The KDE Menu Editor Handbook

Milos Prudek
Anne-Marie Mahfouf
Lauri Watts

Contents

Abstract

KMenuEdit allows editing one of the KDE application launchers.

Chapter 1

Introduction

KMenuEdit allows editing the menu of KDE application launchers.

KMenuEdit can be started either by right mouse button clicking the application launcher button on the panel and choosing **Edit Applications...**, or by typing `kmenuedit` in the KRunner line.

KMenuEdit allows you to:

- View and edit the menu used by the current application launcher

- **Cut**, **Copy** and **Paste** submenus and items

- Create and delete submenus and items

- Change the order of submenus and items

- Hide items and add new submenus and items

By default all applications installed on the computer appear in the application launchers menu of all users. Applications may appear more than once in several different menus. Empty submenu categories defined in the desktop menu specification will show up as well, but are not visible in application launchers unless you install applications belonging to these categories.

The tree view has two different entry types:

- Submenu: Only **Name**, **Comment** and **Description** fields and the button to select an icon are enabled, the **Advanced** tab is disabled. Any submenu can hold additional submenus and/or items.

- Items: Use this entry to enter the data for the application you want to add. For detailed information see Using KMenuEdit.

KMenuEdit has two tree view modes - normal and with hidden entries. To see the latter check the option **Show hidden entries** in the configuration dialog on the **General options** page.

Many additional entries appear in hidden view mode in the tree. Several hidden entries seem to be just duplicates but have different command options. Usually you should never change these hidden entries or you risk to break some functionality of the system.

In hidden mode you will have a special submenu **.hidden [Hidden]** as top level item in the tree. This special submenu is not editable. In this submenu all deleted items will be shown at the next start of KMenuEdit.

It is not possible to delete entries using the GUI in this special submenu. They will reappear at the next start of KMenuEdit.

1.1 Use Cases

1.1.1 Adapt the Menu for a User

1.1.1.1 Reorder Items

This should be done in hidden view mode where only the submenus and items visible in the application launcher menu are displayed.

By default the menu is sorted alphabetically using the English names or descriptions. Using another language than English some submenus and items therefore will appear in an unsorted order.

Use the options in **Edit** → **Sort** to sort either by name or description. If you use the classic application launcher you have to select the corresponding **Format** option in the settings dialog. In the Kickoff launcher check **Show applications by name** if you sort them by name here.

Grouping your frequently used submenus or items together e.g. at the top of the menu makes selecting them easier. To change the order of particular items or submenus in the tree use the **Move Up** or **Move Down** button in the toolbar or these actions in the menu.

All application launchers will use the order of submenus defined in KMenuEdit.

1.1.1.2 Remove Items from the Menu View

Having shown all applications installed on a computer may be confusing for some users so you may want to hide some less frequently used items or submenus. There are two different ways to do that:

Switch to normal view mode without hidden entries. If you delete items they are moved to the **.hidden [Hidden]** submenu. You can move them back into the tree to have them in the menu again.

If you delete a submenu it will be really deleted with all its submenus and items. To recreate them you can use **Edit** → **Restore to System Menu**, but this will remove all your custom submenus and items and the corresponding `.desktop` files are deleted too. This action cannot be reverted.

The preferred way to remove submenus and items in the menu of an application launcher is to check **Hidden entry** on the **General** tab and switch to the hidden view mode. In this mode it is easy to revert changes without destroying the menu structure. The only drawback is that you have to hide all entries in a submenu manually to hide the whole submenu from the view.

1.1.2 Adding Custom Items

To add custom items (submenu or item) use the actions in the menu or toolbar. Items need a Name and a Command, without a command entry an item will not be saved and your addition gets lost.

If you add an entry it is inserted as sub entry at the actual highlighted position in the tree. Move an entry by dragging it with the mouse or using the **Move Down** button to the bottom of the tree to make it a top level entry.

1.1.3 Transfer application launcher settings

There is no way to transfer menu settings using the GUI, you have to do that manually and copy the following files to the target user:

KMenuEdit stores the menu hierarchy in `menus/applications-kmenuedit.menu` in the folder `$(qtpaths --paths GenericConfigLocation)` and `desktop-directories` in the folder `$(qtpaths`

`--paths GenericDataLocation)` contains desktop files for submenus you created. In `applicati ons/` in the folder `$(qtpaths --paths GenericDataLocation)` you find the desktop files for the custom items you created.

The shortcuts for each application are stored in `kglobalshortcutsrc` in the folder `$(qtpaths --paths GenericConfigLocation)`, but export/import does not work because the UUIDs of the shortcuts do not match up between systems, even though the `.desktop` files are the same. You have to assign all shortcuts manually again.
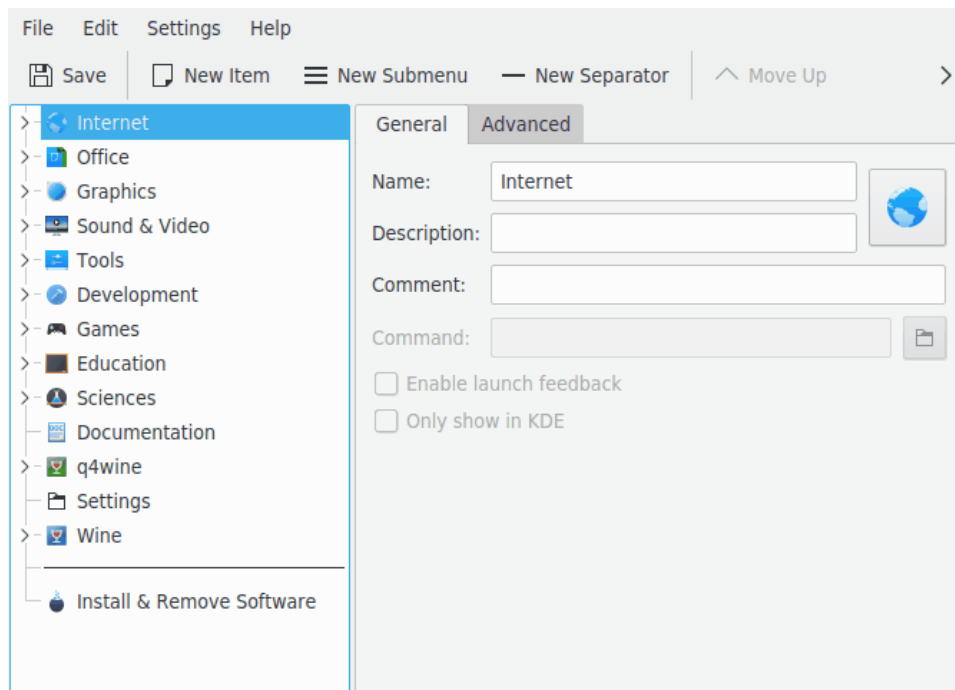
Chapter 2

Adding a Menu Entry

Virgil J. Nisly

In this example, we will add Firefox to the **Internet** submenu.

To start off, we need to open KMenuEdit, so right mouse button on the application launcher menu, click **Edit Applications...** to start KMenuEdit. After KMenuEdit has started, select **Internet** as shown in picture below.

Once you have selected **Internet**, click on **File** → **New Item...**, opening the **New Item** dialog, as shown below. Type the name of the program you want to add, in this case, type **firefox**.
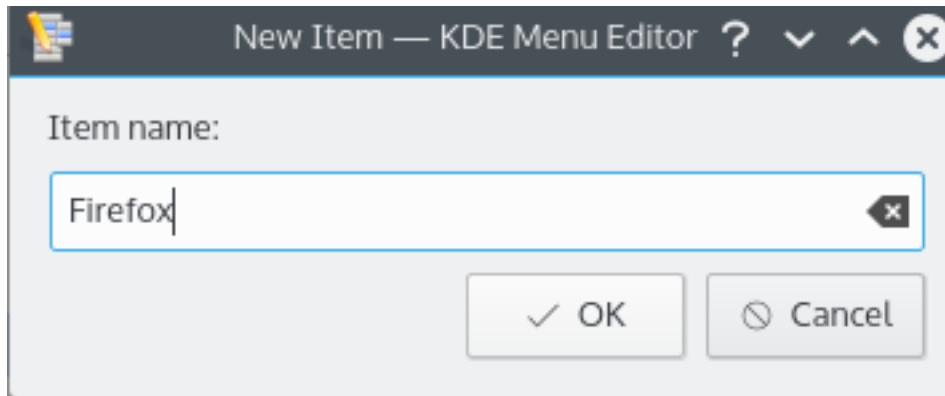
Press return, and you should see something like the picture below in the main window.
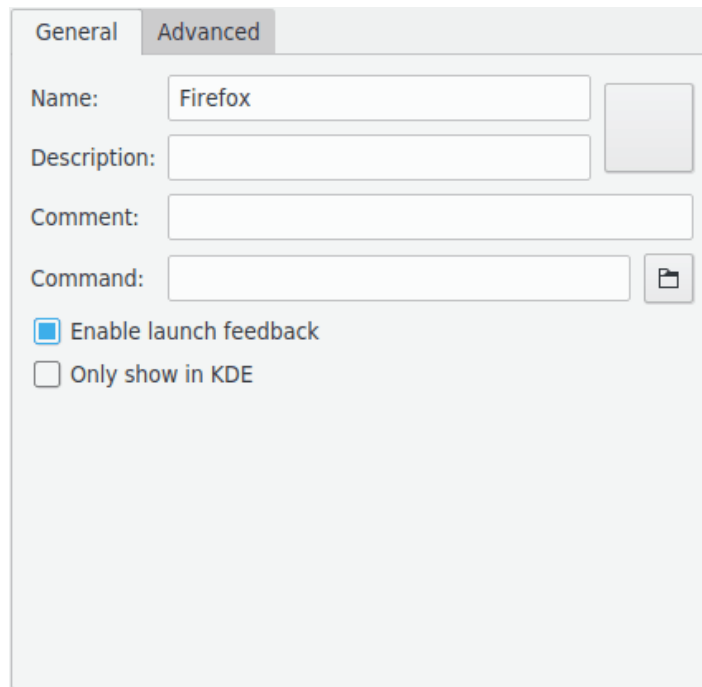
Now lets fill in the **Description:**, in this case type **Web Browser**.

> NOTE
> The appearance how description and name will be displayed depends on the settings in the application launcher e.g. 'Web Browser (Firefox)' or 'Firefox (Web Browser)'.

We will need to fill in the executable name in the **Command:** field, in this instance we will type **firefox**.

The command has to be in your PATH variable or you have to specify the full path to the executable. If you do not know the executable name of an application use the **locate** command to search for the desktop file and enter the string from the Exec line as command here.

NOTE
Following the command, you can have several place holders which will be replaced with actual values when the program is run:
  %f - a single file name
  %F - a list of files; use for applications that can open several local files at once
  %u - a single URL
  %U - a list of URLs
  %d - the folder of a file to open
  %D - a list of folders
  %i - the icon
  %m - the mini icon
  %c - the caption
For example: if you want to firefox to start your web browsing at www.kde.org - instead of **firefox** you would type **firefox %u www.kde.org**.

Most applications accept additional options e.g. the name of a defined profile like Konqueror or Konsole. To see all options for an application launch `applicationname --help` in Konsole.

We would like to have a more creative icon, so we will click the generic icon sitting beside **Name:** (note, the default icon may be blank, in which case click in the area to the right of the name entry box.) It will bring the **Select Icon** dialog which will let us choose the new icon, as shown below.
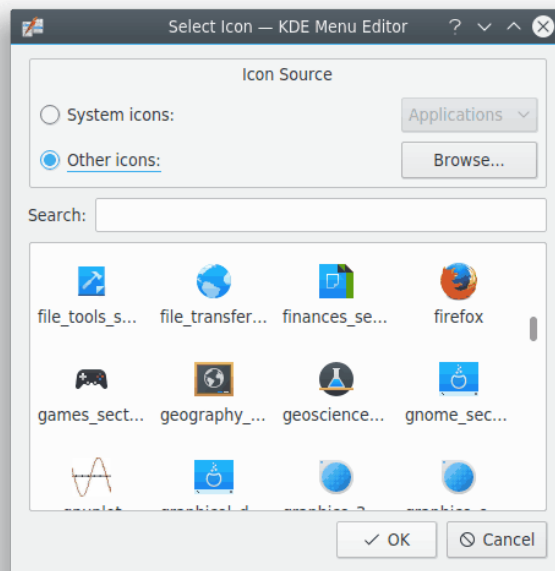
We choose the firefox icon from the list, and press **Enter**. Your finished screen should probably look something like the screenshot below.
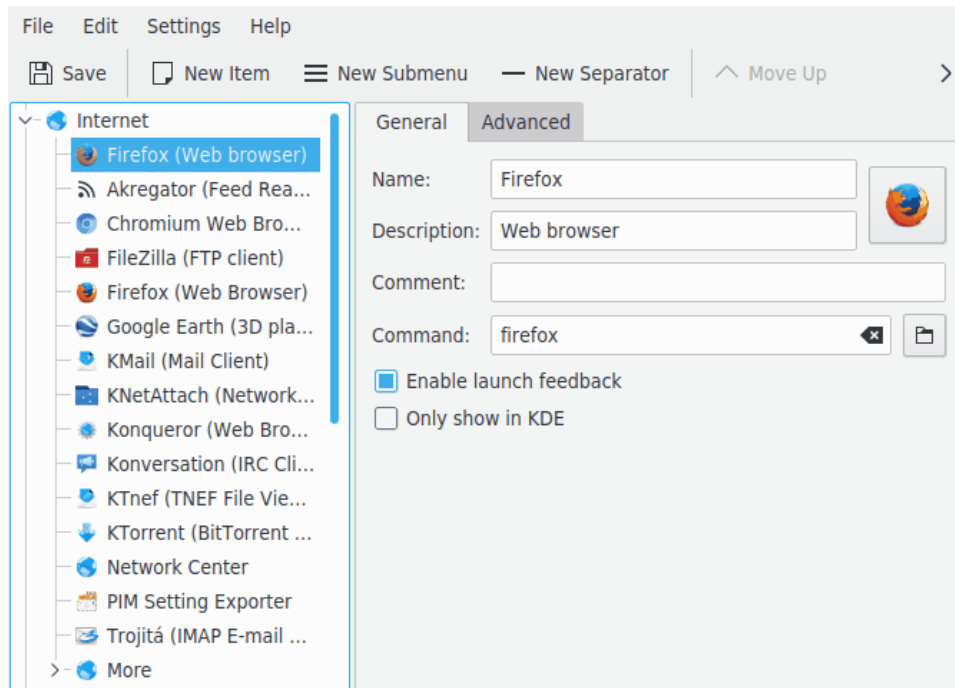
The place of the new menu item can now be changed using **Move Up** and **Move Down** buttons on the KMenuEdit toolbar or by dragging with the mouse.

The submenu items can be sorted using **Sort** button on the KMenuEdit toolbar or **Edit** → **Sort** submenu items.

Click **File** → **Save**, wait for the **Updating System Configuration** dialog to get finished, you should find Firefox in the application launcher **Internet** submenu.

Chapter 3

Using KMenuEdit

The left application panel shows the application launcher structure. When you browse items in the left panel, the right panel shows detailed information for the highlighted menu item.

3.1 General tab

Name:

This is the name of your program as it appears in the application launcher menu. It can be different from the real executable name. For instance the name of **mc** executable is "Midnight Commander".

Description:

The description will be displayed together with the name in the application launcher. This is entirely optional.

Comment:

Describe the program in greater detail in this field. This is entirely optional.

Command:

This is the name of the executable program. Make sure that you have permission to run the program.

Enable launch feedback

If this box is checked, this will display feedback when an application is started.

Only show in KDE

When checked, the application entry will only be visible in all KDE application launchers but not in other desktops environments.

Hidden entry

Remove an entry from the menu view in the application launcher.

3.2 Advanced tab

Work path:

Specify the work path of the program. This will be the current path when the program launches. It does not need to be the same as the executable location.

Run in terminal

You must check this if your program requires terminal emulator in order to run. This mainly applies to console applications.

Terminal options:

Put all terminal options in this field.

Run as a different user

If you want to run this program as a different user (not you), check this checkbox, and provide the username in the **Username:** field.

Current shortcut key:

You can assign a special keyboard shortcut to launch your program.

Click the **None** button to the right of the **Current shortcut key:** checkbox.

The button text will change to **Input...** and you can press the key combination on your keyboard that you want to be assigned to your program.

You can reset the shortcut to **None** by using this button: .

Don't forget to save your setting by clicking the toolbar **Save** icon or using the **File → Save** menu item.

Chapter 4

Menu Reference

Most actions in the menubar are also available in the context menu opened with a right mouse button click on an item in the tree view.

File → New Item... (Ctrl+N)
    Adds new menu item.

File → New Submenu...
    Adds new submenu.

File → Save (Ctrl+S)
    Saves the menu

File → Quit (Ctrl+Q)
    Quits KMenuEdit.

Edit → Move Up

    Moves the selected item up in its submenu.

Edit → Move Down

    Moves the selected item down in its submenu.

Edit → Cut (Ctrl+X)

    Cuts the current menu item to the clipboard. If you want to move menu item, you should first cut it to the clipboard, move to the destination place using the left panel, and use the **Paste** function to paste the menu item from the clipboard.

Edit → Copy (Ctrl+C)

    Copies the current menu item to the clipboard. You can later use the **Paste** function to paste the copied menu item from the clipboard to its destination. You can paste the same item many times.

Edit → Paste (Ctrl+V)

    Paste menu item from the clipboard to currently selected place in the main menu. You must first use **Cut** or **Copy** before you can **Paste**.

Edit → Delete (Del)

    Deletes currently selected menu item.

Edit → Sort

    Opens submenu to sort selected submenu or all menu tree. There are two sorting methods implemented, namely, by the name and by the description.

Edit → Restore to System Menu

> This will restore the application launcher as it was as default and remove all your custom settings. A message box will ask you if you really want to do that.

KMenuEdit has the common KDE **Settings** and **Help** menu items, for more information read the sections about the Settings Menu and Help Menu of the KDE Fundamentals.

Chapter 5

Credits and License

KMenuEdit

Program copyright (c) 2002, Raffaele Sandrini

Contributors:

- Matthias Elter elter@kde.org - Original Author
- Matthias Ettrich ettrich@kde.org
- Daniel M. Duley dan.duley@verizon.net
- Preston Brown pbrown@kde.org

Documentation copyright (c) 2000 Milos Prudek

Documentation copyright (c) 2008 Anne-Marie Mahfouf

Updated for KDE 3.0 by Lauri Watts lauri@kde.org 2002

This documentation is licensed under the terms of the GNU Free Documentation License.
This program is licensed under the terms of the GNU General Public License.

Chapter 6

Glossary

Console Application

Application originally written for non-graphic, text oriented environment. Such applications run fine in KDE. They must run within console emulator, like Konsole. They are not warned automatically when you shut down your KDE session. You therefore must not forget to save documents open in these applications before you log out from the KDE.

Console applications support copying and pasting from KDE-compliant applications.Simply mark the text in the console application with your mouse, switch to the KDE-compliant application and press **Ctrl**+**V** to paste the text. If you want to copy from KDE application to a console application, first mark the text with your mouse, press **Ctrl**+**C**, switch to the console application and press the middle button on your mouse[1].

---

[1]If your mouse does not have a middle button, you must press left and right button at the same time. This is called 'middle button emulation' and it must be supported by your operating system to work.