Improving interoperability by encouraging the sharing of interface specifications

Dr Sally Weston, Senior Principal Academic, Centre for Intellectual Property Policy and Management 'CIPPM' Bournemouth University, England, sweston@bournemouth.ac.uk

Keywords (6)

Interoperability, software interfaces, 3D CAD, standards, reverse engineering

Abstract

3D CAD software is vital to record design information. The industry is oligopolistic and despite standards has all the elements associated with a lack of interoperability, namely proprietary software, network effects and lock-in. Interfaces are similar to standards and their indirect effect amplifies their impact and value and distorts the intended intellectual property protection. The distributed machine code is not readable and the restrictions on reverse engineering are tantamount to making the information a statutory trade secret.

The regulation of interoperability is a balancing act between control by rightsholders and openness of interfaces. Identifying the 'pivot' point must take account of the software's functional nature and data integrity. Existing proposals are evaluated and recommendations with least intervention that encourage market solutions are made. These involve a modest and doctrinally appropriate amendment to Article 6 of the Software Directive to legitimise the sharing of interface specifications obtained by decompilation and encourage the use of a public register to improve dissemination.

1. Introduction

Software interoperability[1] is considered to promote socially desirable goals and public benefit.[2] Interoperability encourages more use of resources and more competition which should stimulate innovation[3] which is likely to be of the 'follow on' type rather than 'breakthroughs'.[4] While there is no systematic body of empirical evidence of a link between interoperability, and competition and innovation, the claim is often supported by illustrative examples.[5]

The adoption of standards, regulation and market pressure has aided interoperability but important challenges remain in many areas including health care systems, cloud computing and 3D Computer Aided Design 'CAD' software.[6] 3D CAD software is crucial to the economy as it records vital design information and knowhow on all engineered products in the developed and developing world. Another vital role is facilitating rapid innovation, which enables the development of sophisticated products. The 3D CAD industry is made up of four main suppliers who are profitable and successful – Siemens, Autodesk, Dassault Systemes and Parametric Technologies – in an oligopolistic market.[7]

---

[1] The Software Directive considers interoperability to be the functional interconnection and interaction between elements of software and hardware and 'the ability to exchange information and mutually to use the information which has been exchanged.' Council Directive 2009/24/EC on the legal protection of computer programs ('Software Directive' or 'Directive' as case requires) [2009] OJ L111/16, recital 10. Interoperability requires two or more programs to exchange and use information. It does not require the programs to use the same code or perform identical or similar functions, but they must be able to exchange and use essential information. The exchange of information between programs takes place through interfaces which can take various forms: application programming interfaces (APIs), protocols, and data file formats.

[2] Pamela Samuelson, 'Are Patents on Interfaces Impeding Interoperability?' (2009) 93 *Minnesota Law Review* 1943; Oracle America Inc. v Google Inc., Brief of Amici Curiae Intellectual Property Professors in Support of the Defendant-Cross Appellant and Affirmance 30 May 2013; Urs Gasser and John Palfrey, 'When and how interoperability drives innovation' (Berkman Publication Series, November 2007) http://cyber.law.harvard.edu/sites/cyber.law.harvard.edu/files/interop-breaking-barriers_1.pdf (accessed 15 October 2015).

[3] Commentators including Mark Lemley, 'Antitrust and the Internet Standardization Problem' (1996) 28 *Connecticut Law Review* 1041, recognise the benefits of interoperability while others consider the position is more ambiguous, for example Mario Gil-Moto, 'Economic aspects of the Microsoft case: networks, interoperability and competition' in Luca Rubini (ed), '*Microsoft on Trial'* (Edward Elgar, Cheltenham, 2010) 344, 359 *et seq.*

[4] Follow on innovation is dynamic rather than static competition, for example coming within the description of dynamic competition advocated by Gregory Sidak and David Teece, 'Dynamic competition in antitrust law' (2009) 5(4) *Journal of Competition Law and Economics* 581, 600 *et seq*.

[5] Gasser (n 2).

[6] Lack of interoperability is not limited to software industries but includes all products which have a software component. Industries that have a need for software to be compatible include cars, traffic control systems, construction, defence and many more.

[7] Siemens PLM, Autodesk, Dassault Systemes and Parametric Technologies 'PTC' represent the 'high-end' and 'middle range' suppliers of 3D CAD software. An oligopolistic market refers to a market structure with a limited number of sizeable firms. Because the behaviour of one firm has an appreciable impact on the overall market condition, and thus indirectly on the situation of each of the other firms, oligopolistic firms are

3D CAD software is proprietary, protected by copyright, trade secrets and patents.[8] Despite attempts to promulgate standards in 3D 'CAD' software formidable interoperability issues remain and users are essentially 'locked in' once they have purchased a particular brand of software.[9] There is evidence that lack of interoperability causes a problem for users and results in expense, waste, reduced efficiency and lock-in which affects competition.[10] Imperfect interoperability is estimated to cost the US automotive supply chain at least $1 billion per year.[11] Incompatibility between two versions of Dassault Systemes' CATIA 3D CAD software delayed the delivery of the A380 in 2006 and resulted in a $6 billion loss for Airbus.[12] Wiring bundles in the A380's fuselage designed in Germany using one version of CATIA, V4, did not fit into the wiring spaces created by the French designers using the more modern CATIA V5. The 3D CAD industry was identified in the 2013 Commission Staff Working Document as experiencing interoperability problems.[13]

This paper starts with an overview of the current legal regulation of software interfaces, namely application programming interfaces 'APIs' and data formats. The next section considers what lessons we can learn from doctrinal and empirical research of the 3D CAD industry.[14] The third part of the paper considers proposals for reform made by various commentators and identifies which proposal could improve openness without damaging

---

interdependent. European Commission Guidelines (2004/C 31/03) on the assessment of horizontal mergers under the Council Regulation on the control of concentrations between undertakings, OJ [2004] C 31/5, footnote 29.

[8] The stated policy in the suppliers' annual reports is to protect their intellectual property rights by a combination of patent, copyright, trademark and trade secret protections, confidentiality procedures and contractual provisions. Apart from Siemens, software and related services is their sole product. Between them they own over 1000 patents.

[9] Cyon Research 'Survey of Engineering Software Users' Cyon Research Corporation 2009, 19, found that users ranked interoperability as one of the three most important selection criteria, along with total cost of ownership and improving product quality. http://cyonresearch.com/Portals/0/files/whitepapers/Cyon%20Research%202009%20User%20Survey.pdf (accessed 31 October 2016); Chad Jackson and David Prawel 'The 2013 State of 3D Collaboration and Interoperability Report' Lifecycle Insights and Longview Advisors, 7 – 8. http://www.plm.automation.siemens.com/en_us/Images/Lifecycle-Insights-2013-Collaboration-Interoperability_tcm1023-210162.pdf (accessed 31 October 2016).

[10] Jackson (n 9).

[11] Smita Brunnermeier and Sheila Martin, Research Triangle Institute 'Interoperability Cost Analysis of the U.S. Automotive Supply Chain' Center for Economics Research for NIST (Final Report, March 1999).

[12] Doug Bartholomew Born in a Storm Baseline www.baselinemag.com/c/a/Projects-Processes/PLM-Boeings-Dream-Airbus-Nightmare/3 (accessed 11 January 2017)

[13] Commission Staff Working Document 'Analysis of measures that could lead significant market players in the ICT sector to license interoperability information' SWD (2013) 209 final, 18 ('Commission Staff Working Document').

[14] Ten interviews in a semi-structured form where undertaken by the author with senior executives of 3D CAD suppliers, industry analysts, members of the ISO STEP committee, 3D CAD users and suppliers of complementary software.

innovation. The proposals mainly require changes to Intellectual Property Rights 'IPR' protection. Fourth and finally a new recommendation is made that requires minimum intervention, balancing the control and access requirements of industry and users. The recommendation centres on improving access to information by the removal of Article 6.2(b) of the Software Directive[15] to allow for the sharing and public registration of interface specifications including those obtained by reverse engineering.

2. Legal Regulation of Interoperability

This section will give an overview of the present regulatory landscape of software interfaces by IPRs and competition law and the role of standards to encourage interoperability.

2.1 *IPRs in software interfaces*

By placing emphasis on the functionality of data formats the CJEU considered that the ideas and principles which underlie interfaces are not expression and are not copyright protected.[16] This means that an interface specification written by analysis of a program without copying the expressive code can avoid infringing copyright.[17] APIs were considered in the US case of *Oracle v Google* where the District Appeal Court was not persuaded by the functional, subject matter approach, to determine the ideas/expression dichotomy. In addition to direct copying of the code, indirect copying of the sequence, structure and organisation had taken place which should be determined using the abstraction, filtration, comparison test. The US can take a traditional approach to ideas/expression and then implement the fair use exception.[18] In Europe the CJEU gave a purposive interpretation of the Software Directives so that the functionality of interfaces should not restrict interoperability.[19]

To improve interoperability, so that follow on innovation is generated, it is necessary to have access to and use of interface information but the machine code which is distributed to users is not readable. Reverse engineering to create interface specifications occurs even though access to the software is restricted by Article 6 of the Software Directive.[20] There is little evidence that reverse engineering presently provides a significant answer or incentive to suppliers in the 3D CAD industry to disclose interface information.

Reverse engineering can use the 'clean room' procedure where analysis of competitors' programs to write an interface specification is kept separate from the writing of the code to

[15] Article 6.2 (b) prohibits the dissemination of information obtained by legitimate reverse engineering as information obtained by decompilation shall not … '*be given to others, except when necessary for the interoperability of the independently created computer program*'(emphasis added).

[16] Case C-406/10 *SAS Institute Inc v World Programming Ltd* [2012] (Judgement of the Court (Grand Chamber) (2012) CMLR 4, para 42.

[17] Case C-406/10 *SAS Institute Inc. v World Programming Ltd* [2011] ECR I-1, Opinion of AG Bolt, para 55 - 57.

[18] *Oracle America, Inc. v Google Inc* (9 May 2014, 2013-1021, -1022) (Court of Appeal Federal Circuit) 50-60.

[19] *SAS Institute* CJEU, para 39 - 46.

[20] The Software Directive imposes certain restrictions on reverse engineering which includes black box reverse engineering in Article 5 and decompilation in Article 6.

implement the specification which prevents the copying of the expression.[21] Interface specifications are created by the decompiler. The machine code is decompiled to a higher level language which can be read by the decompiler. Using the higher level information, a specification is written setting out the characteristics of the interface. The subject of the interface specification can be an API or data file. 3D CAD suppliers were aware of this practice and thought it would be used in their company for reverse analysis.[22]

The interface specification is the outcome of reverse engineering. It is the decompiler's interpretation of the requirements of the interface and does not include either the machine or original source code.[23] It comprises the ideas and principles of the software's interface and hence does not contain copyright belonging to the software's rightsholder. Despite the fact that the interface specification is normally free of copyright material, other than copyright belonging to the decompiler, the decompiler is restricted by Article 6.2(b) of the Software Directive in their ability to share the interface specification.[24] The rightsholder cannot use trade secret law to prevent decompilation[25] but the Software Directive creates a statutory trade secret law.[26]

The proposal for the European Trade Secret Directive confirmed the legitimacy of reverse engineering while leaving unaltered the position for reverse engineering software protected by copyright law.[27] There is normally no limitation on the use of the trade secret once lawfully attained and it is not feasible to differentiate between acquisition and use.[28] However Article 6 of the Software Directive does limit both the acquisition and use of information to the purpose of interoperability and for the independently created software while also preventing sharing of information. In addition the use of the software itself is not

[21] prepared by Amedée Turner QC, one of the Parliamentary rapporteurs ,Eur. Parl. Doc. 136.025 (fin.) Annex II, DOC EN/RR/91422 (Nov. 1989) (the 'Turner Report');

[22] Interviews (n 14).

[23] Turner Report (n 21).

[24] See (n 15) for text of Article 6.2(b).

[25] Inge Graef, 'How can Software Interoperability be achieved under European Competition Law and Related Regimes?' (2014) 5(1) *Journal of European Competition Law & Practice* 17; William Cornish et al *Intellectual Property: Patents, Copyright, Trade Marks and Allied Rights* (Sweet & Maxwell 2013) 326 - no action lies for secrets embodied in physical objects which are available on the open marker which can be analysed to find out its secret content.

[26] Software Directive Article 6 2 (b) prohibits the dissemination of information obtained by legitimate reverse engineering, (n 15).

[27] Commission 'Proposal for a Directive on the protection of undisclosed know-how and business information (trade secrets) against their unlawful acquisition, use and disclosure' COM (2013) 813 final, para 5.1 and Article 4, 1(b); Tanya Aplin 'A critical evaluation of the proposed EU Trade Secrets Directive' (2014) 4 *Intellectual Property Quarterly 257, 262*

[28] Ronald Knaak and others, 'Comments of the Max Planck Institute for Innovation and Competition on the Proposal for a Directive on the protection of undisclosed know-how and business information (trade secrets) against their unlawful acquisition, use and disclosure'(2014) 45 (8) *International Review of Intellectual Property and Competition Law* 953, 961

permitted.[29] The final Trade Secrets Directive itself is more restrictive.[30] It permits reverse engineering[31] where the acquirer *is free from any legally valid duty to limit the acquisition of the trade secret.*[32] The restrictions in Article 6 of the Software Directive could amount to such a legally valid duty and this would take decompilation of software outside the scope of the lawful acquisition, use and disclosure provisions of the Trade Secrets Directive. The consultation documents on the Trade Secrets Directive published by the Commission contain no information to support the amendment or its potential consequences.[33] There is reference to limiting reverse engineering when it has been contractually excluded, involves dishonest commercial practices or is contrary to national unfair competition practices.[34] The impact of the change to the wording of Trade Secrets Directive will need to be established during implementation and subsequent litigation but meanwhile permission to decompile software appears to remain with the Software Directive with all its limitations.

Even where there is no copyright in the interface specification the software's rightsholder may have protected the interface through patent protection. 3D CAD suppliers have been granted numerous patents.[35] With the exception of the Unified Patent, there is no exemption from patent protection for the purpose of interoperability which leaves decompilers exposed to patent infringement even if they comply with the Software Directive.[36] The concept and

---

[29] Ibid, 961

[30] Council Directive 2016/943 of 7 June 2016 on the protection of undisclosed know-how and business information (trade secrets) against their unlawful acquisition, use and disclosure[2016] L 157/1, recital 16 states the Directive should not create any exclusive right to know-how or innovation protected as trade secrets. The Directive's definition of trade secrets is in line with art. 39 TRIPS - secrecy, commercial value and reasonable steps to preserve secrecy. As the spectrum of information currently protected by trade secrets is wide it was acknowledged that extending the existing IPRs or creating a sui generis IPR in trade secrets would result in over protection. Creating such a monopoly right would not allow for 'distinguishing between the misappropriation of information and the mere acquisition of knowledge (e.g. by reverse engineering or by parallel discovery). Commission Staff Working Document Impact Assessment on the proposal for a Directive of the European Parliament and of the Council on the protection of undisclosed know-how and business information (trade secrets) against their unlawful acquisition, use and disclosure {COM(2013) 813 final}, 45

[31] Reverse engineering is referred to in Article 3 (1) (b) of the Trade Secrets Directive as 'observation, study, disassembly or testing of a product or object'.

[32] Trade Secrets Directive Article 3 (1) (b) 'Observation, study, disassembly or testing of a product or object that has been made available to the public or that is lawfully in the possession of the acquirer of the information who is *free from any legally valid duty to limit the acquisition of the trade secret*.' (emphasis added)

[33] https://ec.europa.eu/growth/industry/intellectual-property/trade-secrets_en for example the Commission Impact Assessment (n 30) and the European Parliament report on the proposal for a directive of the European Parliament and of the Council on the protection of undisclosed know-how and business information (trade secrets) against their unlawful acquisition, use and disclosure (COM(2013)0813 – C7-0431/2013 – 2013/0402(COD)) (accessed 19 November 2016)

[34] Recital 16 and 17 of the Trade Secrets Directive.

[35] (n 8).

[36] Estelle Declaye & Matthias Leistner, *Intellectual Property Overlaps* (Hart Publishing 2011), 91, technical matter is usually excluded from copyright but the overlap of software is clearly an exception to the rule; Julie E Cohen & Mark A Lemley, 'Patent Scope and Innovation in the Software Industry' (2001) 89 *California Law Review* 1, 21 Reverse engineering is important in preserving competition and compatibility between products particularly in markets characterised by network effects.

implementation of software patents is flawed, does not incentivise innovation and could restrict the operation of standards and interoperability.[37] Patent protection fails to adequately address the indirect effect of control over interface specifications on interoperability and it is seriously doubted whether patent protection of interfaces can be economically justified.[38] There may be a failure of the market with overprotection of interfaces which cannot be rectified by reverse engineering or, as discussed below, by conventional competition law in an oligopolistic market.[39] Because interfaces are standards and have an indirect effect they give an unplanned expansion of IPR rules both for copyright and patents. The hidden nature of the code giving protection equivalent to a statutory trade secret also overprotects the interfaces.

2.2 *Intervention by competition law*

There is debate as to whether intervention, such as by competition law restricting IPRs to mandate disclosure of interface information, is justified.[40] IPRs are themselves a form of intervention but are seen by many as an almost inviolable right to encourage innovation. Supplier lock-in may justify intervention[41] and it may be necessary to strike a balance between control by the rightsholder to incentivise innovation and openness of interfaces to achieve interoperability. This can be achieved either ex ante by IPRs or ex post by competition law.[42] As interfaces have indirect effects their value comes from being a standard and the balance may favour openness more than for other subject matter in the

[37] Pamela Samuelson (n 2) - while patents may not be well suited to software inventions, there is said to presently be insufficient empirical evidence that patents are such a major impediment to interoperability that the exclusion of interfaces from patent protection is justified. There are however several examples of established firms with strong market positions taking patents on interfaces, possibly with the aim of controlling the development of competing and complementary products. Patents are considered most threatening to competition when they are held by established firms with market power which may use them to leverage their dominant position in one market into an adjacent market; Sally Weston and others 'Open Standards in Government IT: a Review of the Evidence' (UK Cabinet Office September 2012), 33.

[38] Maureen O' Rourke, 'Towards a Doctrine of Fair Use in Patent Law' (2000) 100 *Columbia Law Review* 1177, 1218 the exclusionary power of patents in interfaces is considered strong and noting that many interfaces are arbitrary, obvious and /or of low intrinsic value.

[39] Weston (n 37) 30.

[40] Alan Devlin, Michael Jacobs and Bruno Peixoto, 'Success, Dominance and Interoperability' (2009) 84 (4) *Indiana Law Journal* 1157.

[41] Supplier lock-in and market lock-in are distinguishable as in the former the cost of entering the market is prohibitive while in the later customers can afford to bear some of the cost and the market can rely on market forces and if necessary 'creative destruction' to change a dominant supplier in the market. Carl Shapiro and Hal Varian, *Information Rules - a strategic guide to the network economy* (Harvard Business School Press, Harvard, 1998); Paul Klemperer 'Competition when Consumers have Switching Costs: An Overview with Applications to Industrial Organizations, Macroeconomics, and International Trade' (1995) 62 (4) *Review of Economic Studies* 515.

[42] The merger regulations are however able to provide an ex ante control of interoperability which is also flexible as it takes the individual circumstances into account, for example *Intel/McAfee* Case (COMP M5984) Commission Decision [2011] OJ C 98 -1 and EUR-Lex 32011M5984.

computer program.[43] Industry interviews supported this differentiation in purpose and importance. [44]

To achieve more openness and interoperability disclosure of interface information is needed and competition law can give a remedy in exceptional circumstances.[45] This remedy is only available where there is an abuse of a dominant position, but the 3D CAD industry is oligopolistic with no single dominant supplier. The remedy of disclosure of information under competition law is not available in oligopolistic markets.[46] Interoperability is unlikely to narrow the definition of the market to a single supplier and it is also unlikely that the suppliers will be considered collectively dominant. The argument that has been made that competition law makes amendment to IPRs unnecessary[47] is disproved because in oligopolistic markets, where users are locked-in to suppliers due to a lack of interoperability, competition is affected but no remedy is available.

2.3 *Standards in software interfaces*

Standards are an important means of improving interoperability but can give additional power requiring the law to impose restraints on the royalties and other benefits enjoyed by the IPR holder.[48] Standards, particularly compatibility standards which cannot be avoided, give an unplanned expansion of the protection for both copyright and patents.[49] A further concern is that the royalty that can be imposed may be due to the nature of the standard rather than the value in the IPR. The IPR holder may try to profit from the standard's strategic position and extract excessive rents.[50] Patents in standards must be licensed on FRAND terms which have to be agreed between the parties without the Standard Setting Organisation's assistance. The 3D CAD industry has more than one formal standard including the most widely used

[43] Ashwin van Rooijen *The Software Interface between Copyright and Competition Law* (Kluwer Law International 2010), 43 – 46.

[44] Interviews (n 14)

[45] Case T-201/04 *Microsoft Corp v Commission* [2007] ECR II-3601.

[46]In the software industry dominance has been defined narrowly, Vincent Pickering and Maurits Dolmans, 'The 1997 Digital Undertaking' *(1998)19 (2) European Competition Law Review 108, 109.* Analysis of recent decision does not however give support to a lack of interoperability limiting market definition to single suppliers, Sally Weston 'The Legal Regulation of Interoperability in an Oligopolistic Market' (PhD thesis Bournemouth University 2015) 146 – 154.

[47] Josh Lerner and Mark Schankerman, *The Comingled Code: Open Source and Economic Development* (MIT Press 2010).

[48] The Commission's Digital Single Market strategy will concentrate on standards and interoperability as a critical area for growth of the digital economy. 3D CAD is relevant to several areas of EU policy priorities where standardisation activities play a key role, for example Advanced Manufacturing, Commission's 'Rolling Plan for ICT Standardisation 2016' (GROW_RollingPlan_2016_ICT_standardisation_BROCHURE_160225.indd 1, 2016)

[49] Ken Krechmer, 'Open Standards: A Call for Change' (2009) May *IEEE Communications Magazine* 88, 90-91.

[50] Knut Blind and others, *Study on the Interplay between Standards and Intellectual Property Rights* (Office of the European Union, Final Report 2011)

STEP standard but they all only provide limited compatibility.[51] Although there is no evidence that FRAND, rather than royalty free, has caused a problem for STEP or had an impact on the industry, this may change if, to improve interoperability, interface availability and informal standards become more prevalent. As the information available becomes more dynamic negotiating FRAND licenses could be a drag on innovation. The argument that claiming royalties in interface standards incentivises innovation is nuanced and has certainly not been made convincingly.[52]

3. Interoperability in the 3D CAD industry – what lessons can we learn?

This section will consider the impact of software interoperability in the 3D CAD industry within the present legal regime and what lessons can be learnt to inform future legal reform.

3.1 *Market pressure*

There appears to be little effective market pressure on the suppliers to improve interoperability. Suppliers with a significant part of the market benefit from a lack of interoperability and lock-in effects and may not want to license interoperability information for their product.[53] Their most important customers are OEMs[54] which value integrity of data as highly as interoperability.[55] The suppliers' concept of openness is to make their own software able to 'ingest' data from other suppliers' systems to encourage customers to stay with them.[56] Reverse engineering has not made a significant impact, certainly not sufficient to encourage suppliers to make disclosure of interface information. As the industry is an oligopoly with no dominant supplier a main legal tool for improving interoperability, competition law, is not available.

Despite the lack of pressure there has been a market response to improve interoperability. This has not always come from the suppliers of 3D CAD software but from other firms such

[51] Junhwan Kim and others, 'Standardised data exchange of CAD models with desing intent' (2008) 40 (7) *Computer Aided Design* 760. The essential elements of lost information include: construction history – the procedures used to construct the model; parameters, variables associated with dimensions and other values; constraints relationships between parameter values and geometry and features such as shape configurations. 51% of users of translation software get acceptable results 50% to 75% of the time. Longview Advisors, Inc. 'Collaboration & Interoperability Market Report' 2010, 51 file://bournemouth.ac.uk/data/staff/home/sweston/Downloads/cimr2010-longview.pdf (accessed 11 October 2016).

[52] Ibid; Weston (n 37).

[53] Commission Staff Working Document, 7.

[54] The Original Equipment Manufacturers 'OEMs' have the largest accounts and often dictate what software their tier suppliers use.

[55] Interviews (n 14)

[56] Ibid.

as translators[57] which supply specialised software to enable models in competing 3D CAD software to interoperate to some extent. The translator software companies have seen a market opportunity, and using standards, APIs and reverse engineering, they have developed software to assist the process of transferring data between proprietary 3D CAD systems.[58] While this gives some relief from lock-in it is a costly and complex process and is limited as standards and translation software provide only a partial solution.[59] Although there is presently little pressure on suppliers to disclose interfaces they do provide, APIs, and data formats to the suppliers of translation software and other complementary software.[60] But making the data available remains in the gift of the 3D CAD suppliers and even a version update can lead to temporary interoperability issues for complementary software. Improving the availability of interface specifications would improve reliability. However given the complex nature of the software it may still require the specialist knowledge of the translators to achieve effective compatibility and it is not certain whether it is technically possible for all four systems to achieve full functional compatibility. While market solutions in the form of translation software companies have emerged there generally appears to be little drive in the industry towards improving interoperability.

3.2 *Protecting Users' Data*

Despite the lack of interoperability 3D CAD software provides a good welfare benefit as it improves the potential to develop goods quickly and to manage related data efficiently. 3D CAD is sophisticated modular software developed over several decades and it is more than just a platform. OEMs and other users rely on 3D CAD software to create, edit, use and store what is probably their most valuable data. The software is highly complex and functional and has a 'critical core'[61] function in the users business, particularly as it stores their own proprietary data. Lack of interoperability prevents users switching to another supplier immediately and can result in lost data. Industry interviews showed that OEMs and other customers value integrity of data as highly if not more so than openness and full interoperability.[62] It is important that the problem of interoperability is solved without disrupting the market. The disruption to the music industry caused by digital downloads did

---

[57] For example Theorem Solutions www.theorem.com/Company/overview.htm and Tansmagic https://transmagic.com/. These companies use a variety of methods including APIs and reverse engineering to provide translators to enable the direct exchange of native CAD files (both accessed 31 October 2016).

[58] 66% of respondents used CAD translation software Longview Advisors 2010 (n 51).

[59] Translation software supplied with 3D CAD packages allows for one direction translation only. Less than 33% of engineering companies surveyed used a third party translator and of those only 45% indicate that they get the results they want with these applications better than 75% of the time. 'Collaboration & Interoperability Market Report' (Longview Advisors Inc. 2008) www.proficiency.com/downloads/3DMarketReport2008.pdf (accessed 12 November 2010).

[60] Trevor Leeson, Theorem Solutions (MCADCafe interview 2014 PTC) http://www10.mcadcafe.com/video/Theorem-Solutions-Trevor-Leeson-Principal-Consultant-Tech.-Director-Major-Accounts/43971/media.html (accessed 17 October 2014).

[61] Interviews (n 14)

[62] Ibid.

not directly harm the user as music became more available.[63] Disruption to the 3D CAD industry and to the supply of proprietary software in which users' data is stored could be very harmful to the user and to society generally as manufacturing industry would suffer as vital know-how and legacy data could be lost.

Continuing follow on innovation would appear to have more welfare benefit than some form of creative destruction which could destroy the existing software and replace it with a new platform. 3D CAD suppliers have invested heavily in designing sophisticated software and a model that just replaces it with something else is undesirable, unless it gives full backward interoperability, which given the complexity of the software, is very unlikely. Interoperability helps follow on innovation. Not only does the software have intrinsic value but the users' proprietary data is extremely valuable. It is important to the user that they can access their data now and in the future.[64] Interoperability allows this to happen but any changes to the legal regime should not destabilise the industry as it is necessary to ensure that users can continue to use the format of the software their own proprietary data is stored in. Changes must also take into account users' needs, not only as consumers seeking competitive prices, but also to protect the integrity and access to their own proprietary data. The functional nature of the software is relevant not only to determining the IPRs status of the interface but also to ensuring the legal regime balances the need for interoperability with the need for data integrity.

3.3 *3D CAD software interfaces as de facto standards*

In addition to taking into account the impact, not only on the suppliers' incentive to innovate, but also any harm that could be done to the integrity and continuity of access to the users' data, software interfaces require different considerations and treatment to other subject matter in a computer program. The concept that software interfaces require different treatment has been recognised in previous research[65] in copyright case law[66] and during industry interviews. This is because software interfaces not only directly affect interoperability, but also because of their indirect effects as standards. Interfaces have an indirect function of controlling interoperability and access, not only in competing software and networks but also in complementary software and access to the user's existing own data. Their impact and value is amplified solely because of their role as standards and this extends and distorts the IPR protection they enjoy to the detriment of competitors, suppliers of complementary software, and users. Interfaces will have a different optimal balance than the core subject matter with more openness to counteract the amplifying effect that control over standards has on market power.[67]

---

[63] It is uncertain what indirect effects, such as reduction in choice of artists, the disruption to music industry had on the consumer but the availability of music generally increased with introduction of digital media and iTunes, file sharing and streaming.

[64] Interviews (n14) Users of 3D CAD insisted the Siemens' JT data format was adopted as an ISO standard to make it suitable for legacy storage as the data format will be maintained which is not assured if it remained proprietary.

[65] Ibid.

[66] *SAS Institute* CJEU.

[67] Rooijen (n 43) 43 – 46,197.

3.4 *Summary*

The different attitude expressed during industry interviews to the sharing of interfaces from the core software as well as the recognition that the value of interfaces arose less from their innate innovation than their value as de facto standards, supports existing theory. While there is some market response within the industries to the demand for interoperability, and adoption of formal standards, these give a very incomplete solution. Significant supplier lock-in remains and this can justify intervention. A dominate feature of the industry is the value of the users proprietary data stored in the software. The integrity of data is crucial for the user and intervention must take this into account and ensure backward interoperability is not reduced.

4. A review of current proposals to reform the regulation of interoperability

Regulating to promote interoperability must strike a balance between IPR protection which gives control, and the need for access and use of interface information which gives openness. This will also achieve the economic goals of increasing incentives to innovate and promote efficient allocation of resources.[68] The central economic problem is said to be that copyright protection of software conflicts with the desire for information, particularly interface information, to be disseminated. Overprotection favours present innovation over future innovation.[69] Future innovation can be improved by allowing a degree of copying of interfaces.[70] The reverse engineering provisions in Article 6 of the Software Directive is an attempt to strike a balance. They were however formed more by lobbying than by economic Pareto optimality or empirical evidence.[71] The introduction of the Software Directive was influenced by lobbying from US trade representatives and negotiators and *Computer Associates v Altai* and *Sega Enterprise Ltd v Accolade Inc.*, which held that decompilation to achieve interoperability was 'fair use' were not decided until after the Software Directive had been enacted.[72]

---

[68] Willaim Landes and Richard Posner, 'An Economic Analysis of Copyright Law' (1989) 18 *Journal of Legal Studies* 325; Josef Drexl (ed) *Research Handbook on Intellectual Property and Competition* (Edward Elgar 2008) Ruth Towse and Rudi Holzhauer in the introduction to *Economics of Intellectual Property* (Edward Elgar 2002).

[69] Roger Van den Bergh, 'The role and social justification of copyright: a 'law and economics' approach' (1998) 1 *Intellectual Property Quarterly* 17-34, 28.

[70] Kenneth Dam advocates copying of operating system interfaces to write new application programmes and where follow-on innovation adds substantial surplus value. 'Some Economic Considerations in the Intellectual Property Protection of Software' (1995) 24 *Journal of Legal Studies* 321.

[71] Ian Brown and Christopher T. Marsden in *Regulating Code* (2013 the MIT Press), 87 give another example of policy making that is based more on lobbying by stakeholder than the product of a rational decision-making process.

[72] *Computer Associates v Altai* (1991) 982 F.2d 693; *Sega Enterprises Ltd v Accolade Inc.* (1992) 977 F.2d 150; The case of *Whelan v Jaslow* (1986) 797 F.2d 1222 had been decided and opponents to decompilation asserted that even interface information could be part of the program's sequence, structure and organisation and protected as copyright law in the US. Noam Shemtov, 'The Legal Regulation of Decompilation of Computer Programs: Excessive, Unjustified and in Need of Reform' (PhD thesis QML 2013) 138 - 139.

The Commission appears to consider that as the Directive and in particular the decompilation provisions 'were the result of intensive debate among all interested circles…the balance found then appears to be still valid today' although maintaining the status quo is influenced by not wanting 'to reopen the floodgate of debate'.[73] In 2013 however a Commission Staff Working Document looked at measures that would lead to the licensing of interoperability information. They considered that 'copyright does not offer control over the information per se embodied in a work; only the expression is protected of that information where it constitutes the author's intellectual creation.' Because software is distributed in machine code even where interoperability information is not copyrightable there is no guarantee of effective access to the necessary protocols for developers.[74] The Software Directive also gives control over the information obtained by reverse engineering even when it does not include expression as there are restrictions on sharing the information.[75] While the ideas behind software interfaces are not copyright protected more freedom to access and share the information is needed to improve interoperability.

The Commission Staff Working Document is one of several proposals that try to achieve a balancing interest. While these proposals will be considered it has not yet been determined how best to identify and obtain the correct balance. Neither the correct optimum balance nor the criteria for identifying it have yet been identified.[76] While the need for balance is reinforced by the proposals there has been less progress on establishing criteria for identifying the 'pivot' between control and openness.[77] This means it is difficult to put these principles into practice.

To avoid instability and protect data integrity it is proposed that the correct approach is to start with changes that cause least intervention and only increase intervention if after eva luation and reflection there is insufficient improvement in interoperability. To be effective any changes would need to be made across the EU and the first evaluation will be the recent proposals for an interoperability directive.[78]

The proposals will be considered in reverse order, starting with those that are most interventionist and moving towards those that need the least change to the existing law to give more freedom to the market to use information that does not have IPR protection.

[73] Commission Report 2000/199 on the implementation and effects of Directive 91/250/EEC on the legal protection of computer programs [2000] COM 199 final, 21.

[74] Graef (n 25) 15.

[75] Software Directive Article 6.2 (b) (n 15).

[76] Shemtov (n 72), 44 proposed a model using recoupment of R&D costs as a benchmark for licensing innovative software architecture to competitors when coupled with revenue generated on sales as sufficient incentive to continue and invest in original research and development; four criteria to assess the social welfare effects of the law's recognition of a right to reverse engineer in Pamela Samuelson and Suzanne Scotchmer 'The Law and Economics of Reverse Engineering' (2002) 111 *The Yale Law Journal* 1575.

[77] Ibid; Rooijen (n 43) talks of 'middle ground between openness and control', 234.

[78] The outcome of the negotiations for the United Kingdom's departure from the European Union will determine whether the United Kingdom will be bound to implement any Directive.

4.1 *An interoperability directive*

An 'Interoperability Directive' has been seen as a way to ensure a common approach across Europe to enforce disclosure of interface information and other remedies.[79] A Directive is a mechanism or wrapper that could include various rules to implement changes aimed to prevent IPRs being used to exclude competition without resorting to ex post competition law or to find a deliberate strategy of exclusion to justify intervention.[80] Dominant suppliers, particularly of network or application infrastructure in software based internet services, such as software as a service, would be prevented from 'locking in' certain segments of the market which is said to happen when providing services across different platforms is costly.[81] This would strengthen competition between software based internet providers and platform owners and the increased interoperability should reduce training costs and encourage new entrants to markets. [82]

An Interoperability Directive, including trade secrets and patent licences, was discussed in the Commission Staff Working Document in 2013 with Article 114 TFEU as the legal basis. Proposals included a mandatory license of right on FRAND terms or an interoperability exception to mirror Article 6 of the Software Directive.

The Agreement on a Unified Patent Court[83] provides that the rights conferred by European patents with unitary effect will not extend to the use of information obtained under Articles 5 and 6 of the Software Directive.[84] On establishment of the unitary patent and Unified European Patent Court patent holders will not be able to invoke patents against products implementing interface information obtained by black box or decompilation reverse analysis. This is similar to the provision in the doomed Software Patent Directive. In 2002 the European Commission proposed a Directive on software patentability. The European Parliament proposed an amendment that the use for the purposes of achieving interoperability would not be considered a patent infringement. The Directive was defeated by a vote in the European Parliament in 2005 and has been dropped.[85] As the exemption in the unitary patent does not however apply to classical European patents, or national patents which contain no such exemption, they will continue to stand in the way of using interface information.[86]

The Commission Staff Working Document considered a proposal extending the exemption in unitary patents to all European or national patents which would prevent them being invoked against the use of information obtained from reverse engineering for interoperability purposes. No royalty fee would be payable under this interoperability exception. The

[79] Ibid.

[80] Pierre Audoin Consultants 'Economic Social Impact of Software and Software-Based Services' Final Report August 2010 Smart 2009/0041, 207 - 208 http://cordis.europa.eu/fp7/ict/ssai/docs/study-sw-report-final.pdf (accessed 16 December 2014).

[81] Ibid, 207.

[82] Ibid, 208.

[83] www.epo.org/law-practice/unitary/patent-court.html (accessed 20 January 2017)

[84] Article 27(k) of the Agreement on Unified Patent Court (11 January 2013) document 16351/12.

[85] Commission 'Proposal for a Directive of the European Parliament and of the Council on the patentability of computer-implemented inventions' COM (2002) 92 final, OJ C 151E/129.

[86] Graef (n 25) 16.

exception removes the risk from developers who rely on Articles 5 and 6 of the Software Directive to reverse engineer interoperability information that they would risk infringing patent rights, perhaps unknowingly.[87] An alternative option was for patent claims which relate to interoperability information to be subject to an automatic license of right on FRAND terms similar to the arrangements for essential patents in standards.[88]

A further approach is to licence both patents and trade secrets information on a case by case basis giving individual consideration rather than a general application of an exception or the right to licence regime. This would be modelled on the Framework and Access Directives for access to and interconnection of electronic communications networks.[89] Obligations to licence would be imposed on undertakings with significant market power which would require new bodies in each national regulatory authority to carry out ex ante analysis of the market to identify those suppliers who had significant market power.[90] If the definition of significant market power is equivalent to dominance under European competition law[91] it would not apply to oligopolies such as the suppliers in the 3D CAD market and so would not provide a remedy to lack of interoperability and lock-in in oligopolistic markets. The consultation carried out as part of the Commission Staff Working Document identified that interoperability problems existed with suppliers that would not qualify as significant market players and gave the example of interoperability issues between different CAD systems.[92]

After raising these initiatives the Commission Staff Working Document concluded that an Interoperability Directive should not proceed. Even if effective, establishing the new bodies would be costly, and the analogy with the electronic communications networks breaks down. Software industries, such as 3D CAD, do not have identifiable market bottleneck assets.[93] They also have a different territorial scope as the electronic communications networks are primarily national whereas software markets generally cover the whole of the EU which makes implementation by national regulatory authorities inappropriate and ineffective. It could also introduce a public law approach adding a third dimension to intellectual property law and competition law.[94]

Introducing automatic licences of right, other than just for the unified patent, would entail a revision of legislation in all member states, which presumably the Commission Staff

[87] Commission Staff Working Document (n 13), 11.

[88] Ibid.

[89] Council Directives 2002/21/EC on a common regulatory framework for electronic communications networks and services; 2002/19/EC on access to, and interconnection of, electronic communications networks and associated facilities (as amended by Directive 2009/140/EC OJ L337/37).

[90] Commission Staff Working Document, 12, Safeguards would need to be in place including a framework explaining the principles to follow to ensure alignment between regulatory actions and the breach of interoperability principles as well as avoiding the disclose of information that reveals the technology and functionality implemented by a device or a system beyond its interfaces.

[91] Framework Directive 2002/21/EC (n 89), recital 25.

[92] Commission Staff Working Document, 17 – 18.

[93] Ibid, 14-15.

[94] Rooijen (n 43) 234.

Working Document did not consider feasible. It was also doubted whether the provisions could use Article 114 TFEU as a valid legal basis or meet the principles of proportionality.[95]

It was assumed that implementation would be costly but the new bodies would only be necessary if action was limited to undertakings with significant market power. Automatic rights and interoperability exception for patents would, after amendment to the law, leave implementation and enforcement to the parties.

The Commission Staff Working Document preferred the introduction of non-legislative measures to lower transaction costs and foster a culture of licensing through the use of model licences and guidelines on valuing interoperability to help parties agree royalty rates.[96]

4.2 *Compulsory disclosure and mandatory licenses of right*

Proposals to amend the existing legal position often consider some form of mandatory disclosure of interface information.[97] Article 102 provided such a mandatory licence remedy in *Microsoft.* The effectiveness of the remedy is limited as it is ex post and takes years to provide relief by which time the market may have moved on. Compulsory disclosure through some form of regulation is rare in copyright.[98] Regulation has serious drawbacks including the cost burden to the regulator and regulated and can stem the innovation it intended to create.

Patent holders can already volunteer licences of right, normally in return for a reduction in licence fees. This allows all patent holders the opportunity of licencing the technology on reasonable royalty terms. As discussed earlier the Commission Staff Working Document considered converting this to mandatory licenses of right for patent claims covering interoperability information.[99] Patent holders would be required to offer licences on FRAND terms, similar to the commitments in standard setting organisations. However as this would require changing the legislation of all Member States the Commission considers implementing automatic mandatory licenses of right would be 'very difficult'.[100]

The Software Directive imposes a trade secret regime for computer programs[101] which conflicts with a requirement for disclosure of any code but particularly the source code. Mandatory disclosure arrangements, including registration of the source code, could give copyright protection only in return for disclosure of the subject matter.[102] But the source code is the 'crown jewels' and their disclosure is unlikely to strike the right balance between

[95] Commission Staff Working Document, 14.

[96] Ibid, 15-16.

[97] For example requiring dominant social network systems such as Facebook to provide APIs and other interoperability obligations, Brown (n 71) 190.

[98] Begona Gonzalez Otero, 'Compelling to Disclose Software Interoperable Information' (2013) 16 *The Journal of World Intellectual Property* 2, 8.

[99] Commission Staff Working Document, 11.

[100] Ibid, 14 and 16-17.

[101] Rooijen (n 43) 236.

[102] Ibid, 235.

revealing interface information and the 'thin' protection of copyright granted to the valuable and functional know-how in the source code.[103]

A more direct and proportionate approach is a registration requirement by the rightsholder of the interface specification. A disclosure requirement limited to the interface would not relate to the source code itself but only to the interface specification. Disclosure of interfaces safeguards the public domain by limiting copyright protection to the computer program rather than the interfaces and 'domain of interoperable programs.'[104] This would protect the rightholder's control with less detriment to incentives to innovate. It would also limit transaction costs as it removes the need for third parties to invest substantially and without certainty of success in reverse engineering. The interoperability information would be offered for free without competitors having to get a license from the copyright owner or to determine the royalty rate. The cost of preparing the information rests on the rightholder which is probably the most cost effective solution but these costs may be substantial[105] and may be a disproportionate burden on small developers.[106] It is also possible that what the rightsholder makes available is not the optimum information. This could involve the rightsholder in work that is of little benefit to the consumer. It is effectively imposing a command economy on interface information rather than a demand economy which is achieved by reverse engineering. If however the rightsholder disclosed information in response to the threat of reverse engineering it is more likely the information that is disclosed would more closely map onto the information that is required by the market which should be more efficient.

Even where copyright protection is conditional upon full disclosure of interfaces, monitoring and enforcement remains problematic. It is difficult to define the point at which adequate disclosure has taken place to ensure the body of the computer program has copyright protection.[107] For example, what level of disclosure or incomplete disclosure is required and does the copyright owner have the say on where the interfaces are and how many? As discussed earlier, interfaces can be defined as the rules by which data or instructions can be repetitively transferred between elements of a computer system.[108] Interfaces exist where such transfers occur, making it difficult to categorise which portion of a program is truly an interface.[109] As almost any part of the program can be considered as an interface[110] it would

[103] Ibid, 235.

[104] Ibid, 237.

[105] For example Microsoft costs to comply with the Commissions interface disclosure requirement, Ian S. Forrester, 'Victa lacet mihi causa: the compulsory licensing part of the Microsoft case' in Luca Rubini (ed), *Microsoft on Trial* (Edward Elgar, Cheltenham 2010), 97-98.

[106] The 'cheapest cost informer' is the original developer and owner of the software, Dieter Schmidtchen and Christian Koboldt, 'A Pacemaker That Stops Halfway: The Decompilation Rule in the EEC Directive on the Legal Protection of Computer Programs' (1993) 13 *International Review of Law and Economics* 413.

[107] Graef (n 25).

[108] Michel Colombe and Caroline Meyer, 'Interoperability still threatened by EC Software Directive: a status report' (1990) 12 *European Intellectual Property Review* 325, 328.

[109] Ibid.

[110] Robert Hart, 'Interfaces, Interoperability and maintenance' (1991) 13 (4) *European Intellectual Property Review* 111.

be difficult to establish conclusively what constitutes an interface and to say whether full disclosure has taken place, particularly as the source code would not be available. This introduces a high level of legal uncertainty. This level of uncertainty combined with the degree of intervention required to enforce the rightsholder to compile and disclose information means this option is not recommended.

4.3 *Reducing the term of IPR protection*

The fifty years minimum term protection required by the Berne Convention for literary works far exceeds the useful life of the software. Shortening the term is not however easy. The difficulty of fixing a single appropriate term meant that approach was rejected for replacement parts under design protection law. [111] 'Attempting to influence the dynamics of platform software competition by identifying a single, fixed term of protection for interface specifications similarly appears too detailed an instrument to be effective.'[112] Reducing the term of protection for all software does not improve the position for interoperability but could adversely damage incentives to innovate. As interfaces have indirect effects as de facto standards reducing protection for interfaces alone to a shorter term is theoretically appropriate. This could be based on the time it would take to reverse engineer the relevant specifications. Given that interfaces can be subjectively identified and defined it would be difficult to operate a two tier regime for interfaces and other software. A two tier approach could imply that these specifications are protected which is not the case.[113] Rather than having to determine the nature of interfaces under the ideas/expression dichotomy the courts would be deciding whether code was an interface to determine its duration of protection.

Introducing a shorter term of protection that is applicable to interfaces would be detrimental to vertical interoperability.[114] Users of complementary software could be put in a worse position than they are now. While voluntary data exchange does take place within the 3D CAD industry not all suppliers of complementary software have advanced access to APIs and their software might stop working each time a new version of the operating or platform software is released.[115] Many users would suffer disruption to interoperability between existing complementary software and new versions of operating or platform software. Users of complementary software for 3D CAD systems such as Finite Element Analysis and translation software could experience problems when new versions of the 3D CAD software are released and their complementary software may stop working for weeks or months.[116]

---

[111] A single shorter term of protection was rejected for design protection laws, Rooijen (n 43) 169 & 221 and could conflict with the Berne Convention but Rooijen, 221 text on footnote 1036.

[112] Ibid, 221.

[113] *SAS Institute*.

[114] Vertical interoperability is complementary systems which create new functionality as opposed to horizontal interoperability for systems with similar or overlapping functionality, Commission Staff Working Document, 6.

[115] For example interoperability data is exchanged between Dassault Systemes' CATIA design software and suppliers of FEA (Finite Elements Analysis) software. Even here problems occur. When CATIA is upgraded the FEA supplier may not receive the information in advance and there may be a period when the programs are not compatible. Making the interface information available is in the gift of the 3D CAD supplier.

[116] Interviews (n 14) and ibid. 3D CAD suppliers may have a competing version of the complementary software application, for example FEA analysis, so there is no incentive to release APIs to ensure compatibility.

The suppliers of complementary software generally find APIs are not available before the release, or frequently not at all and have to rely on reverse engineering.[117] Even if the copyright term was reduced to a matter of months, control would still remain with the operating or platform supplier. If interfaces were given a short term of protection rather than excluded under the present regime the position for suppliers and users of complementary software could be worse.

4.4 *Specific exclusion for interface information*

The Software Directive and case law in the CJEU has categorised interface information amounting to ideas and principles as not being copyright subject matter.[118] Case law has given guidance on what effectively amounts to an exclusion for those elements of software which are not concrete elements of expression.[119] While the ideas and principles behind the interface, which can be extracted by reverse engineering, are not protected the code itself can still not be copied. A proposal for a specific exclusion would go further than the existing legal position by excluding the code in the interface from copyright protection. While interfaces can be subjectively identified and defined, the proposal is considered feasible as although many parts of a computer program could be named an 'interface', an interface specification is distinguishable from its implementation.[120] It is only the lines of code that constitute the specification that would be excluded from protection which is only a small part of the interface.[121] In practice this would be limited to the machine code as the source code would not usually be available to the decompiler, only the decompilers interpretation of the source code achieved by decompilation.

An exclusion removing code in interfaces from copyright protection would not entirely remove control as access to the software's code would remain limited to the purposes of interoperability. However when software is reverse engineered under Article 6 the decompiler would have a right to use the code for the permitted purposes. There is a need and benefit in copying the exact expression as it is an efficient way to achieve interoperability and can achieve the benefits of standardisation. Such an ex ante approach could give certainty. The courts have not gone as far, relying on the expression/idea dichotomy to provide a remedy that falls short of this proposed exclusion. If however the information includes code, as that is part of the expression, it is probably copyright protected even though its originality may be subordinate to its purpose.[122] Dissemination of the code, in addition to the interface specification, could economically be a very sensible re-use of the code as it would reduce waste and cost. If it was limited to code implementing interfaces, rationally, it should not reduce incentives to innovate.[123] This implementation does though require

[117] Interviews (n 14).

[118] *SAS Institute.*

[119] *SAS Institute; Oracle v Google.*

[120] Rooijen (n 43) 218.

[121] Ibid, 219.

[122] *Oracle v Google* refuted arguments that interfaces code was not copyright protected just because it had to be compatible, 42.

[123] The originality of code in interfaces is normally limited by its function and the value in interfaces comes from their indirect effects rather than their intrinsic innovation, for further discussion see section 3.3.

amendment to the Software Directive in a way that redefines the copyright protection given to software rather than just changing the arrangements for access to those parts of the code that are not copyright protected. For this reason it is recommended that these amendments should not proceed unless the following recommendation for sharing interface specifications, excluding code, does not achieve a balance that improves interoperability.

5. Recommendation

The above proposals are concerned with imposing obligations on rightsholders and changes to IPRs themselves. The following recommendation only concerns access to interface information which is not copyright protected. The sharing of interface specifications obtained by reverse engineering will be legitimised and a public register established to increase awareness and use. This will improve access to the information needed to achieve interoperability without imposing any positive obligation on the rightsholder.

Presently Article 6 of the Software Directive employs copyright to protect trade secrets. It gives software companies the enviable position that they can license their products to the world while still protecting the trade secrets contained in those products without the need to get patent protection for functional elements. If there was no need for the software to be compatible that position could be acceptable, however the strong protection of copyright and trade secret skews the balance to be overly closed.

No other general principle of intellectual property law exempts the ideas underlying products from study by those wishing to create competing products.[124] The law of trade secrets and confidentiality permits the study of ideas by reverse analysis.[125] Trade secrets once learned by another are theirs to use. The ideas contained in cookery books can be studied to develop competing recipes provided the expression is not copied.[126] What is considered here is redressing the anomaly that ideas and other non-copyright protected aspects of software are not visible and that there are also restrictions on disseminating any information obtained from decompilation.

5.1 *Reverse engineering software interfaces*

Decompilation and reconstruction in a higher level language can result in the exposure of some vital knowhow that would otherwise remain protected as a trade secret. It will reveal code which is copyrightable expression. As the location of the code that is essential for interoperability may not be identified without an analysis of the wider program even the conscientious engineer may discover more than is essential for interoperability. It is for these reasons that Article 6 was so contentious and drafted restrictively.[127] However reverse

[124] Maurizio Borghi and Stavroula Karapapa, *Copyright and Mass Digitization* (OUP 2013); Colombe (n 108) 327.

[125] Samuelson and Scotchmer (n 76).

[126] Colombe (n 108) 328.

[127] Article 6 is more restrictive than the common law approach as under fair dealing exceptions certain circumstances will justify the performance of otherwise restricted acts, regardless of whether other means exist to accomplish the same objective. Article 6 presume and requires the decompiler to have reviewed all other means to achieve the same end and found them to be inadequate. Bridget Czarnota and Robert Hart, *Legal Protection of Computer Programs in Europe - A Guide to the EC Directive* (Butterworths 1991) 76.

engineering even of the entire program will not of itself reveal all of the secrets of its design and development.

> Software reverse engineering does not lay bare a program's inner secrets. Indeed, it *cannot*. The inner secrets of a program, the real crown jewels, are embodied in the higher level of abstraction material such as the source code commentary and the specification. This material never survives the process of being converted to object code.[128]

As the Explanatory Memorandum to the Initial Proposal for the Software Directive states 'Although it is technically possible to decompile a program in order to find out information concerning access protocols and interfaces this is a lengthy, costly and inefficient procedure.'[129] It would not give any benefit to the pirates who have other shortcuts to produce illegal copies of programs. Decompilation is not a preferred technique as it is difficult and expensive but sometimes it is the only feasible means of obtaining the interface information.[130] The justification for preventing the disclosure of the interface specification obtained by reverse engineering has not been made out.

> 'A pirate wanting to copy a program can and will do just that – copy the available object code.' [Attempting to] 'recreate an entire program after it is compiled would be as sensible – and as economically efficient – as trying to unscramble an egg.'[131]

Also the product if used for commercial purposes would infringe copyright as a 'translation' of the original program.[132]

Reverse engineering is intended to act as a safety valve to enable a second program maker to develop an interoperable program when an existing program is not available.[133] It is also intended to encourage the copyright holder to disclose the interface information voluntarily,[134] although its success in this is mixed. In the public consultation of the Commission Staff Working Document, only 24% of respondents considered the possibility of reverse engineering interoperability information by third parties represented an incentive to license interoperability information while 21% considered it did not.[135] Industry executives

---

[128] Andrew Johnson-Laird, 'Software Reverse-Engineering in the Real World' (1994) 19 *University of Dayton Law Review* 843, 896.

[129] Explanatory Memorandum to the Initial Proposal for the Software Directive, 1989 OJ (C91) 4, 8 para. 3. 14.

[130] Alan Palmer and Thomas Vinje, 'The EC Directive on the Legal Protection of Computer Software: New Law Governing Software Development' (1992) 2 *Duke Journal of Comparative & International Law* 65, 82. Also Technology Protection Measures, also known as Digital Rights Management, can also be deployed as a protective shell around software and need to be circumvented before decompilation can take place. The anti-circumvention regime in the Information Society Directive does not apply to prevent reverse engineering of computer programs for the purpose of interoperability, Council Directive 2001/29/EC on the harmonisation of certain aspects of copyright and related rights in the information society [2001] L167/10, recital 50.

[131] Michel Colombe and Caroline Meyer, 'Seeking interoperability: an industry response' (1990) *European Intellectual Property Review* 12 (3) 79, 82.

[132] Ibid.

[133] Czarnota (n 127) 76.

[134] Rooijen (n 43).

[135] Public Consultation on the Access to Interoperability Information of Digital Products and Services in the Commission Staff Working Document. Q 3.6 Half those surveyed consider that the question was 'not

and experts were either unaware of the possibility of reverse engineering or did not consider it important to their decision making.[136] This indicates that reverse engineering is not considered a strategic issue in the 3D CAD industry. While reverse engineering provides a solution to interoperability in certain circumstances it is not a significant driver for the disclosure of interoperability information by rightsholders.

5.2 *Economic rationale for reverse engineering*

Permitting reverse engineering is thought to be economically sound as the innovator is protected by the costliness of reverse engineering and the lead time due to the technical challenge of reverse engineering[137] Costs and lead time allow the original innovator to recoup its investment and protect incentives to innovate. However the welfare benefits of allowing reverse engineering of interfaces differ from those of reverse engineering the software's core subject matter. In manufacturing industries reverse engineering is done to make directly competing stand-alone products.[138] Copyright law prevents direct copying of software and interfaces are reversed engineered to improve interoperability of both complementary as well as competing programs. Interfaces also have a different, indirect and magnified effect on interoperability and hence on competition and innovation.[139] Interfaces are standards and different considerations should apply to the treatment of IPRs, if any, present in the interface information. IPRs in the interface could be used to leverage market power in a way that was unintended as a matter of intellectual property law.[140] The interface's value comes predominantly from this interdependence rather than intrinsic innovation. Decompiling interfaces is not a market-destructive means of reverse engineering even in the absence of costs and technical challenges.[141] The 3D CAD suppliers certainly distinguished between protection of the interfaces and the inviolable kernels.[142] The logic that to be economically sound reverse engineering needs to be costly and difficult does not apply to interfaces. With respect to interfaces, cost and difficulty is only a waste and economically undesirable. There is no rationale for protecting the first comer. The Software Directive already specifies that decompilation can only be done for interoperability which restricts software reverse engineering more than in traditional manufactured items.[143] Reverse engineering is legitimised by its purpose,[144] and when restricted to interfaces, difficulty or cost, does not give any welfare benefit. The goal should be to make reverse

---

applicable' so the 24% may actually be higher among relevant constituents but still gives a very incomplete solution to the problem of interoperability.

[136] Interviews (n 14).

[137] Samuelson and Scotchmer (n 76).

[138] Ibid, 1613.

[139] Rooijen (n 43) 43 -46, 197 - 198; section 3.3

[140] Samuelson and Scotchmer (n 76) 1620.

[141] Ibid, 1653.

[142] Interviews (n 14).

[143] Samuelson and Scotchmer (n 76) 1655.

[144] Ibid, 1655.

engineering of interfaces as efficient as possible. This will aid both horizontal and vertical compatibility.

The Software Directive prevents access to ideas and other non-copyright protected aspects of software. Noam Shemtov argues that the purpose of reverse engineering should not be limited to interoperability. He makes the case that not only is this justified on the doctrinal grounds of the ideas and expression dichotomy but also by economic efficiency considerations. The relaxation of the restrictions on decompilation would not cause a loss of incentives to create and develop but result in a more balanced system properly addressing the unique properties of software products.[145] The US 'fair use' doctrine is not restricted solely to interoperability and is said to be more flexible than Article 6 of the Software Directive. The Software Directive does however make null and void any contractual provision attempting to prevent reverse engineering.[146] There is no such statutory provision in the US and contractual provisions in software licences prohibiting reverse engineering are common and the enforceability of restrictions on reverse engineering has been highly contentious.[147] While decompilation in the US is not specifically limited to interoperability there must be a legitimate reason,[148] the most prominent of which is for the purposes of interoperability.

5.3 *Art of the possible*

It is highly unlikely the climate in Europe has changed significantly to allow for a major change in the hard won provisions of Article 6. The Commission has looked at the Software Directive and the question of interoperability on at least two occasions since the introduction of the Software Directive and on both occasions has shied away from making any changes.[149] There appears to be little appetite for legislative amendments in any form[150] and certainly not on the psychological scale that would be required to permit decompilation of all software even with an ex post review by the courts as envisaged by Shemtov.[151] While there may be doctrinal validity in allowing access to non-copyright protected aspects of software, the effect on the required balance and the economic consequences have not been established to the extent that a convincing case could be made to a hostile audience.[152] Interviews in the 3D CAD industry reveal a clear distinction between allowing access to interfaces with a strong resistance to any dilution in control of what were considered core aspects of the computer

[145] Shemtov (n 72) 45.

[146] Software Directive Article 8(1).

[147] Samuelson and Scotchmer (n 76) 1626.

[148] *Sega Enterprises v Accolade* (n72) 1518.

[149] Commission, 'Report from the Commission to the Council, the European Parliament and the Economic and Social Committee on the implementation and effects of Directive 91/250/EEC on the legal protection of computer programs' COM (2000) 199 final; Commission Staff Working Document.

[150] Ibid, the Community Institutions have been urged 'not to re-open the floodgate of debate on this Directive', 21.

[151] Shemtov (n 72) 152 – 161.

[152] Thomas Vinje, 'The Legislative History of the EC Software Directive' in Colin Tapper and Michael Lehmann (eds), *Handbook of European Software Law* (OUP 1993).

program.[153] The need to strike a balance between control and openness is recognised[154] but the means of converting this recognition into a reliable model to identify the pivot's position is not yet available. Rather than taking the purist approach, which might be doctrinally correct but which would meet with over-riding resistance, the guiding principle should be the art of the possible. It would be best to take a step approach starting with the minimum intervention and evaluating its impact.

A form of step approach was advocated by Rooijen in the form of a regulator with rulemaking, dispute resolution and monitory powers. To achieve the middle ground between openness and control would require the use of levers such as lifting the ban on sharing decompiled code or shifting the burden of proof for availability of the interface information to the rightsholder.[155] The regulator would not be called upon to carry out any market-specific analysis.[156] Regulators are used in some Member States to regulate Technology Protection Measures (TPMs) under the InfoSoc Directive[157] and telecommunications.[158] As a form of step by step approach, where the impact of the change can be monitored to achieve the optimum balance, the approach has advantages but it is unclear how it would be implemented. Devolving to the regulator powers to vary the rules would involve a wholesale change to highly contentious provisions of Article 6 of the Software Directive and could lead to uncertainty if the regulator changes the rules too often. It is unclear whether any execution of the rulemaking task would be effective at Europe wide level or only national or even specific case levels. Nevertheless the approach has merit and as with the proposal for a specific exclusion for code in interfaces[159] it should be considered if the following recommendation to permit sharing of interface specifications does not achieve the required balance and improve interoperability.

5.4 *Restrictions on sharing information*

Article 6 of the Software Directive permits decompilation where it is indispensable to achieve interoperability of an independently created computer program provided inter alia the information obtained is not 'to be given to others, except when necessary for the interoperability of the independently created computer program'.[160]

Article 6 (2) (b) was submitted by the French delegation in April 1990 and considered by the Council working group.[161] It was adopted in the final directive and prevents the dissemination of information obtained from reverse analysis even when that information is

[153] Interviews (n 14).

[154] Samuelson and Scotchmer (n 76) 1608.

[155] Rooijen (n 43) 226-233.

[156] Ibid, 231.

[157] Council Directive 2001/29/EC on the harmonisation of certain aspects of copyright and related rights in the information society [2001] L167/10.

[158] Electronic Communications Directive (n 89).

[159] Section 4.4.

[160] Software Directive Art 6 (2) (b).

[161] Vinje (n 152).

not covered by copyright. The literature on the adoption of the Software Directive makes little mention of this provision and it does not appear to have met with opposition. Energies were perhaps understandably concentrated on resisting other proposals such as preventing reverse engineering interface information being used to develop a competing program. When making the point that decompilation should be permitted to produce non-infringing programs regardless of whether they compete with the decompiled program, ECIS said that to do otherwise would be contrary to the fundamental copyright tenet that the ideas and principles underlying a copyrighted work are dedicated to the public.[162] To apply that principle to the non-sharing provision begs the question that if the work is dedicated to the public why should the decompiler be prevented from making it public?

Decompilation is limited to 'code' and cannot reconstitute other preparatory material. Underlying ideas and principles derived from reverse engineering may be used only so far as the provisions of the Directive permit.[163] The non-sharing clause prevents dissemination of those ideas and principles obtained by decompilation. By contrast under Article 5.3 ideas and principles derived from 'black box' engineering, namely observation, study or testing by loading, displaying, running, transmitting or storing the program, has no restriction on the sharing of the ideas and principles. It is only the information, including the interface specification, that remains protected as though it were a trade secret.

The separation of Article 6 into two parts corresponds to separate possible violations of the author's rights. Article 6.1 concerns decompiling the original program and 6.2 producing an infringing program based on the results of decompilation.[164] The Directive does not explain the nature of the obligation imposed on the decompiler in 6.2 and as most Member States merely duplicated the article no insight is given on their understanding of the obligation. Some commentators have argued that the use of the information is not a copyright issue. Access to the information contained in the program cannot however be given without a change in the normal rules of copyright. This makes it necessary to ensure that removing the 'copyright barrier' to access does not result in abuses that undermine the very protection the Directive was intended to give.[165]

This ignores certain realities. Firstly copyright protects expression which by implication means it protects something that is visible, audible or otherwise communicated to the public. It is not intended as a means to protect trade secrets. Secondly, the restricted acts under Article 4 are basically reproduction, adaptation and distribution. These are the normal acts which if undertaken with the rightholder's consent will reveal the underlying ideas and principles. The difference with computer programs is that, unlike most works that have copyright protection, the ideas and principles are not necessarily revealed when there is a legitimate reproduction, adaption or distribution of the software. The problem is that acting within the '*normal* rules of copyright' does not reveal these ideas and principles and something more and sui generis needs to happen. It is a secondary consequence that the expression does not reveal all the ideas and principles of the copyrighted work.[166]

---

[162] European Committee on Interoperable Systems ECIS : 'Why Interoperability Must Be Defined To include Competing Products' and Colombe (n 108) 327.

[163] Czarnota (n 127) 77.

[164] Ibid, 81.

[165] Ibid, 81.

[166] William Cornish, 'Inter-operable systems and copyright' (1989) 11 (11) *European Intellectual Property Review* 391-393 (emphasis added).

Thirdly, it is not a 'copyright barrier' that has to be removed but a technical hurdle that needs to be overcome to enable copyright to work in its normal manner to give a balance between protecting the expression while not giving a monopoly on the underlying ideas and principles. To restrict the purpose for which the program may be reproduced, as imposed by the Software Directive, is to restrict rights beyond that normally enjoyed.[167]

Not only are software and the Software Directive unique in keeping the ideas underlying products exempt from study,[168] but normally copyright law does not prevent the use of or dissemination of ideas or other non-copyright protected information.[169] The Software Directive however restricts the use of information, such as the interface specification, to only achieving interoperability of the independently created computer program and prevents the information being given to others except when necessary for the interoperability of the independently created computer program. To deny access to the ideas and principles underlying a computer program that are inaccessible without reverse engineering is analogous to saying that copyright protection of a book prevented the purchaser from reading it. According to the Turner Report this is inherently wrong and would have a serious restrictive effect on innovation and competition.[170]

5.5 *Encouraging the sharing of interface specifications*

The change that is recommended is to amend the Software Directive to allow for the dissemination of interface information obtained by reverse engineering. This would normally be in the form of an interface specification which is compiled from information obtained by reverse engineering. The interface specification would not contain information protected by the software supplier's copyright.[171]

Limiting the use of the information obtained by reverse engineering is not understandable from an economic standpoint.[172] Such information has the nature of a public good and welfare consideration requires that once produced there should not be exclusions of possible

[167] Ibid, examples given are that the reader of a book or the viewer of a painting is perfectly free to analyse and create; but because of the nature of software, he has first to obtain a version of the text in order to get the picture. Making an analysis for one's own experimental purposes, is just the kind that would be allowed in respect of patentable invention. Engaging in an act of reproduction (or its equivalent) for the sole purpose of evaluation, analysis, research or teaching, which by common consent is a necessary freedom when it comes to the protection of semiconductor chip topographies.

[168] Colombe (n 108).

[169] Thomas Dreier and P. Bernt Hugenholtz Eds, *Concise European Copyright Law* (Kluwer Law International 2006).

[170] Turner Report (n 21). William Cornish in 'Computer program copyright and the Berne Convention' (1990) 12(4) *European Intellectual Property Review* 129-132, 130, provides an analysis as to how the reverse engineering provisions of Article 6 comply with Article 9(2) of the Bern Convention. The Convention permits exemptions to Article 9 (1), which requires Union countries to give authors exclusive rights to the reproduction of their literary and artistic works, only (1) in certain special cases, (2) provided that such reproduction does not conflict with a normal exploitation of the work and (3) does not unreasonably prejudice the legitimate interests of the author.

[171] *SAS Institute;* Interviews (n 14) and technical discussions and conversations between author and software design engineers 2014.

[172] Schmidtchen (n 106).

users from that information.[173] Presently each supplier must repeat the painstaking decompilation for itself but lifting the restriction would remove duplication of effort and avoid waste, allowing firms to specialise in providing interoperability information to other vendors. Firms would be able to innovate secure in the knowledge that an interface is available[174] while encouraging decompilation by the most efficient and specialised firms.[175] It would create a market for interface information which could encourage, but not oblige, suppliers to make their own interface information available to ensure its quality, and could also bolster the use of standard interfaces. If firms could sell the interface information obtained by reverse engineering they could recoup the costs of the process.[176] It could encourage start-up firms that specialise in this sort of information.[177] While the 'cheapest cost informer'[178] is the original developer and owner of the software, other firms which share interface specifications they acquire by reverse engineering would also be efficient. By stopping multiple competitors from collaborating on their reverse engineering efforts it leaves reverse engineering as a viable option only to large developers. This makes it harder for smaller firms to enter and compete in the market.[179] The pressure group SAGE[180] recognised that an exception permitting research and analysis could disadvantage small companies which could not afford to conduct reverse engineering.[181]

This market response will no doubt be resisted by many in the software industry with the same vehemence displayed when the Software Directive was introduced. There is concern that:

> 'The sharing of interface specifications could evolve into a complex pattern of unmanageable (sub) licencing arrangements. It could be prohibitively difficult for the rightsholder to ascertain whether use of his or her interoperability information by a third party stems from a valid (sub)license or whether it was obtained in a different manner and, therefore, possibly constitutes an infringement'[182]

Properly decompiled interface specifications do not contain code or other copyright material belonging to the software rightsholder. There is no doctrinal justification for the rightsholder being able to prevent its dissemination. Legitimate conduct cannot be prevented just because

---

[173] Ibid, 425.

[174] Ann Walsh, '*Microsoft v Commission*: interoperability, emerging standards and innovation in the software industry' in Rubini, L, (ed) *Microsoft on Trial* (Edward Elgar 2010), 296- 297.

[175] Rooijen (n 43) 230.

[176] Schmidtchen (n 106) 425, the cost of decompilation should be as small as possible.

[177] Van Den Bergh, 'The Role of Social Justification of Copyright: A 'Law and Economics' Approach' (1998) *Intellectual Property Quarterly* 17, 30 and Schmidtchen (n 106).

[178] Schmidtchen (n 106).

[179] Rooijen (n 43) 90.

[180] Dominant American software companies established a group called the Software Action Group for Europe (SAGE).

[181] Vinje (n 152) 51.

[182] Ibid, 221.

it is difficult to distinguish from illegitimate conduct.[183] As will be seen the recommendation provides for a conservative step approach and encourages registration which could allow the rightsholder some element of monitoring of its interfaces. This limited modification of Article 6 is intended to move a moderate step towards more openness. It is also hoped the increase in access to interface specifications will challenge software rightsholders to make the interface information available themselves.

The safeguards that have been imposed by Article 6 (1) can remain unaltered.[184] Allowing the sharing of interface specifications obtained by reverse engineering would not require any dilution of these restrictions. The original compiler would still be required to have a licence or other permission, to ensure the information was not already available and to limit the reverse analysis to only what was necessary for interoperability. Indeed the dissemination of interface specifications could reduce the need for reverse engineering as more information would be available. This could reduce the opportunity or excuse for looking at parts of programs wider than the interfaces.

With regard to Article 6(2), reverse analysis could still be limited to cases where the original decompiler has an independently created computer program but the original decompiler could then make the interface specification available to others. Ideally however the requirement for an independently created computer program would be removed to allow for software engineers to carry out decompilation purely to write the interface specification. This certainly happens in the software community[185] and encouraging the practice would help disseminate interface information. The restriction that information should not be 'used for the development, production or marketing of a computer program substantially similar in its expression, or for any other act which infringes copyright' would remain valid. The statement is really superfluous anyway as nothing in Article 6 or elsewhere permits the use of copyright material.[186]

Some decompilers may not want to share the interface specifications as they may want to retain the competitive advantage the information gives them.[187] The possibility of sharing interface specifications was discussed during industry interviews and several interviewees expressed interest in the possibility.[188] Some also expressed surprise that it was not already permitted as they know of interface specifications being shared in the software community.[189]

---

[183] It is the same reason for reverse analysis not infringing the third requirement of the Berne Convention to not unreasonably prejudice the legitimate interests of the author, Cornish (n 170)

[184] Article 6 (1) requires that decompilation is carried out by someone who has a license or other right to use a copy of the program; the information necessary to achieve the interoperability has not previously been readily available; and the acts are confined to parts of the original program which are necessary to achieve interoperability.

[185] Interviews (n 14). Many software engineers already decompile and share interface information unaware the sharing of the information is unlawful. They realise they are not using the code but information they have discovered and recorded themselves and do not understand the justification for them not being able to use that information as they wish.

[186] Article 6 does not even give an express right for the decompiler to use decompiled information. The origins of the clause and its significance are discussed in Vinje (n 130) 83

[187] Samuelson & Scotchmer (n 76) 1658, who considered that reverse engineers typically keep the resulting know-how secret for competitive advantage.

[188] Interviews (n 14).

[189] Ibid.

In practice interfaces of individual computer programs would only have to be completely decompiled once. When new versions appeared decompilation could be limited to those aspects that seem novel from running and observing the program. If interface specifications from the original decompilation were continually and widely available, updating would be less time consuming and expensive.[190]

The wider software industry does have a pattern of sharing information, not least in the open source community. If there was a market for the interface information it would be possible for smaller developers to buy the interface information which would remove the burden of reverse analysis. This could allow smaller developers to enter and compete in the industry. A market for interface specifications minimises 'deadweight loss' of administration and potentially enforcement. The availability of the information would mean there was less need to embark on reverse analysis. Permitting the sharing of interface specifications is a form of 'prosumer law'[191] which allows the user of the internet, or in this instance the computer programmer, to have an active role in obtaining interoperability including the ability to exit the program with their data.[192]

The rationale behind Article 6 was said to be to encourage the rightsholder to voluntarily disclose the interface specification but this does not appear to have had a significant impact.[193] If however decompilers can make that information available then this can either relieve the rightsholder of the burden of preparing the information for disclosure, or may encourage them to disclose the information themselves as they may feel more in control of the situation.

5.6 *Specific amendments to Article 6*

No amendment is required to Article 6.1 which can remain unaltered. [194]

While Article 6.1(b) can remain unaltered it has been criticised as it is uncertain when information is 'readily available'. Is it legitimate to charge for the information and if so how much and how complete must the information be? Shemtov proposes inserting the words 'readily available under fair and reasonable terms'.[195] This would be a sensible amendment

---

[190] Ibid.

[191] The term 'prosumer' refers to the online creator who is not just a consumer Brown (n 71) 184.

[192] Ibid, 185. Here the prosumer could choose to exit if they were not locked-in to the supplier's software.

[193] Only 24% of respondents to the Commission Staff Working Document consultation thought the possibility of reverse engineering of their software represented an incentive to license interoperability information. No evidence of any incentive was revealed in 3D CAD industry interviews.

[194] Article 6 (1) The authorisation of the rightholder shall not be required where reproduction of the code and translation of its form within the meaning of points (a) and (b) of Article 4(1) are indispensable to obtain the information necessary to achieve the interoperability of an independently created computer program with other programs, provided that the following conditions are met:

(a) those acts are performed by the licensee or by another person having a right to use a copy of a program, or on their behalf by a person authorised to do so;

(b) the information necessary to achieve interoperability has not previously been readily available to the persons referred to in point (a); and

(c) those acts are confined to the parts of the original program which are necessary in order to achieve interoperability.

[195] Shemtov (n 72) 150.

but not essential to the working of this recommendation. Article 6.1 (c) has been criticised as until decompilation takes place it is not possible to know exactly what part of the code makes up the interface and needs to be decompiled. It is likely that some extraneous code could be decompiled. Inserting the word reasonable in the text so that it reads 'acts are performed to the extent *reasonably* necessary to achieve interoperability of the independently created computer program' could address this. However this amendment is not essential to this recommendation. To implement this recommendation no amendment would be needed to Article 6.1.

Article 6.2 amended to read:[196]

2. The provisions of paragraph 1 shall not permit the information obtained through its application to be used for goals other than to achieve the interoperability of ***an*** *independently created computer program.*

Article 6.1 refers to the interoperability of ***an*** independently created computer program while Article 6.2 refers to the interoperability of ***the*** independently created computer program. Replacing *the* with *an* in Article 6.2 removes the requirement for the decompiler to have already developed a specific computer program. This would allow for decompilation by software engineers with the aim of licensing or otherwise distributing the interface specifications they have created. Because of the difficulty of reverse engineering there is unlikely to be a rush of independent decompilers. However during interviews some interest was expressed for this model, although not from the 3D CAD suppliers themselves. The model could prove an efficient way of making the interface information available[197] although some concern was also expressed that this could create competitive standards which would not be efficient.[198]

Article 6.2(b) *(not)…to be given to others, except when necessary for the interoperability of the independently created computer program*; ***would be deleted in its entirety***

This would remove the restriction on the sharing of the interface specification.

Article 6.2 c does not appear to add anything but can be included to avoid the doubt that the rest of Article 6 in some way gives an implied licence to infringe the copyright in the decompiled software.

These amendments would allow for dissemination of the non-copyright protected information. They do not relieve the restrictions that exist in Article 6.1 that limit decompilation to interoperability nor do they give any right under the Software Directive or other legal provision to infringe another parties intellectual property rights, other than the existing right to reproduce or translate software for the purpose of interoperability.

---

[196] Article 6 (2) The provisions of paragraph 1 shall not permit the information obtained through its application:
(a) to be used for goals other than to achieve the interoperability of the independently created computer program;
(b) to be given to others, except when necessary for the interoperability of the independently created computer program; or
(c) to be used for the development, production or marketing of a computer program substantially similar in its expression, or for any other act which infringes copyright.

[197] Interviews (n 14).

[198] Ibid.

5.7 *Registration and publication of interface specifications*

A registry of interface specifications is proposed with the aims of: encouraging decompilers to make their information available; improving knowledge as to what interface information is available, informing rightsholders when their interfaces have publically available specifications. The registry would publish sufficient information on the interface specification to allow anyone searching the registry to identify the software interface that had been decompiled. Interfaces of computer programs do not normally change substantially, although when new versions are released there will be some changes. Once a version of the interface is registered and available the specification can be re-used, and combined with decompilation of changes to the latest version to give a current interface, without repeating the original work.[199]

The rationale for the reverse engineering provisions in Article 6 was to encourage rightsholders to make interface information available. The dissemination of interface specifications would be further pressure on the rightsholders to this end. Indeed there is no reason why rightsholders should not avail themselves of the register. This would give notice that the interface information was available. It would address the difficulty identified in the Commission Staff Working Document of how to find and obtain interface information[200] and help with one of the non-legislatives measures the Document recommended to improve advertising of the availability of interoperability information.[201] Publication will also support the Digital Single Market strategy to make better use of standards.[202]

5.8 *Safe Harbour*

Registration is not compulsory, but to encourage the use of the registry, decompilers who notify the registrar that they have created an interface will be relieved of normal liability for infringement of copyright. Where the decompiler complies in good faith[203] with the provisions of Article 6 the rightsholder will not be entitled to an injunction nor to damages. To meet this requirement decompilers will be expected to keep, and when required disclose, records of the decompilation process. The only remedy that will be available to the rightsholder is to claim a royalty on a FRAND basis for any copyright that might have unwittingly been included in the interface specification.

Ideally the changes to copyright law in the Software Directive would be accompanied by the introduction of a patent interoperability exception. This exception would state that, where the

---

[199] Ibid.

[200] Commission Staff Working Document, 17.

[201] Ibid, 15.

[202] Standards have been identified as a critical area to maximise growth of the digital economy and various policy initiatives promote better use of standards including in public procurement, Commission Staff Working Document 'Against lock-in: building open ICT systems by making better use of standards in public procurement' COM (2013) 455 final; Commissions Digital Single Market strategy (n 48).

[203] Good faith is used for while there is no general principle of good faith in English law there are judicial interpretations of the concept and it's meaning is more established in other European Union member states. The concept is used in the safe harbour provision of competition law, for example Commission Notice (2014/C 291/01) on agreements of minor importance which do not appreciably restrict competition under Article 101(1) of the Treaty on the Functioning of the European Union (De Minimis Notice).

results of decompilation or other reverse engineering for interoperability cannot be used because they are covered by a patent, such patents could not be invoked against products implementing the information.[204] Alternatively the patent holder's right would be to claim a royalty based on FRAND terms.[205] As this arrangement only applies to interfaces and is intended to improve interoperability with minimum cost and regulation, a royalty free arrangement is preferable. This would avoid the cost and delay of parties negotiating royalties on prices valued prior to the code becoming a standard interface. This is considered to be difficult to achieve in the absence of further regulation.[206] It is envisaged that the right to claim any FRAND royalties will be normally be invoked only when the interface specification infringes a patent right. Interface specifications will generally not include copyright protected material such as code and there is little evidence of royalty claims for copyright in standards.[207] This may also apply to interfaces as they are de facto standards.[208]

The patent exception is not essential to this recommendation but it would remove the concern that, while responsible decompilation and dissemination of interface specifications avoids liability for copyright, it could still face claims of patent infringement. If however a patent exemption could be brought into the regime a further amendment could not only encourage rightsholders to disclose interfaces on the register but also deter litigation by patent trolls. It would provide that if the rightsholder makes interface information on their own software available through the register and it subsequently transpires the interface infringes a third party patent the safe harbour against normal intellectual property claims could protect the registering party. This would make the rightsholder liable at worst for FRAND royalties. This would help protect registered rightsholders from claims by patent trolls. Patent trolls are attracted to de facto software standards as infringement is easier to identify.[209] If the rightsholder registered the interface the trolls would not be able to threaten injunctions or high damages claims so convincingly and would have to settle for a FRAND royalty.[210]

5.9 *Contractual preference for registered interfaces*

Another benefit that would flow from the introduction of the registry of interface specifications is that industry could prefer software where the interfaces are registered. This

[204] Commission Staff Working Document, 11

[205] Ibid, 11; a proposal that would also increase the available information is for publication of the source code of patented programs, Declaye and Leister (n 36) 331.

[206] Josh Lerner and Jean Tirole, *Standard-Essential Patents (2014)* Working Paper IDEI-803, 3.

[207] Following the *Microsoft* case and the settlement in 2009, Microsoft still continued to charge royalties for its interface patents but not for non-patented interface information. http://www.microsoft.com/openspecifications/en/us/programs/default.aspx (accessed 20 December 2014).

[208] Blind (n 50), 11.

[209] Nicholas Saunders, 'Litigation of patents essential to technical standards – what is the future for patent trolls?' CIPPM Spring Lecture 25 April 2013.

[210] As 3D CAD Software is sold globally the jurisdictional issues of the protection would need to be considered. Patent trolls might be able to avoid the protection of the safe harbour by issuing proceedings outside the protected jurisdiction.

could be specified in contracts, including industry standard form contracts.[211] This would start to bring the pressure to increase openness that is seen by procurement practices of public authorities[212] into the private commercial environment. This aspect could be more influential than any other aspect of the recommendation. It is clear from industry interviews that public authorities specify open standards and will give preference to bids that will improve compatibility. This requirement influences the behaviour of their suppliers to design their products using open standards.[213] However public authorities are able to share information and collaborate to improve their knowledge and practices on using open standards. As commercial enterprises conventionally compete, collaborating on open standards is harder. By defining open standard software in contracts as including software whose interfaces are registered and available either for free or on FRAND terms, commercial enterprises can increase the use of compatible software in an efficient and effective manner.

5.10 *Registration platforms*

Interface specifications could be posted on a register similar to the European Federated Interoperability Repository (EFIR) which includes the Joinup platform with a catalogue to enable Member States and the European Commission to document and share their solutions to interoperability.[214] These initiatives are presently focused on public administration but the concept could be available to allow for recording and sharing of interface specifications for all commercial software. This could be done, preferably by extending the existing repository, as some software has uses in both the public and private sphere, or a separate register for software used by the commercial sector. This has the attraction of a one stop shop across Europe.

However as it is proposed that registration will have the legal implications of relieving some liability it may be more appropriate for registration to be with the intellectual property offices. These offices already have in place systems to record when registration takes place and to ensure the integrity of the data. There would be no need to inspect or examine the information that is registered. The register's purpose is to publicise the existence of the interface information and provide a safe harbour to the registering party. All negotiations and transactions concerning the information will be done directly between whoever registered the information and the prospective licensee. A disadvantage of using intellectual property offices is that they are mainly nationally based and individual searches would need to be undertaken of each register. However this could be overcome if a common service such as Espacenet is used.[215]

---

[211] For example use in the Institution of Mechanical Engineers and Institution of Engineering and Technology, Model Form of Contract for the design, supply and installation of electrical, electronic and mechanical plant, MF/1.

[212] Interviews (n 14) and for procurement practice of public authorities http://standards.data.gov.uk/ (accessed 28 October 2016). From conversations it appears that construction contracts often specify the architectural design software that will be used by contractors and sub-contractors to ensure interoperability.

[213] Interviews (n 14) and conversations between author and software design engineers 2014 – 2015.

[214] This is part of the Interoperability Solutions for European Public Administrations (ISA) programme https://ec.europa.eu/isa2/home_en (accessed 26 January 2017). ICANN is an alternative worldwide register.

[215] Espacenet offers free access to more than 80 million patent documents worldwide. The EUIPO would give Europe wide coverage.

The register would increase access to interface information and improve interoperability without the cost of repeated decompilation of interfaces. All that is required is a system of registering certain information and publishing it in an ordered way. Several systems are available and could be modified appropriately. The increased dissemination of the interface information in this relatively low cost but efficient manner would help to reduce under-utilisation of the information. Transaction costs are borne by the prospective licensor and licensee which should be efficient and minimised, particularly if implemented alongside the soft measures recommended in the Commission Staff Working Document. The recommended model licences could be made available on the same website as the register of interfaces which could encourage uptake. The methodology or guidelines for assessing the value of the interoperability information will also help parties to minimise transaction costs and avoid deadweight loss.

5.11 Challenges

Challenges that have been identified to the successful implementation of the recommendation are firstly whether there is sufficient incentive to register the interfaces. To remove disincentives there should ideally be no charge for registering the specification, which should be presented in a standard form.[216] Inspection on registration would be minimal which should contain the cost of maintaining the register. To provide some financial incentive to the decompiler a system could operate allowing them to charge individual requests for access to the full registered interface specification. User may be willing to pay for the convenience of accessing specifications which have a good reputation. As the exception from copyright protection for interfaces will catch most of the information in the registered specification the ability to extract a direct monetary return from the information is uncertain. The incentive to register interface information may need to rely on the indirect benefits for decompilers of the safe harbour and for rightsholders that their software is more likely to be specified in contracts.

A further challenge is to keep pace with technological progress. The register may encourage rightsholders to increase the use of digital rights management to deter decompilation. While the anti-circumvention regime in the Information Society Directive does not apply to prevent reverse engineering of computer programs for the purpose of interoperability[217] the increasing strength of encryption means circumvention is more available to those with sophisticated technologies.[218] Also, as more software services are provided via web based systems the opportunities for meaningful decompilation are reduced. These technical challenges however underscore the need for better dissemination and sharing of the interface information to enable openness without increasing regulation.

[216] The Joinup platform is funded by the EU via the ISA project and uses the ADMS vocabulary for interoperability solutions https://joinup.ec.europa.eu/asset/adms/description

[217] Council Directive 2001/29/EC on the harmonisation of certain aspects of copyright and related rights in the information society [2001] L167/10.

[218] Hugh McCarthy, 'Decoding the Encryption Debate: Why legislation to restrict strong encryption will not resolve the "going dark" problem' (2016) 20 (3) Journal of Internet Law 17,

6. Conclusion

Despite standards and the market driven solutions of translator companies, interoperability problems persist for users of 3D CAD software. The software is vital to users to create, edit and store crucial valuable data and integrity of data is highly valued. Improvements in interoperability and openness by continuing follow on innovation must avoid disrupting the industry.

Conventional reverse engineering is tolerated as it is costly and time consuming which protects the rightsholder. This rationale does not apply to software interfaces as their value is indirect, coming from their ability to control interoperability and networks rather than their intrinsic innovation. There is less need to protect first comers as they should recoup their research and development investment from other aspects of the software rather than from the interface. The Software Directive already restricts decompilation to the purpose of interoperability, which limits software reverse engineering more than in other fields,[219] so that reverse engineering of software interfaces is legitimised by its purpose.[220] Making the reverse engineering process any harder or costly is wasteful and without welfare benefit.[221] It is not rational to protect the first comer and making access more difficult is not only inefficient but harms the consumer.

Amending Article 6 of the Software Directive to legitimise the dissemination and registration of interoperability information obtained by reverse engineering, which does not contain the software rightsholder's copyright, is doctrinally appropriate and is a modest step. Monitoring the impact of the initiative will inform whether the regulation is in the right place to balance control and openness over the correct pivot. If not, and more openness is required, then other proposals, such as allowing reuse of interface code, should be considered. While the recommendation could be effective in isolation, ideally it should be accompanied by an exception to patents. If the legislation could go further and provide a safe harbour for rightsholders who register their interfaces this could encourage registration and reduce action by patent trolls. Software with interfaces available through the register can be specified in contracts which will help commercial enterprises follow the lead set by public authorities in stipulating open standards. These initiatives combined with the 'soft' measures of model licenses and guidance on setting royalties recommended by the Commission Staff Working Document are a practical and measured step towards increased interoperability with minimum regulation and cost.

Acknowledgements

I wish to thank Stavroula Karapapa, Estelle Derclaye and Roger Brownsword for providing invaluable feedback on an earlier version of this article. The usual disclaimers apply.

[219] Samuelson and Scotchmer (n 76) 1655.

[220] Ibid, 1655.

[221] Section 5.2