

Notícias do Kernel

■ Números

De vez em quando alguém se surpreende publicamente com o montante gigantesco de trabalho dispendido no kernel do Linux. Recentemente, Jeff Garzik foi um deles. Com Linus Torvalds usando o BitKeeper, há uma grande oferta de números e estatísticas para se admirar.

Jeff, por curiosidade, foi conferir os números relativos às alterações de código (Changeset) em um único dia – na verdade, o dia anterior ao lançamento oficial da versão 2.6.9 – e encontrou nada menos que 850 delas, além de 3383 revisões. São números impressionantes. Se usarmos outro parâmetro de medida, como por exemplo cada um dos itens do registro de alterações (Change-log) anotados por versão em vez de num espaço de tempo, vemos uma quebra interessante de paradigma, quase uma reviravolta. Ao invés das contribuições e alterações arrefecerem até o congelamento prévio ao lançamento – como era o normal nas séries anteriores – as coisas têm esquentado cada vez mais. A “culpa” recai sobre a mudança na filosofia de desenvolvimento, que praticamente aboliu a diferenciação entre as árvores *estável* e *em desenvolvimento*.

Enquanto os primeiros kernels da série 6 – entre 2.6.0 e 2.6.5 – contaram com uma média de 1100 “remendos” (*patches*) cada um, os quatro seguintes, até o 2.6.9, contam com a impressionante marca de mais de 2500 por cabeça. O número de voluntários que contribuem com o projeto também deu um salto: de 255 até o kernel 2.6.5 para uma média de 390 nos outros 4.

Estimo que o desenvolvimento continuará ativo no 2.6 até que a pedra fundamental do “ramo” 2.7 seja lançada. Nesse momento o ritmo de novas adições e refinamentos na série 2.6 será diminuído, embora nunca paralisado. A estabilidade pura e simples do kernel 2.6 só será alcançada quando a 2.8 já estiver nas ruas. Linus e Andrew Morton parecem considerar indesejável o

longo período de completa estagnação que ocorre após o lançamento oficial de uma série estável. Ao que parece, a grande massa de *kernel hackers* concorda com eles. ■

■ Mais números

Linus Torvalds está sendo apedrejado duramente por suas “lambanças” recentes relativas à numeração de versões do kernel. A versão 2.6.8.1 contém uma porção de penduricalhos considerados inúteis. Sua decisão de lançar uma versão 2.6.9-final seguida de uma versão 2.6.9 radicalmente diferente da “final” também não contribuiu muito para sua imagem. Ambas as decisões inutilizaram uma quantidade enorme de código e scripts de hackers do kernel dos quatro cantos da terra. Como era de se esperar, um número nada desprezível deles protestou – e bem alto! Aproveitando a maré, eles também argüíram contra a distinção arbitrária entre as versões *-pre* (*preview release*) e *-rc* (*release candidate*).

Linus já havia admitido, espontaneamente, seus recentes escorregões, classificando-os como resultado de sua insatisfação com o sistema de atribuição de versões atual. Sendo uma pessoa prática, Linus realmente prefere um sistema de 4 dígitos (como o 2.6.8.1), embora o ache absurdamente feio. Ele testou, em vão, os mecanismos *-pre*, *-rc*, *.1* e *-final*, na esperança de fugir dessa fealdade. Não logrou sucesso na empreitada e decidiu voltar ao sistema antigo, usando o sufixo *-rc* apenas para versões intermediárias. Uma decisão salomônica que nunca agradará a todos mas, ao menos, é consistente.

Para as próximas séries de desenvolvimento, se é que ainda vai existir tal coisa, não haverá mais nenhum *-rc* ou coisa parecida. Esperando que ele se atenha à sua própria decisão, feita em abril de 2002 à época do 2.5.8, haverá apenas as versões 2.7.1, 2.7.2 e assim por diante. Naquela data, Linus afirmou que as séries ímpares, por serem

de desenvolvimento (i.e. não estáveis) são, por definição, *pre-releases*. Como parece que todos mudaram de idéia com relação a isso (inclusive Linus), a forma como o novo modelo de desenvolvimento da série 2.6 causará impacto na 2.7 ainda é uma incógnita. ■

■ Encaixotando Tux

O desenvolvimento da versão do Linux portada para o Xbox, console de jogos da Microsoft, corre a todo o vapor. Esse esforço não seria diferente de nenhum outro projeto de adaptação do sistema para outras plataformas de hardware, não fosse um único detalhe: é impossível iniciar o Xbox com o Linux (ou qualquer outro sistema operacional) sem contornar as restrições impostas pelo hardware. Há outro detalhe perigoso, embora mais comum: portar o Linux para o Xbox poderia ser considerado uma violação ao infame DMCA – Digital Millennium Copyright Act, controversa lei norte-americana de proteção aos direitos autorais e de propriedade.

Ed Schouten já trabalha no projeto há algum tempo. Ele afirma que o esforço de portar o sistema está sendo feito com o único propósito de escrever software interoperável. Isso é permitido pela DMCA, mais precisamente na Seção (Section) 1201(f) – Reverse Engineering exception of the DMCA (Exceção de Engenharia Reversa da DMCA). Se esse argumento vai “colar” em um possível processo judicial movido pela Microsoft é algo que ainda estamos por ver.

Mas mesmo sem a DMCA muitos desenvolvedores torcem o nariz. O fato de ter de alterar o hardware para que o Linux rode no Xbox é um grande argumento contra a iniciativa. Para eles, é como portar o Linux para uma plataforma que não existe. Outros camaradas, particularmente os que lutam na arena dos dispositivos embarcados, bem como os que acham que tudo, até sua geladeira, deve rodar Linux, não vêem problema em nada. Linus Torvalds, comentando sobre o assunto em

julho de 2003, disse que a idéia em si é boa, mas que ele não irá permitir uma árvore oficial do Xbox até que ele seja adotado por um número expressivo de usuários. Uma das razões para essa postura hesitante é a política de "hackear" hardware proprietário. ■

■ Acessibilidade para todos

O Linux já suporta terminais Braille há um bom tempo e recebeu mais de uma dúzia de patches ao longo dos últimos dois anos. Mario Lang, trabalhando no driver *brltty*, recentemente conseguiu conectar um teclado ao seu monitor Braille, que além disso possui uma interface Bluetooth. Dessa forma, criou uma estação de trabalho completamente sem fio. As dificuldades com os *scan-codes* e outros detalhes relativos aos teclados foram obstáculos adicionais durante o processo. No fim, a solução final aproveitou-se do *uinput* para enviar os dados vindos do teclado diretamente do espaço do usuário para o kernel, já previamente configurado.

Enquanto isso, Samuel Thibault colocou seu esforço no terminal Braille Visiobraille (TVB). A implementação final ficou dramaticamente melhor do que os drivers incluídos nos sistemas operacionais da Microsoft. Depois de resolver um problema encardido relativo ao controle da interface serial, Samuel construiu patches para as séries 2.4 e 2.6. Thibault tem trabalhado nessa área há alguns anos. Em 2003 ele desenvolveu uma maneira de permitir que os terminais Braille rolassem a tela para trás - da mesma maneira que os terminais virtuais comumente fazem. Além disso, criou uma espécie de protetor de tela, que limpa o terminal Braille e o mantém em branco mesmo quando o usuário está digitando alguma coisa. Dessa forma, a privacidade dos usuários com deficiências visuais é preservada. Além do trabalho com terminais Braille, Samuel também lida com outros tipos de console e com placas de som. ■

■ Dança das cadeiras

Andries Brouwer decidiu se afastar do cargo de mantenedor dos pacotes *kbd*, *man*, *man-pages* e *util-linux*. Michael Kerrisk apresentou-se para tocar o projeto das *man-pages*, mas os outros pacotes ainda se mantêm sem dono. Andries

continuará a mantê-los até que os substitutos se apresentem. Michael é um "calouro" no desenvolvimento do kernel. Sua única contribuição direta foi um conserto, na versão 2.6.8-rc3, para um erro de estouro por 1 ("off-by-one") no código que verifica quanto tempo de CPU deve ser dado a cada processo.

Veremos muitos patches vindo de Michael, já que seu trabalho mantendo as *man pages* irá demandar um alto grau de fluência com chamadas de sistema, à biblioteca *libc* e várias convenções e aspectos internos do kernel. ■

■ De volta para o futuro

Num caso incomum de "forwardporting", Andi Kleen portou para o 2.6 um recurso útil da série 2.4: os LEDs do teclado piscam quando ocorre um *kernel panic*. Na verdade, as luzes piscantes na versão 2.6 são uma atualização bastante profunda do que tínhamos na 2.4, já que o código evita *#ifdefs* e trabalha com o novo driver de teclado. Ele também não se apóia no falho sincronismo da interrupção do timer, diferentemente da versão antiga.

Infelizmente, Andi não tem interesse em fazer com que os LEDs pisquem em código morse, informando o texto do erro. Isso seria muito útil nos casos em que a falha é grave a ponto de nenhuma mensagem poder ser exibida na tela. Como há muitos maníacos por código morse na comunidade do kernel (mais do que imaginamos...) não se espante se um patch para isso aparecer logo. ■

SOBRE O AUTOR

A lista de discussão *linux-kernel* é o centro do desenvolvimento do kernel Linux. O volume de tráfego é imenso e se manter em dia com todo o processo é uma tarefa humanamente impossível. Uma das poucas pessoas corajosas o suficiente para aceitá-la é Zack Brown, que já publica um "resumão semanal" das discussões, na forma da lista *kernel-traffic*.

Esta coluna mensal manterá você informado sobre as últimas novidades e decisões relativas ao kernel, selecionadas direto da fonte e resumidas pelo próprio Zack.



CD-ROM da Linux Magazine

Não saia de casa sem ele!



- Todo mês um CD-ROM diferente
- Repleto de programas interessantes
- Coletâneas especiais para facilitar a sua vida
- Distribuições Linux prontas para instalar

E no fim do ano, todos os artigos da revista em um único CD-ROM, com máquina de busca para você achar rapidinho tudo o que precisa!

Central de Assinaturas:

Linux New Media do Brasil
 Av. Luís Carlos Berrini, 1500
 Cj. 103 – Brooklin Novo
 São Paulo – SP – Brasil
 Tel.: 0xx11 3345 1002
 Fax: 0xx11 3345 1081

assinaturas@linuxnewmedia.com.br