

## 21. SOFTWARES LIVRES E O “NÓS” DO AGIR EM ECONOMIA DE MERCADO: COMO OS DIREITOS AUTORAIS RESTRINGEM A LIBERDADE SOCIAL

**Marcos Vinício Chein Feres**

**Jordan Vinícius de Oliveira**

**Illanah Vieira Falconiere**

**Palavras-chave:** código fonte, softwares livres, liberdade social.

O presente resumo visa investigar diante das interações contemporâneas entre o Direito e a Tecnologia, se a restrição do código-fonte do software, através do Copyright, é um elemento imprescindível no estabelecimento da proteção autoral.

Adota-se como referencial teórico a concepção do “nós” do agir em economia de mercado trabalhada por Axel Honneth (2014), em sua obra “O direito da liberdade”. Segundo Honneth, nas últimas décadas a economia capitalista assumiu uma forma social oposta à institucionalização da liberdade social. Tal assertiva se deve ao fato de que nesse novo sistema de ação os indivíduos não possuem uma relação prévia de reconhecimento recíproco, o que deturpa um dos pré-requisitos fundamentais para o funcionamento do livre mercado: a liberdade.

Ao constatar que só existe liberdade com a anuência do outro como indivíduo de direito, contrapõem-se disposições individuais e irrestritas de liberdade - liberdade negativa, à necessidade de uma liberdade que complemente, mas não restrinja o interesse de cooperação com os demais indivíduos, a liberdade social (HONNETH, 2014).

Quanto à metodologia, realiza-se uma análise empírico-qualitativa de conteúdo. Estabelecem-se hipóteses iniciais decorrentes do amplo estudo do material aferido sobre o tema, contrapõem-se este material à realidade fática, para redimensionar o objeto estudado por meio de indagações frente ao ideal normativo (BABBIE, 2007).

Entende-se como hipótese que a proteção jurídica e o devido crédito ao autor ou detentor do software pode ser assegurada sem prejuízo da disseminação do saber empregado na criação do programa de computador.

Para analisar esta assertiva é necessário o prévio estabelecimento do conceito de software. O conceito que será utilizado é o de Pressman (2010), que reconhece o software como um sistema composto não por elementos físicos, mas lógicos, dotado de características peculiares. Assim, o software se difere de um componente físico (*hardware*), na medida em que é fruto de um característico processo de desenvolvimento, não se desgasta e é feito sob medida.

Os softwares possuem uma especificidade em relação às propriedades artísticas e literárias, também protegidas pelo direito autoral: o chamado código-fonte. Este pode ser aberto ou fechado, conforme opção do programador/empresa, sendo que o fechado restringe o acesso ao código, disponibilizando-o apenas para os detentores de seu direito, enquanto o aberto, decorre da opção dos proprietários de tornar livre o acesso ao código para quaisquer interessados (MARENGO; VEZZOSO, 2006).

Para tanto, é preciso entender o marco jurídico brasileiro no qual se estrutura a complexa dinâmica dos softwares para verificar sua adequabilidade. A Lei 9.609/98, que rege a propriedade intelectual de programas de computador nacionalmente, dispõe no seu artigo 6º sobre a proteção jurídica ao autor do software e sua exclusividade, enumerando-se nos incisos as hipóteses que não violam o controle sobre o software, como a reprodução de um único exemplar.

Propõe-se aqui uma interpretação legislativa que aborde o dispositivo não de maneira tecnicista, mas por via de um enfoque crítico, diante do arcabouço teórico adotado. A partir da ideia da liberdade social, infere-se que a legislação, como produto de um mecanismo institucional, deveria tutelar os processos de regulação de interesses e de consolidação jurídica da igualdade de oportunidades (HONNETH, 2015). Contudo, verifica-se que o artigo 6º reforça a proteção de uma liberdade negativa do proprietário, como assegurada nos seus incisos.

Desta maneira, o monopólio legal sobre o software apresenta-se juridicamente insatisfatório, sendo que restringe, também no cenário econômico e tecnológico, o conhecimento. No artigo 2º da referida lei se estabelece que o regime

de proteção aos direitos do autor e registros é o mesmo que o aplicado para a legislação de direitos autorais. Isso significa que o legislador brasileiro expressamente concede a proteção destinada às obras literárias, o regime de direito autoral, aos softwares.

Tal peculiaridade é abordada por Marengo e Vezzoso (2006), pela ótica da engenharia reversa, que consiste no processo de obtenção do know-how, quer dizer, como um bem é desenvolvido, quais problemas técnicos enfrenta e quais são suas funcionalidades. Quando aplicada a engenharia reversa em obras literárias é possível compreender como elas foram criadas. Ao ler um livro, por exemplo, a estrutura e os recursos utilizados são cognoscíveis, do material das páginas até as técnicas de narrativa e estrutura textual. Esse processo foi estudado e reutilizado em novas obras, gerando aprendizado e evolução. Assim, a contribuição do conhecimento e de cultura perpassa a engenharia reversa, com a colaboração de outras mentes (MARENGO;VEZZOSO, 2006).

O objetivo da engenharia reversa em um software não é o da produção de mera cópia, mas o do aprendizado das técnicas de programação, soluções específicas a problemas técnicos, funcionamento e modificação de serviços, entre outros aspectos.

Todavia, a logística dos direitos autorais transportada aos softwares protege o código-fonte, restringindo a compreensão do *know-how* numa versão que é fechada, o que é paradoxal pois o copyright restringe estruturas cujos processos por ele protegidos sempre foram transparentes. Ao excetuar esta regra ao software, tal proteção demonstra-se contraditória, visto a diferença na codificação entre obras literárias e softwares.

Considerando a lógica da liberdade social, infere-se que a proteção autoral de softwares fechados impossibilita a engenharia reversa, de modo que, apenas os detentores do direito de propriedade autoral podem reter o conhecimento ali empregado, privando-o da comunidade. Logo, a condução da integração do mercado, fundamentada pela compreensão de uma liberdade comunicativa, não atinge seu ápice, já que não propicia o enriquecimento da própria comunidade (HONNETH,2015).

Após o estudo de questões levantadas parte-se à análise jurídica das licenças aplicáveis aos softwares livres. Existem várias licenças de Software livre, entre elas estão a General PublicLicense (GPL) e a Lesser General PublicLicense (LGPL). A diferença principal entre essas é que a GPL, difundida pelo programa GNU/Linux, por exemplo, não permite que o software livre seja incorporado por programas proprietários, garantindo sua difusão apenas por softwares livres. Já a LGPL é utilizada para ligar o software livre a aplicações proprietárias. Segundo a Free Software Foundation, as licenças se estruturam conforme a necessidade do usuário, de modo que o caráter adaptativo delas é uma constante. Elas diferem não só quanto ao objetivo, mas também quanto à distribuição do código fonte, podendo ser classificadas como licenças compatíveis ou incompatíveis com a GPL (FREE SOFTWARE FOUNDATION, 2016)

A liberdade, portanto, se garante na distribuição do software livre, não significando ausência de preço. Decerto estabelece-se de diversos modos, como quando o programador recebe o código-fonte ou na disponibilidade de obtê-lo, caso escolha realizar mudanças num software livre, ou utiliza pedaços deste para a elaboração de novos programas.

A obrigação jurídica que as licenças copyleft impõem é que o usuário respeite a liberdade dos demais ao distribuir cópias, modificar ou criar um software a partir do já existente. Assim, o código aberto ou os meios de obtê-lo devem estar sempre disponíveis. A proteção jurídica de um software livre é concedida por duas etapas: a certificação de que o software está protegido por copyright e o oferecimento da licença ao usuário, indiscriminadamente, concedendo permissão para cópia, distribuição ou modificação (FREE SOFTWARE FOUNDATION, 2016).

Infere-se, portanto, que licenças como a GPL refletem a possibilidade de reconhecimento do autor e de benefício à comunidade pela disponibilização do conhecimento empregado sem prejuízo do seu devido crédito, viabilizando ao consumidor real liberdade de escolha.

Ao interligar tal lição à lógica dos softwares livres, identifica-se que tais meios jurídicos de distribuição não obstam o acesso e a livre cognição dos usuários. Eles reconhecem o software como um componente conjuntamente desenvolvido e o protegem de forma que este não saia do âmbito coletivo. Assim, aos demais usuários,

programadores e à comunidade não é negado aquele conhecimento de que fazem parte, como ocorre com a proteção autoral vigente no Brasil.

## REFERÊNCIAS

BABBIE, Earl. *The Practice of Social Research*. Eleventh Edition. Belmont : Thomson Wadsworth, 2007.

EPSTEIN, Lee. KING, Gary. Pesquisa empírica em direito [livro eletrônico] :as regras de inferência. São Paulo : Direito GV, 2013 (Coleção acadêmica livre) 7 Mb. ; Título original: *The rules of inference*. - Vários tradutores.

MARENGO, Luigi.; VEZZOSO, Simonetta. *Dynamic Inefficiencies of Intellectual Property Rights from an Evolutionary/Problem-Solving Perspective: Some Insights on Computer Software and Reverse Engineering*. Social Science Research Network, jun. 2006. Disponível em <<http://ssrn.com/abstract=1358920>>. Acesso em: 15 nov. 2013.

FREE SOFTWARE FOUNDATION. *Various Licenses and Comments About Them*. 03 de novembro de 2016. Disponível em: <<https://www.gnu.org/licenses/license-list.html#SoftwareLicense>>. Acesso em: 13 nov. 2016.

PRESSMAN, Roger. *Software Engineering: a Practitioner's Approach*. Seventh Edition. New York: McGraw-Hill, 2010.

HONNETH, Axel . *O Direito da Liberdade*, São Paulo: Martins Editora Livraria Ltda, 2015.