

## A first look at KDE 4 beta 2

As part of Ars Technica's continuing coverage of the KDE 4.0 development cycle, I took the opportunity today to sit down with a fresh build of KDE 4 from KDE's publicly accessible source tree.

By Troy Unrau | Last updated September 4, 2007 1:32 AM CT



As part of Ars Technica's continuing coverage of the KDE 4.0 development cycle, I took the opportunity today to sit down with a fresh build of KDE 4 from KDE's publicly accessible source tree. My tests are conducted on a 64-bit Kubuntu 7.04 system with very few modifications other than the installation of a bunch of *-dev* packages that were required to build the many parts of KDE 4. I built the whole thing to be self-contained within a single user's home folder, so that I can safely test it without making any permanent changes to my otherwise perfectly functional KDE 3.5.7 machine. To my surprise, today is the first day that I can say that I'm really comfortable using KDE 4, which is good news for the KDE community at large who have (up until recently) been somewhat afraid of 'eating their own dog food' and using KDE 4 full time. Beta 2 was tagged last week, so the sources that I am testing are fairly similar to that release, which will be announced with packages for many distros early this week.

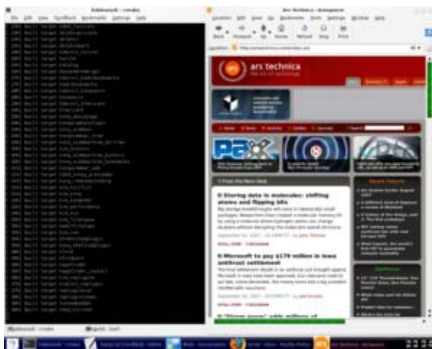
## Building the core of KDE 4, skipping the dirty details

Building KDE can be somewhat tiresome the first time, but being somewhat seasoned at this task, I simply followed the directions over at the KDE Techbase to set up a few shell macros to make my life easier. Other than the actual KDE sources, I also built by hand KDE's in-house branch of Qt 4.3.1 (known as qt-copy) using the instructions in the accompanied readme file, and FreeDesktop.org's Shared-Mime-Info 0.22, a project that shares mimetype definitions between the various desktops which were previously each implemented independently. KDE's modules are nicely split into a number of pieces, of which 5 or so are required to build before getting a functioning desktop at this stage. I'll go over them briefly to give an idea of what each module does, in the order that they needed to be built.

First, there is the kdesupport module that contains a number of libraries upon which KDE depends but that are not KDE specific. These libraries include Strigi, a comprehensive meta-data extraction and search system. The Eigen library is a set of utility functions that allows for some of the fast vector math required in 3D visualizations (such as that required by Kalzium's 3D molecule viewer). Taglib is a tag extraction and manipulation library for a number of media formats and is used extensively by KDE's media players to provide a tagging API that doesn't depend on the file format in question. There are several other libraries contained in this module, and distributions usually package them separately. None of them are linked to KDE in any concrete way except that they are used by KDE and are often written by KDE developers.

Next comes the core of KDE, the kdelibs module. This module contains all of the libraries that are absolutely expected to be present on any system if you wish to use KDE applications. These libraries include many of the main KDE widgets, as well as core technologies like the KIO slaves required for KDE's famous network transparency. Also included in this package are a base set of functional libraries for hardware detection and management, spelling, multimedia, image handling, and more.

The kdeimlibs module contains a number of additional libraries (which used to be part of kdelibs in previous KDE versions) that relate to personal information management (PIM). This includes libraries for things like address book handling, calendaring, and email protocols which make writing PIM or other communication-based applications for KDE much easier. There are some libraries in here that are brand new to KDE 4 such as the blogging API which should allow for easy creation of blog management programs.



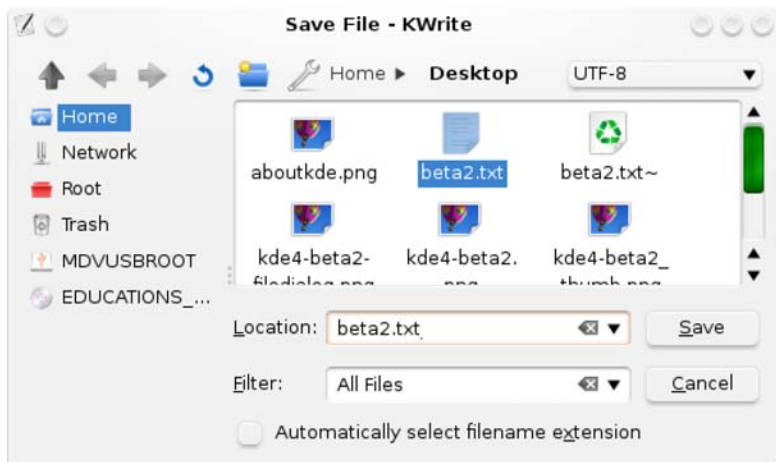
The kdebase module being built from within the KDE 4 environment

The big fish to build is the kdebase module. This module has been heavily reworked for KDE 4 to help clearly distinguish those parts that depend on Xorg and the full KDE Workspace from those that can function without running inside the KDE environment. This has helped make porting efforts to platforms like Mac OS X smoother, as those platforms do not need a functioning KDE desktop or window manager. The kdebase module is home to many of KDE's core applications, from the Konqueror web browser and the KWrite text editor, to the newly minted file manager, Dolphin. Also included in kdebase are the updated, prettier, Composite-powered window manager and the new Plasma workspace. In addition to the kdebase module, there is an additional module known as playground/base where many of the new Plasma plugins (referred to as plasmoids) are being developed. At this early stage, building kdebase without building playground/base would leave you with an extremely minimalist environment (read as: no task bar) so one would expect at some point that the better plasmoids will eventually make their way into the proper kdebase module before 4.0.0 hits the streets.

There are a number of other modules that can be built that provide additional libraries or programs as needed, but many of them will not be releasing on the same schedule that KDE's core will be using. The core modules currently have a targeted release date of early December as the KDE release team recently added a few additional betas to the cycle to ensure that the release is stable.

## Impressions on this build

I've been following KDE 4 development very closely for quite a while and have noticed several neat features that have been independently cropping up all over the system in the last year. Perhaps it is the emergence of the new visual themes that make it feel fresh and new, but I get the feeling from this build that I'm finally starting to see the big picture.



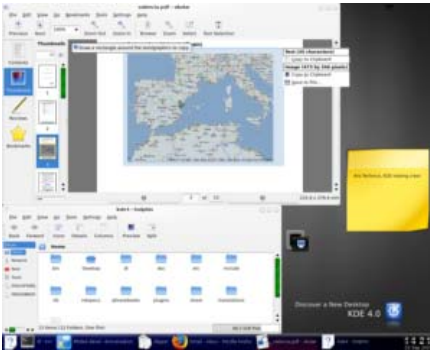
As an example, I plugged one of my flash drives into my system today. It didn't show up on the desktop as I expected it to (it probably will though before the final release hits the streets), but when I opened Dolphin, I found that the device was simply automatically registered and showed up in the sidebar alongside my Home, Trash, and other important folders. When I go to a file dialog in KWrite, it shows up there just as well. In either location, right clicking on it allows me to 'Safely Remove' the thumb drive just as I'd expect. Everything appears seamless, integrated, and stable.

In fact, the only crashes I've experienced when using this build is a recurring crash when exiting programs (probably some broken garbage collection or similar C++ voodoo). I've even risked writing the text of this article in KDE 4's KWrite, something that I would not have attempted a month ago. That's not to say that there are no problems though: there are some drawing glitches that are showing up in the new Oxygen style, and the new window decoration buttons need some sort of labels still. There are a few icons that are still missing, but the artwork is filling out quite nicely (whether or not you like the new folder icons is a matter of taste, but all of the icons are beautifully rendered SVG!). When using the Composite extensions for KWin, I get a few performance problems, but that may be related to my specific combination of hardware and will likely be mostly solved before December.

In addition, while the Plasma workspace is extremely promising, it currently lags behind a few of the other KDE 4 programs in terms of feature completeness. While I did get a taskbar and clock on the panel, the Plasma project still definitely deserves the label of beta, for the time being. When KDE 4.0 is released, Plasma is expected to be functional but not mind-blowing (yet). It is a new technology that will see heavy work throughout the KDE 4.x series. By comparison, Kicker, the current KDE panel implementation, was first introduced in the KDE 2.0 release nearly 8 years ago. Plasma is built on a redesigned architecture that should allow it to far exceed the capabilities of KDesktop or Kicker in the long run, but these old technologies do have a substantial head start in terms of feature completeness.

07/04/2009

A first look at KDE 4 beta 2 - Ars Tec...



This beta may not be the end result yet, but there has definitely been a lot of progress since the last one. For the first time I have been able to stop writing about individual features within KDE 4 and start to talk about the level of integration, and general feel of the system to get an impression of how KDE 4.0 will feel like to use. With a few exceptions (like Plasma), KDE is now in a feature freeze which means that this build somewhat reflects the final product more than previous alpha and beta releases have. I will continue to use it on a day-to-day basis, loading the occasional KDE 3 application as needed so that I can watch the progress as it unfolds and keep everyone at Ars up to date.

Serving the technologist for  $1.0756 \times 10^{-1}$  centuries