

ZACK'S KERNEL NEWS

Status of Reiser4

The Reiser4 project is still under active development, despite Hans Reiser's murder trial. Eric Hopper recently asked for a status update on the project: specifically, whether there was much chance of the code being merged into the mainline kernel in the near future. Eric got a mixed response. Rik van Riel didn't feel that Reiser4 was on the right track. He pointed out that Namesys had made development decisions on the basis of business needs rather than on what would be best for the kernel, such as the company's decision to include filesystem plugin support in Reiser4 instead of in the Linux VFS. Rik felt that decisions like this, and the general disarray of the project, even before Reiser's murder charge, made it unlikely that Reiser4 would go into the kernel soon.

But Andrew Morton had a different take. The Namesys engineers just hadn't asked for the code to be included re-

cently, which would typically be enough to block any project from inclusion. Also, Andrew pointed out that the kernel developers had not been reviewing the Reiser4 code. Without that technical review to identify the remaining problems, it would be nearly impossible for the Namesys folks to know what to do to bring their code into an acceptable state.

The Namesys engineers also have their own take on the situation. With the future of Namesys far from certain, one question the engineers have been considering is what kind of time commitment they might make if the project became a strictly volunteer effort. Edward Shishkin of Namesys said he plans to devote about a quarter of his working day to the project, whether he gets paid or not. There could be another engineer also interested in some kind of similar time commitment.

Edward also disagreed with Rik's assessment of Reiser4 plugin support. According to Edward, Reiser4 plugins are directly related to that filesystem's disk layout and would not make technical sense in the VFS. By implication, it seems Edward is saying that Rik is wrong about Namesys making any development decisions for business reasons. Edward also explained that the Namesys engineers had not submitted the code for inclusion recently because they still had a lot of stuff on their "to do" list. At this point in the discussion, several folks expressed interest in testing the filesystem and helping out in various ways, and other folks started arguing over the good and bad points of Reiser4, but there was not much indication that any key kernel developers would sud-

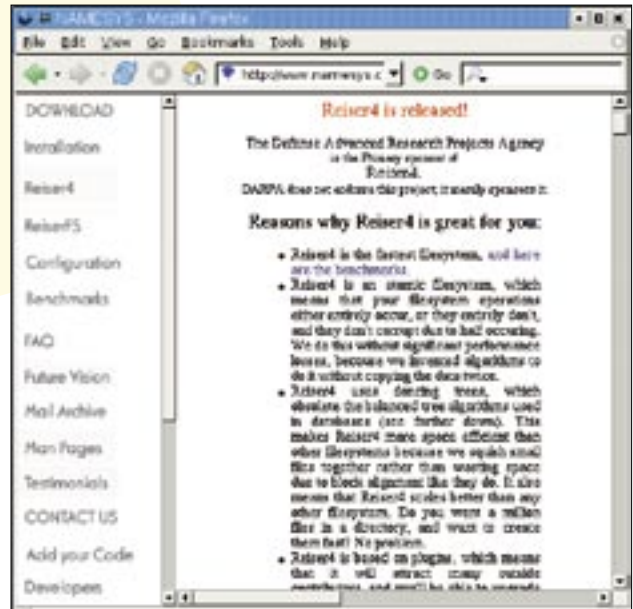


Figure 1: See www.namesys.com for more on the innovative Reiser4 filesystem.

denly take a renewed interest in giving feedback to the Reiser4 folks. Bitter arguments with Reiser in the old days took the enthusiasm out of a number of people who had been very active in offering feedback, and those folks just might not have it in them to get back into it now. But it does at least seem as though there is room for consideration, and Andrew seems willing to accept patches.

Removing the X86_SPEEDSTEP_CENTRINO_ACPI Code

Adrian Bunk has posted a patch to remove the X86_SPEEDSTEP_CENTRINO_ACPI code. This code has been long deprecated in favor of the very similar but better X86_ACPI_CPUFREQ module. The patch was posted before 2.6.21 came out, and Dave Jones felt it was too late to get the code removed in time for that release. Dave suggested not even considering the patch until after 2.6.21 was out the door. Bill Davidsen was also hesitant about the patch, though for different reasons. His thought was that this feature was used on relatively recent computers, so it was important to make sure everyone knew what the alternative was be-

The Linux kernel mailing list comprises the core of Linux development activities. Traffic volumes are immense, often reaching ten thousand messages in a given week, 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 to take on this task is Zack Brown. Our regular monthly column keeps you abreast of the latest discussions and decisions, selected and summarized by Zack. Zack has been publishing a weekly online digest, the Kernel Traffic newsletter for over five years now. Even reading Kernel Traffic alone can be a time consuming task. Linux Magazine now provides you with the quintessence of Linux Kernel activities, straight from the horse's mouth.



fore taking out the code. But Adrian reminded Bill that the code had been deprecated for a while already and had some explanatory text in the *feature-removal-schedule.txt* file. Regardless of these minor objections, it does look like this code will be taken out within the next couple of kernel releases.

Status of Software Suspend

Nigel Cunningham recently made a thoughtful argument to include his suspend2 code in the kernel. He pointed out many ways in which his code was superior to Pavel Machek's swsusp project, in terms of code organization, supported features, ease of use, and solid support from mailing lists and Nigel himself.

Nigel also considered some of the drawbacks of merging suspend2, such as the large size of the patch, the fact that swsusp existed already, and the idea that swsusp could implement some of suspend2's features in the future.

He argued that none of those objections were very serious. As Nigel put it, the case in favor of suspend2 was that

swsusp just didn't solve enough peoples' problems and wasn't going to any time soon, whereas suspend2 existed now and worked well. Initial discussion was sparse. As has been the case for years, Pavel maintains the software suspend portion of the kernel, and he prefers his own implementation. And as John Anthony Kazos Jr. points out, Linus Torvalds doesn't want to have multiple competing versions of software suspend in the kernel.

Linus weighed in on the issue, saying that he was unhappy with the approach the developers were taking, and he was hoping someone new would step in and take the project in an entirely different direction. He felt the current developers were too entrenched in their positions to consider the alternatives, but he did affirm that if a good software suspend solution could be found, he'd support putting it into the kernel. He doesn't feel that this kind of thing is destined to be a user-space problem. Linus also did something typical for him when confronting certain types of problems. He

seems to have thought through some sort of completely different direction that will ultimately (in his view) solve the software suspend problem, and he's now making statements that seem crazy to other kernel people. If Linus stays true to form, a bunch of people will argue with him, then suddenly the basic idea will become clear, someone will run off and write it, and an entirely new software suspend implementation will appear that does everything differently and better.

New FBDev Driver

Alan Hourihane recently posted some code to provide an FBDev driver for Intel's LE80578 chipset. Intel had funded the driver work through Tungsten Graphics, where Alan works. A number of people had technical comments to make about the code, all of which were very minor, and Alan replied quickly with updated patches. It does seem like there are no significant obstacles to getting this code included, and I would guess it will be folded into the next full kernel release.

Advertisement