Patentes de Software para não-advogados (V1.02)

Paulo Trezentos (ISCTE / ADETTI / ANSOL)
Paulo.Trezentos@iscte.pt

12 de Abril de 2003

Contribuições

- Versão 1.01 - Contribuições de João Neves (ANSOL).
- Versão 1.02 - Contribuições de João Neves (ANSOL).

1 Breve Justificação

As patentes de software são um tema que estão a levantar alguma polémica na Europa em virtude da proposta de directiva apresentada pela Comissão Europeia (em 2002).

Este documento não pretende ser uma explicação exaustiva e aprofundada do tema, mas tão só uma abordagem sistemática e simples do que realmente pode implicar para todos nós se esta directiva for aprovada e, posteriormente, transposta para legislação nacional.

A nível português, a ANSOL (Associação Nacional para o Software Livre) tem tido reuniões com vários eurodeputados tentando sensibilizá-los para a questão. Alguns dos contactos verificaram-se positivos e tiveram repercussões directas.

2 História e Conceito

O gabinete americano de patentes (USPTO) começou a aceitar patentes sobre software há largos anos. Isto significa que se uma empresa achar que um seu *software* incorpora conceitos novos pode patentear esse mesmo software. Há que esclarecer que quando falamos em *software* não nos estamos a referir elementos mecânicos que interagem com as forças da Natureza, mas à mera implementação numa linguagem de programação de ideias que envolvam estruturas lógicas.

Na **Europa**, o *software* não é patenteável. Em termos legais, o Art°52 do EPC (*European Patent Convention*), revisto em 2000/11, diz claramente que não se pode patentear programas de computadores [1].

Apesar de legalmente não ser possível patentear software, o organismo europeu responsável pelas patentes (European Patent Office - EPO) estima em 30.000 o número de patentes que estão à margem da lei, pois são classificáveis como de software. A principal razão para este comportamento do EPO resulta da influência do lóbi das 10 maiores empresas informáticas mundiais, que só por si são responsáveis por uma grande parte desse grupo de patentes.

Esse grupo de empresas tem um nome, *Business Software Alliance* (BSA). A BSA é constituída pelas maiores empresas mundiais da área informática, como a Microsoft, IBM e Adobe.

A actual forma de actuação do EPO, provocou uma "incerteza jurídica", quer no registo (patentes registadas pelo EPO tinham sido inicialmente recusadas nos respectivos gabinetes nacionais), quer na invalidação de patentes (os tribunais decidiram de forma diferentes em relação à validade de patentes parecidas em países diferentes). Esta situação levou a Comissão Europeia, sob pressão da BSA, a propôr ao Parlamento Europeu uma directiva em que as patentes seriam estendidas ao software de forma a acabar com a "incerteza jurídica".

A proposta encontra-se em processo de discussão no Parlamento Europeu e a votação em plenário está prevista para 3 de Junho de 2003.

3 Conceito: Patente de Software

Definir o conceito de "patente de software" não é tão claro como se podia supor porque a própria definição de software pode ser entendida de diferentes formas.

Mesmo assim, existem pontos de vista que são partilhados por ambas as partes.

Por exemplo, ideias matemáticas e formas de cálculo são pacificamente aceites como não-patenteáveis. Se eu definir um novo conjunto de números que possuem uma característica especial (como todos os números que dão resto 1 quando divididos por 300) não os posso patentear e exigir que me paguem uma licença para utilizar esse conjunto de números.

Simultaneamente, também é aceite pelos opositores às patentes de software que uma inovação de carácter técnico [2], que interaja com as forças da

[1] *The following in particular shall not be regarded as inventions within the meaning of paragraph 1: ... 3– schemes, rules and methods for performing mental acts, playing games or doing business, and programs for computers.*

[2] Carácter técnico com um sentido físico, material e empírico adjacente.

natureza e tenha um efeito directo explicável pelas leis da física, pode ser patenteável.

O desacordo centra-se na ideia de que ideias, processos de negócio e programas de computador, enquanto sequências de instruções, possam ser patenteáveis.

4 Porquê discordar das patentes de software?

As razões podem variar consoante a nossa sensibilidade política, ideológica e de mercado. Algumas das seguintes justificações encontram-se entre as mais apontadas:

1. O software é **imaterial**. Assim sendo, é mais fácil a sua construção / implementação do que de bens físicos susceptíveis de desgaste e que podem exigir ferramentas complexas para os manipular. Se é mais fácil, porque será que devemos permitir patentes que ditam o mesmo tempo de protecção do que às patentes clássicas? A razão entre investimento e protecção (do retorno financeiro dos investimentos em inovação) é muito mais favorável a quem patenteia o software, ficando o mercado e a sociedade a perder com esse desequilíbrios.

2. Nenhum software é construído inteiramente do zero. Pelo contrário, é desenvolvido recorrendo a **blocos**, como numa casa de Lego, já existentes e que em conjunto com novos "blocos" lógicos originais, formam uma nova aplicação de *software*. Assim, ao patentear-se um pequeno "bloco", está a evitar-se que surjam um conjunto mais alargado de aplicações que pudessem vir a utilizar, mesmo que residual mente, esse "bloco". E como se pode ter a certeza que determinado bloco já está patenteado quando se desenvolve software? E como garantir que quando alguém pretende patentear um bloco, esse bloco não é feito utilizando outros blocos já patenteados? Toda a ciência é incremental[3], mas a indústria de software é excepcionalmente propícia a que tal aconteça derivado da facilidade de reutilização de componentes. Isto é notório se compararmos com com a indústria farmacêutica. Um novo comprimido para as dores de cabeça tem um número limitado de elementos (blocos) que o constituem, ao contrário de um software constituído por milhões de linhas de código.

3. Se não existirem patentes de software deixará de haver **inovação**? Julgamos que não. As novas ideias (blocos) nascem durante a construção de sistemas mais complexos e a sua divulgação é feita para que a mesma possa ser validada pela comunidade, por brio profissional ou para criar notoriedade em redor do individuo / empresa. Ninguém

[3] "If I have seen further it is by standing on the shoulders of giants", Isac Newton

deixará de ter uma nova ideia ou não divulgá-la só porque não pode a patentear...

4. As patentes de software irão **atrasar** o ciclo de desenvolvimento. Se na construção do software for necessário acautelar se se está a infringir uma patente já existente com os blocos que se utilizam, isso obrigará a um atraso do desenvolvimento / programação. Por outro lado, se existir durante a fase de desenvolvimento uma nova ideia que se julgue original, tem de se perder tempo em defini-la, caracterizá-la e proceder aos testes necessários para o processo de a patentear.

5. As patentes de software irão tornar mais **caro** o desenvolvimento de aplicações. Pelo que foi dito anteriormente, para desenvolver-se uma aplicação com um grau de complexidade razoável tem de se ter um departamento jurídico que trabalhe nas seguintes questões: a) já existem patentes sobre as tecnologias que utilizamos? b) as nossas novas ideias são patenteáveis? c) empresa HAL está a utilizar uma ideia semelhante à nossa. Podemos processá-la? d) empresa HAL processou-nos por usar uma tecnologia supostamente deles. Podem defender-nos? e) empresa HAL tem uma tecnologia que gostaríamos de utilizar e nós temos uma tecnologia que eles precisam. Podem tratar de fazer um licenciamento cruzado? Naturalmente, este departamento vai pesar na estrutura de custos e encarecer o software para o utilizador final[4].

6. O trabalho de que desenvolve software está sempre protegido através do **direito de autor**. Assim, a questão da permissividade em relação utilização ilegítima de software não licenciado (vulgo *pirataria*) não se aplica às patentes de software mas aos mecanismos legais de protecção do direito de autor.

7. As patentes não permitem **recuperar** do verdadeiro investimento do software. Um conceito na área de software é facilmente demonstrável, mas dificilmente reutilizado. Por exemplo, mostrar uma lista dos ficheiros que estão num meio físico com uma disquete em MS Windows não ajuda alguém que queira apresentar uma lista de ficheiros de um CD noutro sistema operativo. Todos os passos têm de ser reimplementados devido às diferenças de mecanismos físicos, forma de acesso, formato de dados, interfaces de sistema operativo, etc. É aqui que é feito o verdadeiro investimento no software, nestas reimplementações que trazem resultados úteis.

[4]Alguns números interessantes: as empresas americanas gastam entre 25% a 35% do seu orçamento para Investigação e Desenvolvimento em juristas (Jordan Greenhall, CEO da Divx Networks, FTC 2002). Cerca de 90% da população activa em Washington é formada em direito.

Apesar de não deverem ser consideradas como razões justificativas, existem outros argumentos válidos do **ponto de vista prático**:

1. O EPO (gabinete de patentes europeu) não tem **capacidade técnica** para validar se uma patente de software é não-óbvia, não-trivial ou inovadora. Face ao avançado estado das inúmeras áreas das ciências da computação (Inteligência Artificial, Redes, Sistemas Informação, HPC, etc...) seria necessário ter um alargado grupo de peritos em cada área para poder validar todas as propostas apresentadas.

2. Sendo actualmente o **custo estimado** de uma patente orçado em 38.000 Euros (variando consoante algumas variáveis) é muito complicado a uma pequena e média empresa (PME) poder suportá-lo. E um programador independente?

3. Actualmente na Europa existe uma directiva da Comissão que pretende definir as **linhas estratégicas** para o desenvolvimento das Tecnologias de Informação: eEurope 2005. Algumas das propostas dessa directiva, apontam o software livre como a maior oportunidade para a Europa evitar alguma da dependência política e económica das empresas americanas de TIs, fazer frente ao ciberterrorismo e ganhar vantagens competitivas em termos empresariais. Apesar do conceito de Software Livre ser ortogonal às patentes de software, concorde-se que não faz muito sentido patentear uma coisa que se pretende que seja... utilizada por todos sem custos de licenciamento.

4. O EPO **ganha** por patente aceite. Logo, quanto mais patentes de software forem aceites, maior o orçamento dessa instituição. Isso não constituiria um argumento forte se existisse um orgão de controlo democrático sobre o mesmo. A prática dos últimos anos (em que o EPO patenteia à margem da lei) mostra que tal não acontece e que esses mecanismos precisam de ser definidos.

5. O **ciclo de inovação** na indústria de software é de 3 a 5 anos. O tratado TRIPS, de que Portugal é signatário, define 20 anos como a duração de uma patente.

6. As patentes são facilmente utilizadas para **evasão fiscal**. Como em muitas economias os lucros resultantes de investimentos em patentes não são tributáveis durante vários anos, as empresas tendem a utilizar esses incentivos fiscais de forma ilegítima para lucros não derivados de nenhuma patente.

5 Ai Portugal Portugal

Ai Portugal Portugal
De que é que estás á espera
tens o pé numa galera outro no fundo do mar
Jorge Palma

As patentes de software irão pesar especialmente para economias como a portuguesa em que as empresas que produzem software (as *software houses*) têm à escala mundial uma **dimensão** muito reduzida.

A sua eficácia no registo de patentes de software vai ser muito reduzido comparado com as grandes empresas americanas que, para além de maiores recursos, possuem uma tradição legalista.

A própria **história** da Sun Microsystems, hoje uma das maiores empresas americanas, atesta os perigos das patentes de software. Gary Reback, que na década de 80 foi advogado da Sun, descreve que um dia foram processados pela IBM que os acusava de estarem a infringir 7 patentes[5]. Para discutir esse assunto, combinaram uma reunião onde inesperadamente apareceram 14 advogados representando a IBM. Estes expuseram as acusações que pretendiam apresentar.

Os advogados e técnicos da Sun responderam apresentando argumentos demonstrativo que as acusações eram infundados e apenas uma delas era passível de ser levada a tribunal. O chefe dos 14 advogados então disse: "Está bem. Talvez não tenham infringido estas 7, mas a IBM tem mais 10.000 patentes. Querem mesmo que voltemos para a sede da empresa e achemos outras 7 patentes que tenham de facto infringido ou preferem pagar-nos logo os 20 milhões de dólares?".

No momento em que uma empresa portuguesa tenha um pouco mais de êxito, arrisca-se a que um caso como o anterior lhe aconteça.

Do mesmo lado estão as próprias universidades e centros de I & D. As parcerias com entidades empresariais que possam rentabilizar as patentes são menores do que o mercado americano, funcionando apenas como **obstáculo** à inovação e não como incentivo.

6 Próximos passos

O procedimento legislativo passa pela aprovação da directiva pelo Parlamento Europeu em co-decisão com o Conselho da Europa.

[5] Uma delas era relativa à técnica de transformar uma linha fina numa linha mais grossa. A técnica pode resumir-se a indicar os 4 pontos nos extremos e pintar o rectângulo por eles definidos.

Antes que o Parlamento proceda às votações, a Comissão de assuntos jurídicos pode apontar emendas à directiva inicial.

A CULT e a ITRE já deram os seus pareceres. Falta a JURI decidir as que são aceites para a versão final a apresentar ao plenário. Os documentos estão na lista dos documentos de trabalho da JURI da reunião de 22/4.

O facto muito citado que a Europa deve **harmonizar** com os EUA estas questões de forma a facilitar a assinatura de acordos internacionais deve ser equacionada de outra forma. Porque não os EUA harmonizarem as suas leis de patentes com as nossas? Em audições no Congresso americano muitas empresas opuseram-se às patentes, bem como reputados cientistas da áreas tecnológica e económica.

Pelo atrás exposto urge que tomemos as iniciativas que estão ao nosso alcance.

Ao nosso alcance, primeiro enquanto cidadãos. Nesse mesmo sentido, podemos aderir a várias petições que se encontram disponíveis, sensibilizar os nossos políticos e exigir-lhes uma atitude responsável e consistente com os interesses que representam.

Apesar do reduzido peso de Portugal nas instâncias europeias quando comparado com os 5 grandes países, existem atitudes que podem marcar a diferença. Por exemplo, houve um Eurodeputado português que se absteve no ma votação da Comissão de Cultura (CULT) do Parlamento Europeu quando todo o grupo europeu a que pertencia votou contra. Essa atitude pode não só valer pelo seu voto em si, mas pelas legítimas dúvidas que levanta em relação aos interesses que movem esta directiva.

7 Um caso prático

Um exemplo que posso dar é um que conheço bem: o Linux Caixa Mágica [6].

O Linux Caixa Mágica é um conjunto de *programas informáticos* que permitem a fácil instalação, utilização e configuração do sistema operativo Linux e suas aplicações.

Este software tem sido desenvolvido nos últimos dois anos no contexto de um centro de investigação sem fins lucrativos, a ADETTI. Para a dimensão portuguesa, falamos de um centro de médias dimensões que envolve cerca de 50 investigadores e que foi responsável por 10% dos projectos portugueses do 5º Programa Quadro (na área das tecnologias de informação - IST).

A construção dos diferentes programas que permitem a instalação e configuração foram feitos à custa de outros **blocos já existentes**. Por exemplo: à custa de um sistema de empacotamento de programas (RPM), de um sis-

[6] http://www.caixamagica.pt

tema de ficheiros (Ext2), de um sistema de particionar o disco (Parted), de um sistema de desenho de janelas (Gtk+), etc..

Paralelamente, desenvolvemos nós próprios um sistema de extracção de informação a partir de ficheiros XML que disponibilizamos a todos que o desejem utilizar.

Se a directiva da Comissão já estivesse em vigor, provavelmente muitos dos blocos que nos ajudaram a construir todo o sistema não estariam disponíveis para os podemos utilizar. Mais, provavelmente até alguns dos blocos originais que desenvolvemos poderiam já ter sub-componentes patenteados e não poderíamos então implementá-lo.

Por outro lado, cada vez que nos fosse colocada a questão de adicionar mais uma nova funcionalidade teríamos de ter muito cuidado em tentar verificar se estaria sujeito a patentes.

Os mais cépticos poderão talvez defender que podíamos nós próprios patentear ideias que surgissem no desenvolvimento. Quem defender isso, não conhece certamente a realidade da investigação em Portugal. Hoje em dia, já tem de se passar 30% do tempo total angariando financiamentos ou a tratar de procedimentos administrativos, pelo que passar outros 30% a (tentar) lidar com patentes de software não é uma bonita perspectiva.

Com segurança posso afirmar que o software a que me referia, e de que foram distribuídas 50.000 cópias, que está presente em escolas, universidades, empresas e estações de trabalho de particulares, nunca chegaria a ver a luz do dia.

Como ele, muitos poderão no futuro nunca chegar a ser concretizados. E em nome de quê?

Referências

[Aig] Philippe Aigrain. 11 questions on software patentability issues in europe and the us - http://cip.umd.edu/aigrain.htm.

[dCE02] Comissão das Comunidades Europeias. Proposta de directiva à patenteabilidade dos inventos que implicam programas de computador, http://paulo.trezentos.gul.pt/articles/patentes/patentescomissao.pdf, February 2002.

[faFII] Foundation for a Free Information Infrastructure. Proposta alternativa de directiva à patenteabilidade dos inventos que implicam programas de computador,http://swpat.ffii.org/.

[Sta02] Richard Stallman. The danger of software patents , http://paulo.trezentos.gul.pt/articles/patentes/stallmanrichard.pdf, 2002.