

# Pluralismo estratégico no desenvolvimento tecnológico: a controvérsia dos “patches ck” e o conflito entre interesses empresariais e coletivos no software livre

Miguel Said Vieira<sup>1</sup>

Embora não seja um campo estritamente ligado ao desenvolvimento científico, o desenvolvimento de software livre se presta a analogias ao modelo de Hugh Lacey sobre a interação entre ciência e valores (LACEY; MARICONDA, 2014). Neste trabalho, aponto relações entre esse modelo e uma controvérsia — a dos *patches* “ck” — ocorrida no desenvolvimento do kernel Linux.

## Escalonamento de processos

A controvérsia envolve o escalonamento de processos — uma função tipicamente desempenhada pelo núcleo de um sistema operacional (como o Linux). Na prática, o escalonamento determina qual processo deve ser executado a cada ciclo do processador; e os algoritmos de escalonamento usam diferentes métodos e critérios para determinar essa sequência, de acordo com (combinações de) diferentes finalidades: a maximização do uso do processador, a priorização de certos tipos de processos, a distribuição equânime de ciclos etc. O escalonamento é bastante complexo, e quando não se realiza de forma adequada pode produzir diversos efeitos indesejáveis, que incluem redução de performance ou responsividade (se o mouse não se move na tela aparentemente no mesmo momento em que é manipulado, p.ex., a experiência do usuário é muito degradada).

## As contribuições de Con Kolivas e sua saída da comunidade

Embora o kernel Linux tenha sido desenvolvido inicialmente para uso em computadores pessoais, ele tem uma relação estreita com a computação científica e corporativa, por ser em certa medida descendente do Unix, uma família de sistemas operacionais bastante usados nesses setores. Talvez em função disso, os sistemas operacionais baseados no Linux sofreram durante bastante tempo com problemas que afetavam particularmente “usuários comuns” (incluindo os de movimentos sociais, do setor público, do Sul Global etc.), que utilizavam Linux em computadores modestos, para fins pouco especializados, e nem sempre possuindo experiência longa com a computação; embora com o passar do tempo parte desses problemas tenham sido superados, eles impactaram negativamente a adoção de Linux entre usuários comuns. A performance inadequada para tarefas de multimídia é um exemplo particularmente relevante desses problemas aqui, pois teve relação direta com a questão do escalonamento: processos ligados a essas tarefas precisam ser executados em frequência constante e tempo adequado, ou o áudio / vídeo ficará picotado ou distorcido.

Os chamados *patches* “ck” foram um conjunto de contribuições ao Linux voltadas a melhorar sua funcionalidade em situações como essa; seu autor, Con Kolivas, é um anestesista que envolveu-se com o desenvolvimento do Linux como programador amador, e chegou a desenvolver um novo algoritmo de escalonamento de processos. Suas principais contribuições, porém, não chegaram a ser incorporadas ao código principal do Linux, e por isso eram mantidas como *patches*: pedaços de

---

<sup>1</sup> Doutor em Educação, professor adjunto na Universidade Federal do ABC, [miguel.vieira@ufabc.edu.br](mailto:miguel.vieira@ufabc.edu.br), <<https://impropriedades.wordpress.com>>.

código que podem ser aplicadas a um programa para modificá-lo, sem que façam parte de seu código original (mas que, por isso, são muito mais difíceis de serem utilizados por usuários finais, e de serem mantidos — uma vez que precisam ser modificados a cada vez que o programa original é atualizado).

As razões pelas quais as contribuições de Kolivas não foram incorporadas são relativamente complexas,<sup>2</sup> mas duas das justificativas apresentadas por lideranças da comunidade à época (como Linus Torvalds, o criador e principal mantenedor do Linux) são relevantes para esta discussão: 1) o entendimento de que era melhor manter um único algoritmo de escalonamento que fosse “bom o suficiente”, do que modularizar o escalonamento: oferecer a possibilidade de escolher diferentes algoritmos, mas correndo o risco de uma queda de qualidade (em função da fragmentação e diluição do desenvolvimento e testagem); e 2) a consideração de que, por ser um programador amador e não ter envolvimento profissional com o desenvolvimento do Linux (ao contrário do que já ocorria com os contribuidores responsáveis pela grande maioria desse desenvolvimento), Con Kolivas não era confiável para continuar mantendo e atualizando as contribuições que ofertava naquele momento. A não incorporação dessas contribuições rendeu à época debates acalorados, e motivou Kolivas a abandonar seu envolvimento mais direto com a comunidade do Linux.

## Relações com a tese do pluralismo estratégico

À primeira vista, pode parecer inadequado analisar esse caso à luz do modelo de Lacey sobre as interações entre ciência e valores: o desenvolvimento de software, ainda mais em contexto não acadêmico, é tipicamente aplicação tecnológica; e com efeito, embora os atores se dedicassem a buscar comparar empiricamente os algoritmos propostos e seus desempenhos, esses esforços não eram particularmente sistemáticos nem precisos (inclusive pela notória dificuldade em realizar testes de escalonamento): o foco era muito mais produzir software funcional, do que construir conhecimento fundamentado.

Por outro lado, também é possível identificar aspectos do caso que se assemelham, como que em escala reduzida, a conceitos propostos por Lacey. Embora de maneira não monolítica, a comunidade de desenvolvimento do Linux apresenta fortes vinculações com o valores do progresso tecnológico (particularmente na vertente *open source* do movimento, que enfatiza a eficiência em detrimento da autonomia),<sup>3</sup> e em termos materiais é cada vez mais vinculada ao capital e ao mercado: a larga maioria dos participantes ativos na comunidade são empregados por grandes corporações, como Google e IBM (STEWART; KHAN; GERMAN, 2020). E ainda que não se possa diferenciar na controvérsia analisada a adoção de estratégias descontextualizadoras / sensíveis ao contexto, é possível enxergar elementos de *estratégias* conflitantes — havia uma diferença clara entre, por exemplo, o que era percebido ou não como um problema digno de investigação no Linux: para contribuidores amadores, sua baixa responsividade em desktops era um problema grave; para desenvolvedores profissionais, não só ela era pouco notada (por terem acesso a computadores mais avançados), como seria um *trade off* aceitável caso o escalonamento garantisse eficiência em servidores (como aqueles das grandes empresas financeiras e industriais, que indiretamente remuneram os desenvolvedores profissionais). Por fim, o ponto mais crucial de contato talvez seja com a tese do pluralismo estratégico: a escolha por não modularizar o algoritmo de escalonamento

---

2 Essa controvérsia é apresentada com mais detalhes em Vieira (2014, 2019).

3 Sobre as duas vertentes do movimento software livre, ver Evangelista (2010).

literalmente determinou a exclusividade de uma das abordagens que estavam em discussão, e na prática inviabilizou o desenvolvimento e testagem da outra.

Tomadas em conjunto, estas relações sugerem que o modelo de Lacey talvez seja relevante para analisar áreas mais diretamente ligadas ao desenvolvimento tecnológico (que à pesquisa científica), e a relevância de investigar eventuais adaptações necessárias para tanto.

## Referências bibliográficas

EVANGELISTA, Rafael de Almeida. *Traidores do movimento: política, cultura, ideologia e trabalho no Software Livre*. 2010. Tese de doutorado – Unicamp, 2010. Disponível em: <<http://repositorio.unicamp.br/jspui/handle/REPOSIP/280201>>.

LACEY, Hugh; MARICONDA, Pablo Rubén. O modelo das interações entre as atividades científicas e os valores. *Scientiae Studia*, v. 12, n. 4, p. 643–668, dez. 2014. Disponível em: <[http://www.scielo.br/scielo.php?script=sci\\_abstract&pid=S1678-31662014000400643&lng=pt&nrm=iso&tlng=pt](http://www.scielo.br/scielo.php?script=sci_abstract&pid=S1678-31662014000400643&lng=pt&nrm=iso&tlng=pt)>.

STEWART, Kate; KHAN, Shuah; GERMAN, Daniel M. *Linux Kernel History Report 2020*. . [S.l.]: The Linux Foundation, 2020. Disponível em: <[https://project.linuxfoundation.org/hubfs/Reports/2020\\_kernel\\_history\\_report\\_082720.pdf?hsLang=en](https://project.linuxfoundation.org/hubfs/Reports/2020_kernel_history_report_082720.pdf?hsLang=en)>.

VIEIRA, Miguel Said. *Os bens comuns intelectuais e a mercantilização*. 2014. 365 f. Tese de doutorado – Faculdade de Educação da Universidade de São Paulo, São Paulo, 2014. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/48/48134/tde-01102014-104738/>>.

VIEIRA, Miguel Said. Software livre, mas para quem? O conflito entre interesses empresariais e coletivos na controvérsia dos “patches ck”. In: ESOCITE.BR, 2019, Belo Horizonte. Disponível em: <<https://www.esocite8.cefetmg.br/anais-2/>>.