



Entrevista com Theodore T'so

# Pura diversão

Para a nossa edição de aniversário, a Linux Magazine entrevistou Theodore T'so, o mais antigo desenvolvedor do kernel Linux nos Estados Unidos, hoje trabalhando na IBM. Ele esteve pela primeira vez no Brasil para participar da sexta edição do Fórum Internacional de Software Livre (FISL 6.0) e recebeu a Linux Magazine na sede da IBM em São Paulo.

POR RAFAEL PEREGRINO DA SILVA

**Linux Magazine»** Você poderia fazer um apanhado geral do seu trabalho no Linux e da sua trajetória até o momento atual?

**Theodore T'so»** Bem, o meu trabalho no kernel do Linux teve seu início em setembro de 1991; não me recordo de ninguém mais nos Estados Unidos que estivesse trabalhando com Linux na data. Estávamos na versão 0.09 então e o Linux era somente um hobby para mim naquela época. Eu trabalhava no MIT (*Massachusetts Institute of Technology*) como chefe da equipe de desenvolvimento do Kerberos V5, além de trabalhar também na *Internet Engineering Task Force* (IETF), prestando consultoria em questões de segurança em TI. Nesse período fui participante ativo em diversos grupos de desenvolvimento de padrões abertos para a Internet, tais como *Telnet*, *Privacy Enhanced Mail*, *Common Authentication Technology* e *Networked News Transport Protocol* (NNTP). Até 1999 o Linux continuava a ser um hobby. Foi quando iniciei minhas atividades na empresa VA Linux Systems, atuando como principal engenheiro e membro da equipe sênior de liderança do Grupo de Engenharia de Software, bem como da equipe de desenvolvimento do kernel na

empresa. A VA Linux Systems almejava ser, por assim dizer, a “IBM do Linux”, mas faltava a ela credibilidade junto aos “Global Players” do mercado de TI. Não vamos esquecer, também, que havia uma outra empresa que queria ser a “IBM do Linux”: a IBM! [Risos]. Trabalhei até setembro de 2001 na VA Linux Systems, e comecei em dezembro do mesmo ano na IBM. Nesse intervalo de quase três meses tentei começar a tomar algumas aulas de vôo – minha primeira aula era para ter acontecido no dia 11 de setembro de 2001, mas acabou por ser cancelada por motivos óbvios...

**LM»** Em que se resumem, especificamente, as suas atividades na IBM?

**TT»** Eu trabalho apenas com Linux, mas não somente no desenvolvimento do kernel. Diria que 50% do meu tempo é dedicado ao desenvolvimento do kernel especificamente. Os outros 50% são divididos em diversas outras atividades dentro da empresa, mas principalmente em atividades da equipe *Advanced Linux Response Team*, que é uma espécie de “SWAT” do Linux: ela TEM que dar conta de tudo quanto é problema técnico insolúvel que as outras equipes de suporte e pré-venda encontra-

rem pela frente. Assim, damos assistência à empresa e às suas equipes, de modo *ad hoc*, com o intuito de auxiliar no fechamento de negócios sempre que o que estiver em jogo for Linux rodando em hardware da IBM e o “dedo” de uma equipe de “experts” no assunto for crucial. Isso compreende visitas a clientes para resolver problemas complexos, melhoria no código de aplicativos de clientes, treinamentos e assessoria para CIOs dessas empresas etc.

**LM»** E no desenvolvimento do Linux?

**TT»** Meu trabalho se resume no desenvolvimento de sistemas de arquivos, mais especificamente do *Ext3*, no qual sou responsável pelas ferramentas em espaço de usuário. É fato que esse sistema de arquivos é o mais amplamente utilizado no Linux, apesar de haver outros que são realmente muito bons. Isso ocorre provavelmente porque pessoas de muitas empresas participam do desenvolvimento dele e por ele ser compatível com o *Ext2*, o sistema de arquivos padrão do Linux desde os seus primórdios. Dentro da minha trajetória de trabalho no Linux, implementei e mantive o driver da porta serial do sistema, desenvolvi o sistema de controle de processos e de

gerenciamento de sinais POSIX, além de ter criado as ferramentas de configuração e manutenção para o Ext2. Mas não trabalhou só no desenvolvimento do kernel, não: sou também membro-fundador do *Free Standards Group*, a organização que publica o *Linux Standard Base* (LSB) – o conjunto de especificações que permite que um aplicativo Linux rode em qualquer distribuição Linux que as siga. Também sou responsável pela organização do *Kernel Summit*, evento que reúne os 75 desenvolvedores de maior expressão do kernel desde o ano 2000.

**LM»** *Que novidades podemos esperar para o desenvolvimento do Ext3?*

**TT»** Há várias coisas interessantes sendo feitas no momento: *multiple block allocation path* e *delayed allocation/multiblock allocation* sem extensões, por exemplo, são recursos que auxiliam a reduzir a possibilidade de fragmentação do sistema de arquivos quando vários processos e *threads* estão alocando dados no e do sistema. Isso aumenta também o desempenho – por exemplo, ao escrever todos os metadados do *journaling* do sistema de arquivos de uma vez só. Outra coisa importante é tornar o Ext3 um sistema de arquivos de 64 bits, de modo que ele possa escalar e manipular arquivos gigantes. E, claro, é crucial buscar sempre melhorias de desempenho.

**LM»** *As idéias implementadas no JFS (o sistema de arquivos de alta performance com journaling, desenvolvido pela IBM), são úteis nesse ponto?*

**TT»** Somente no nível da concepção, não da implementação – as tecnologias são simplesmente muito diferentes para que possamos compartilhar algum código.

**LM»** *Ainda no quesito desempenho e escalabilidade, como você encara técnicas como tailmerging (usada para aproveitar melhor o espaço que sobra no disco quan-*

*do se escrevem arquivos cujo tamanho é menor que uma unidade padrão de armazenamento do sistema de arquivos – o inode) ou indexação do sistema de arquivos (recurso utilizado para acelerar a busca e a leitura/escrita de dados do/no sistema de arquivos)?*

**TT»** Esses recursos, embora interessantes, são, por enquanto, de baixa prioridade. Muito mais importante do que isso são coisas como *inline extended attributes*, que o pessoal do projeto SAMBA gostaria de utilizar e que poderiam ser encaixados no espaço que resta em um ou vários *inodes*.

**LM»** *Como você encara a “novela” BitKeeper? Como avalia o impacto disso no desenvolvimento do kernel? (Nota da redação: o BitKeeper era, até recentemente, o sistema de gestão de código fonte proprietário usado no desenvolvimento do Linux. Ele era disponibilizado pela empresa BitMover, de propriedade de Larry McVoy – amigo pessoal de Linus Torvalds – sob uma licença especial para o desenvolvimento do kernel Linux. Essa licença proibia expressamente qualquer tentativa de engenharia reversa do aplicativo. Recentemente, Andrew Tridgell, líder do projeto SAMBA, criou uma ferramenta de interoperabilidade com o BitKeeper, o que levou à cassação da licença de uso do programa para o desenvolvimento do Linux e gerou uma série de conflitos entre os desenvolvedores do sistema do pingüim).*

**TT»** O BitKeeper ajudou no desenvolvimento do Linux de muitas maneiras, sem sombra de dúvida – especialmente no que tange à escalabilidade do controle do desenvolvimento. Ele mostrou que era possível manter esse monstro que é a árvore de diretórios e arquivos do kernel sob controle. Permitiu a Linus e a Andrew Morton, bem como aos mantenedores de outras versões do kernel (ou de sub-projetos dele) aplicar ou rejeitar *patches* (trechos de código com correções de erros

de implementação ou melhorias) de modo muito mais eficiente e rápido. Isso, com certeza, contribuiu para acelerar o processo de desenvolvimento do sistema, além de tornar mais tempo disponível para controle de qualidade. Enfim, tudo isso contribuiu para gerar bastante inovação e acredito que Larry McVoy prestou um bom serviço à comunidade Linux. Acredito que a revogação da licença do programa para uso com Linux era um direito de McVoy, uma vez que ele considerou que a engenharia reversa do Tridgell era uma ameaça para ele.

**LM»** *E a falta dessa ferramenta não está prejudicando (ou prejudicou) o desenvolvimento do Linux?*

**TT»** De maneira marginal. A partir do momento em que a licença de uso do BitKeeper foi revogada, Linus começou a procurar por uma alternativa livre, mas não encontrou. Então ele decidiu desenvolver a sua e nasceu o projeto *git*, no qual ele trabalhou por seis semanas (entre abril e maio deste ano). Houve um certo impacto no desenvolvimento do kernel nessa época, mas foi mais um “solução” do que um atraso – mesmo porque Andrew Morton continuou trabalhando. Assim que o *git* atingiu uma versão que pudesse ser usada com relativa segurança, Linus voltou à ativa e abriu o projeto para a comunidade, disponibilizando-o sob a GPL.

**LM»** *Qual caminho você acredita que o Linux deva percorrer daqui para a frente? De que ainda sente falta no sistema?*

**TT»** Para mim os esforços de desenvolvimento devem se concentrar mais em questões de usabilidade do sistema; em torná-lo mais simples de usar para usuários leigos, por meio da criação de interfaces gráficas mais amigáveis e de boa documentação. E isso não apenas para o computador de mesa, mas também para a configuração de servidores – temos que facilitar a vida dos administradores de sistemas. ■