Zack's Kernel News

Keep The Change

As of 2.6.0-test11, Linus Torvalds has abdicated control over the 2.6 tree, and entrusted it to Andrew Morton. This hand-off has been long in the works, and Andrew's influence over 2.6-test releases has grown steadily through October and November. The decision to make Andrew the 2.6 maintainer was under consideration perhaps as long ago as the late months of 2002 or before. All stable kernel series have been handed off to other maintainers.

David Weinehall is still the official maintainer of the 2.0 series (though new 2.0 releases are few these days); Alan Cox puts out the occasional new 2.2 kernel; and Marcelo is still actively maintaining 2.4. But the hand-off of the stable series to another developer continues to be a protocol under development, as is every other aspect of Linux kernel development. The experiment this time, is to give Andrew the final word on when the kernel is ready for the first true 2.6 release, 2.6.0. Always before, Linus has maintained the stable tree himself until he felt proud enough of it to relinquish control. Perhaps he has numbered the 2.6-test kernels such that he has still done that; but the numbering itself is significant.

Andrew will have a tremendous influence over the schedule of 2.6 adoption by the various Linux distributions; and also over the responsiveness of the direction of the 2.6 kernel to the needs of those distributions. Once Andrew releases 2.6.0, the 2.6 tree will become the official stable kernel series, the most advanced Linus kernel available. The 2.7 tree, as all development series, will be of interest almost exclusively to developers, especially during the first mad-cap year of its development. It will be interesting to see how the 2.8/3.0 hand-off differs from that of 2.6. ■

Walk on the wild side

A December altercation on the linux-kernel mailing list has shown that Linux still firmly roots itself in the practical world, avoiding the addition of features for their own sake.

The ide-scsi driver has been broken for some time, but there have been very few complaints about it, and no one has sent in any patching code to bring it up to date. At the time of this writing, it seems more likely that the driver will be removed from 2.6, rather than left there to rot. Bill Davidsen has argued strongly in favor of maintaining the driver, saying that there were Zip drives and ATAPI tape drives that required it; but as Linus Torvalds has pointed out, without someone caring enough about the driver to actually send in patches, the driver must remain dormant.

One of the key elements of open source development is its dependence on the group of people interested in a given project. Linus may control some aspects of kernel development, such as the quality of the code he accepts, the schedule of new releases, the particular approach used in implementing a given feature; but he has only a small amount of influence over what most developers work on.

With now over a thousand kernel hackers sending in patches, quite a broad area of development is covered; but it is not uncommon to find drivers, even those with hardware in active circulation in the world, slipping into an unmaintained state. It is, in a sense, the law of the wild.

Kernel development, like most other open source development, takes an unpredictable path. Certainly things like the Kernel Summit can help determine the key areas to be worked on during a given development series, but even that requires interest on the part of developers; and if their interests should change over the course of a given development cycle, the kernel will by its very nature of needing their input have to adjust to that change. ■

INFO

The Kernel Mailing List comprises the core of Linux development activities. Traffic volumes are immense and keeping up to date with the entire scope of development is a virtually impossible task for one person. One of the few brave souls that take on this impossible task is Zack Brown.

Our regular monthly column keeps you up to date on the latest discussions and decisions, selected and summarized by Zack. Zack has been publishing a weekly digest, the Kernel Traffic Mailing List for several years now, reading just the digest is a time consuming task.

Linux Magazine now provides you with the quintessence of Linux Kernel activities straight from the horse's mouth.

All over now

It recently came to light that the CramFS compressed filesystem, written originally by Daniel Quinlan, has become orphaned; some of Daniel's patches had been rejected, including some that seemed clearly good; and Daniel had stopped submitted his patches. Apparently Alexander Viro, the Virtual Filesystem maintainer since time immemorial, has been doing his own rewrite of the CramFS code, but this has not been confirmed. ■

Hard to handle

Pontus Fuchs has taken the radical step of trying to load certain MS Windows drivers under Linux. His ndiswrapper driver project attempts to interface with certain Windows network driver API drivers, just enough to get a given card working under Linux. The decision to try this bizarre approach was made because some vendors refuse to release their specifications, or even a binary-only Linux driver for their cards.

This seemed like the only viable solution to him. So far he has had some success with his Broadcom 4301 card; and Pavel Machek has also had success with it on his Broadcom 94306 card. It would be odd to see such a bizarre driver actually get into the kernel itself, considering that it is attempting to run drivers intended for a completely different operating system, but anything is possible. ■

Stand Up and Be Counted

With the 2.6 kernel coming out any day (as of this writing) and the hand-off to Andrew Morton complete, Marcelo Tossatti has rethought his handling of the 2.4 kernel, up until now the primary representative of the Linux kernel in the world.

With 2.6 quite usable (even in the final stages of its -test incarnation), Marcelo has decided to bring the 2.4 tree into a deeper freeze. Up until now it had been acceptable to 'back-port' features from 2.6-test back to 2.4, once those features had stablized. Also, some new 2.4 code had been acceptable as well. But now Marcelo is clamping tightly down on any new features, and plans to restrict future 2.4 releases to bug-fixes and security-fixes only, with only very rare true enhancements.

As an example, the libata patches from Jeff Garzik, already looking quite good in 2.6, were rejected in 2.4, whereas they certainly would have been accepted just a few weeks before. At the time of this writing, however, the 2.4 deep freeze was still not completely solid. At first, Marcelo thought he would accept the libata patches, and it was only after some reflection that he changed his mind.

The XFS journaled filesystem found itself in a similar boat, though perhaps with a slightly different outcome. The XFS developers tried to submit a new enhancement to Marcelo, which he naturally rejected due to the deep freeze of the kernel. However, in that case the outcry was quite loud against that decision. After a struggle, Marcelo agreed to consider the patches in this special case, after certain other key developers looked them over.

Whatever the outcome of libata and XFS, it seems clear that Marcelo is attempting to draw a hard line over the addition of new enhancements, and to say firmly that folks wishing to use more advanced features than 2.4 offers, should upgrade to 2.6 instead.

Straight From The Desk

An interesting difference between the 2.4 and 2.6 signal handling code has come to light. Certain signals are "thread synchronous", which means that the thread must deal with them before continuing, and cannot block them.

The difference between 2.4 and 2.6 is that in 2.4, if such a "thread synchronous" signal is blocked, it will just ignore being blocked, and hit the thread regardless. In 2.6, the thread will be killed by that signal. The rationale for this is that the 2.4 behavior tends to hide bugs, since there is no legitimate reason for a program to block a signal that can't be blocked.

In the 2.6 case, the thread is killed, thus alerting the user or developer that there is a problem somewhere in the code. Linus Torvalds considers the 2.6 behavior to be correct; but it is also possible to tell the kernel to revert back to the old 2.4 style, by setting the SA_NODEFER flag to the sigaction() function.