning System" provides a version management system for files. The source code is stored on a server where developers can add bug fixes or new functions. You can use this system to download the sources to your own computer. This in turn allows you to build an executable version.*

kdenonbeta: *Programs from kdenonbeta are typically in an early design stage and are not officially part of the KDE release. This means that they are not affected by release plans or freezes. On the downside, programs from this branch are not typically available as RPMs or Debian packages. Locale support is also restricted.*

MySQL: *A database that supports the formulation of queries in the SQL ("Structured Query Language") database language. Data are stored in tables. You can use the command-line client mentioned later on in this article to manage MySQL databases. PhpMyAdmin,. an application written in the PHP scripting language, is a popular alternative client. PhpMyAdmin has a GUI and supports browser access. To use PhpMyAdmin, you will need a working Apache Web server, however, and that does mean some additional installation work.*

SQLite: *A C library with embedded database server software. If you link Krecipes up with SQLite, the program can then access SQL databases without needing a separate database server running. The SQLite library reads and writes the database directly from or to the disk.*

Figure 1: MySQL insists on setting up a database user.

Figure 2: A MySQL configuration needs user credentials.

The pull-down menu to the right of the *Search:* field allows you to restrict the search to specific categories. Categories are assigned when entering recipes. A single dish can be assigned to multiple categories.

After finding and selecting the desired recipe in the list, you can use the two longish buttons below the list view to open or edit the recipe. The *Delete* button removes unhappy memories of culinary mishaps from the database.

```
cd krecipes
./configure -with-qt-dir=➥
/directory
```

To install, enter the *make* and *make install* commands.

If you opted for MySQL, you now need to launch the database server. You should be able to launch the MySQL server, or set the server up to launch automatically when you fire up your machine.

The first time you launch the recipe management tool, by typing *krecipes &* in the console, or via the K menu, a wizard appears to help you with the initial setup. MySQL users have two steps more than SQLite users, as MySQL needs a database user.

Jump Start

The wizard asks you what database you will be using immediately after displaying the welcome screen. If you opt for MySQL, you can specify whether the database user has the required database privileges, or if you have set up a system administrator for the database (see Figure 1). If you do not select any of these options, and click on *Next* > instead, the next dialog displays your current Linux username without the password. You need to supply a password. If not, Krecipes will not run for lack of user credentials, even if you complete the rest of the steps in the wizard. Box 2 tells you how to create a MySQL database user.

In the last dialog box, which pops up immediately if you select SQLite, the wizard asks you if you want to put the sample recipes in the database. Don't expect too much; there are only four recipes at present. Close the dialog to launch Krecipes in the typical "*Find/Edit Recipes* " view(see Figure 3).

Finding the Right Recipe

Each program view has a navigation bar on the left. The rest of the window area has a browsable list of recipes. If you are looking for a specific dish, you might prefer to use the search function – especially as the database starts to grow.

To do so, enter the recipe name in the *Search:* field, and hit [Enter] to confirm. This tells Krecipes to display the recipe, or a list of all possible recipes. The list indicates the category, the recipe title, and the index number assigned when the recipe was created. You can click a column heading to sort by each of these criteria.

Of course you need to create a few recipes first, before you can start deleting them. The *New* button in the toolbar takes care of this. A bug causes a display error in the dialog box if you have Qt 3.3 (see Figure 4).To remove this error, change the window size by dragging the lower right-hand corner of the window.

New Creations

Three tabs are available for entering your recipe data. The first tab, *Recipe* is for the recipe title, the name of the author, the categories and the number of people the recipe serves. The + button next to the *Authors* and *Categories* fields pops up a dialog box where you can select an author or category.

The second tab is for entering the ingredients and quantities. A list of known ingredients makes things easier.

Box 1: Compiling Qt to add MySQL support

To compile Qt, you need the Qt sources which are either available on your distribution disks or from the Qt manufacturer, Trolltech [4].

After unpacking the archive, change to the directory with the Qt source code. Use the *-plugin-sql-mysql* option to configure the library features to add MySQL support to Qt:

```
cd qt-directory
./configure -system-zlib -qt-gif
-system-libpng -system-libjpeg -
plugin-imgfmt-mng -thread -no-
exceptions -plugin-sql-mysql
```

The *-system-zlib* configuration option enables the use of the *zlib* compression library on your machine, this also applies to the libraries for the PNG and jpeg image formats. Instead of using a library for the GIF image format, you should use the Qt feature (*-qt-gif*), and Qt needs a plug-in for the MNG ("Multiple-Image Network Graphics") image format.

The *-thread* option enables multithreading for Qt-based programs. The final configuration option that you need to enable as a default is *-no-exceptions*; this means that exception handling as defined by program authors will not be excepted. As the example shows, you can add more options when calling the command.

After finishing off the configuration steps, go on to build the Qt library and a few tools that you need for Qt and KDE:

```
make sub-src sub-tools
```

This can take a while. There is no need to install Qt when you finish, however!

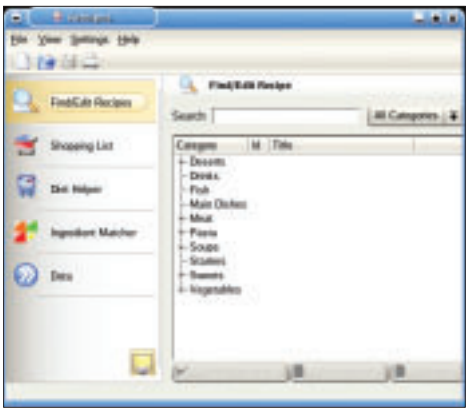

Figure 3: Using the "Find/Edit Recipe" view.

You can select a unit for the required quantity or enter a new unit. If you have all the details you need for the ingredients, click *Add ingredient* to store them.

You can use the arrow keys to move ingredients around in the list. This allows you to enter the ingredients in the order in which you will need them for the recipe. Click on the button with the cross to remove incorrect ingredients.

The third tab has a large text box for the cooking instructions. You can click on the button below the text box to spell check your entries.

There are several buttons below the tabs that allow you to save the recipe (floppy symbol), close the dialog (circle with an x), display the recipe (magnifying glass), or add it to your shopping list (the shopping cart icon, see Figure 4). Also, the *Scale recipe* function (that's the button with the blue arrow) allows you to change the number of people served and automatically calculate the additional quantities of the ingredients.

If the recipe view (see Figure 5) is not your liking, and you would prefer to have the ingredients in a different place, why not change the *Settings* | *Page preferences…*? You can drag the individual elements, for example the category or ingredient list, with your mouse to rearrange them (see Figure 5). Click on *Save and Close* to store your changes.

Shopping Planner

After deciding on a menu, Krecipes' *Shopping List* function, which is located in the navigation bar on the left-hand side of the window, can help take the pain out of shopping.

As shown in Figure 6, you can select one or more recipes from the list of existing recipes, and use the right arrow button to add them to your *Shopping list*. Clicking the left arrow removes the recipes from the list, and clicking on *Clear* removes the whole list.

The *OK* button displays your shopping list (see Figure 6). You can click *Print* to create a hard copy for you next trip to the supermarket. Click on *OK* to close the window.

At the bottom left in the shopping list window there is another button called *Diet Wizard* that leads you to the same dialog as the *Diet Wizard* function in the navigation bar. This tool really can help you create diet plans with lots of detail (see Figure 7).

Fighting the Flab

First of all, you need to specify how long you intend to diet, how many meals per day you will be eating, and the number of courses each meal should comprise. If you have more than one dish per meal, you can use the *Previous Dish* and *Next Dish* functions to toggle between dishes.

If you check *Enable Category Filtering*, Krecipes will only display selected recipe categories. This allows you to avoid pizza or steak for breakfast, for example.

On the lower right you can select the levels of carbohydrates, vitamins, proteins, and fat. Krecipes will offer a limited selection of recipes that match your preference. When you click *Create the diet*, Krecipes not only compiles a dietary plan for the specified period, but also displays the plan.

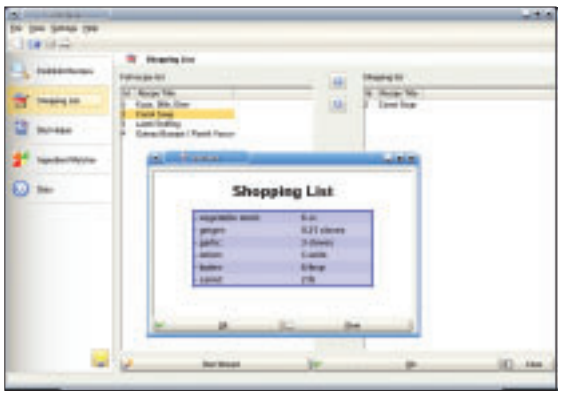

Figure . 6: Automagically creating a shopping list.

Ever opened the refrigerator door and not known what to make with the ingredients you found? The *Ingredient check* item in the navigation bar can help you. Simply select the ingredients you have at home from the list, and click on *Find matching recipe*. Let's hope you find a nice recipe in the list and can head off down to the kitchen with a mission.

Ingredients

If you were wondering in the "Fighting the Flab" section if Krecipes really knows the nutritional value of somewhat unusual foodstuffs, such as pancetta, urad dal or water spinach, check out the *Data* item in the navigation bar to find out. This leads you to more navigational entries to edit your collection of data.

For example, the *Ingredients* item in Figure 8 leads to a list of all the ingredients in the database. You can add any missing items here, that is anything that has not been used so far in a recipe, just in case. To add an ingredient, click the button with the plus sign. To remove an ingredient, click the button with the

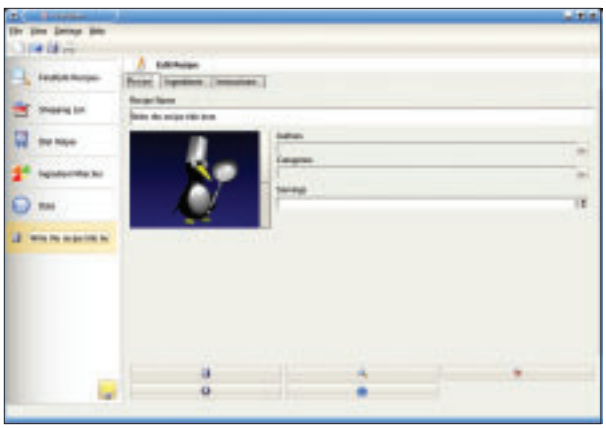

Figure 4: You need to enter the recipes.

Figure 5: Changing the view.

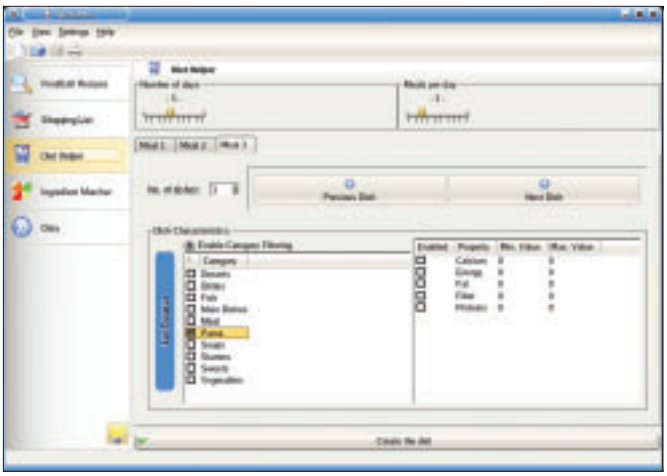

Figure 7: Krecipes diet helper for healthy eaters.

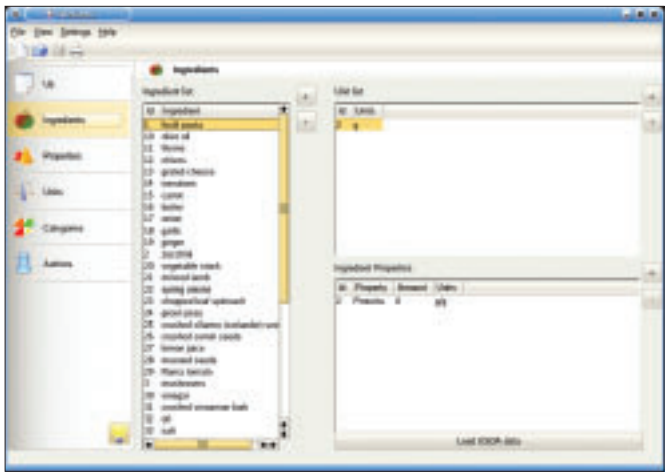

Figure 8: You can access the "Data" item to add entries to the database.

minus sign. Both buttons work the same way in the other lists.

You can assign an appropriate unit, and of course define the protein, fat and other nutritional values for each ingredient. *Load USDA data* makes this easier, however, it is not quite finished.

What it does so far is to load a list of foodstuffs and their nutritional values from a database at the United States Department of Agriculture [5], but it fails to add the list to the Krecipes database. The developers are working hard to get this function sorted out.

Properties leads you to an overview of the nutritional values entered so far. You can add entries, and remove superfluous entries, using the plus and minus buttons as described previously.

Units leads to a list of all the units the database knows, including a conversion table for joules to calories. You can add more conversion tables. Follow the steps outlined above to add new units.

Categories manages recipe categories, and *Authors* is the place to add and edit the names of recipe providers.

Import

You might have a lot of data to add, and will not want to re-type all those recipes that you have already entered into your PC. Wouldn't it be nice to add all those recipes you discovered on the Internet to your database without all that typing?

To allow this (in most cases at least), Krecipes supports Recipeml ("Recipe Markup Language") [6], a XML based language especially designed for displaying recipe data. Krecipes has no trouble opening recipes in this format. It can import recipe formats from programs like Master Cook or Meal Master into the database via the *Import* in the *File* menu.

The program setup dialog below *Settings* | *Krecipes preferences* (see Figure 9) is the place for tweaking the recipe import facility. You can also change the username and password for the MySQL database. *Numbers* modifies the way numbers are displayed for quantities.

All of these functions work well, despite the early stage of development. This makes Krecipes suitable for daily use, and augers well for the future.

Figure 9: The "Krecipes preferences" dialog allows you to configure Krecipes.

Box 2: Setting up a MySQL database user

To assign the required read and write privileges for the MySQL database to a user, become the database administrator, *root*, in the Linux command line, and open up a MySQL shell. Take care not to confuse the *root* database administrator with the *root* Linux system administrator!

If this is the first time you have worked with the *root* database admin account, you will not have assigned a password. This does not mean that the user does not exist, but simply that the account is unprotected. In this case, you do not need to supply a password to access the MySQL prompt:

```
mysql -user=root mysql
```

To assign read and write privileges for your recipe database to your own non-privileged account, type the following command at the prompt. This assigns select, insert, update and delete privileges for the database:

```
GRANT SELECT, INSERT, UPDATE,
DELETE ON Krecipes.* TO
'username'@'hostname' IDENTIFIED
BY 'password' ;
```

The recipe database that Krecipes automatically creates is called *Krecipes*. Replace *username*, *hostname* and *password* with the database user credentials, the password, and the hostname. If you are running the MySQL server on your local machine, use *localhost* as the hostname. Alternatively, replace *Hostname* with the remote database server name (this can be a dedicated server on your local network, for example).

You can now tell the *Krecipes* assistant that you have the required privileges (see Figure 1). In the next dialog box (see Figure 2), enter the user credentials and password that you just assigned.

INFO

[1] Krecipes: *http://krecipes.sf.net/*
[2] MySQL: *http://www.mysql.com/*
[3] SQLite: *http://www.sqlite.org/*
[4] Qt download: *http://www.trolltech.com/download/qt/x11.html*
[5] United States Department of Agriculture: *http://www.usda.gov/*
[6] More information on Recipeml: *http://www.formatdata.com/recipeml/*