BILETA
British & Irish Law, Education and Technology Association
2007 Annual Conference
Hertfordshire 16-17 April
UH

Software asks the questions: what's the law's response?

Ruth Atkins
Lecturer in Law,Department of Law & Criminology, University of Wales, Aberystwyth
Email: rda@aber.ac.uk

Abstract

As technology continues to develop with a dynamic drive towards convergence, fresh challenges are being presented which require a response from the law. Scanning the horizon of technological developments reveals that there are vital and contemporary questions to be asked of the law. An assessment of how effective the law is in response to those challenges can be made by considering the fundamental research question - why is software special? This paper addresses that question from two perspectives: first with reference to the ways in which software can be supplied and; second by examining the nature of software defects. From documenting the methods by which software may be made available and studying the nature of problems which may arise from its implementation and operation, it is noted how software has distinct characteristics which the legal framework must accommodate within its structure. The convergence of current technologies and of those predicted for future creation and production, exemplifies how these questions must be continually re-examined in a present day and forward-thinking manner. The conclusion is drawn that although software does raise questions which are demanding and challenging, the response from the law thus far, is weak and unconvincing.

Background

Predictions for significant developments which will affect IT lawyers in the future include an increased reliance on technology by businesses and the accompanying need for security and protection.[1] Recent technological innovations include BlackBerry[2] (a gadget for wireless communications which can be used to email, page and phone) and the convergence of devices is increasingly seen in the latest creations. For example, the distinction between computers and television sets is eroding further, as televisions are being used for e-mailing and browsing and computers are increasingly used for downloading and viewing content.[3] The 'convergence of mobile phones into entertainment centres' also illustrates how the divide between devices has diminished further, as 'big-screen content migrates to small-screen devices'.[4] Thus, as technology continues to develop through diversification and through convergence, fresh challenges are presented which require a response from the law and this paper considers such challenges.

Foundations of this study stem from a current work in progress which examines the issues in their broadest context: offering a critical analysis of the scope of contractual liability for defective software and a consideration of the rights and responsibilities of a software supplier. It is within this field of research that a fundamental question may be raised - why is software special? The paper answers this question from two perspectives: first with reference to the ways in which software can be supplied and; second by examining the nature of software defects. From documenting the methods by which software may be made available and

[1] See further, Desai (2006) 'Predictions 2006: More Technology, More Protection', *Comps. & Law* Vol. 16(5) p 12.

[2] For an early review on the Blackberry wireless communications experience, see Eastham (2003) 'Blackberry', *Comps. & Law,* Vol. 14(5) pp. 14 – 15.

[3] Davies (2007) 'Predictions 2007 – IT Law', *Comps & Law* 17(5) p. 31.

[4] Garrod (2006) 'Mobile TV: Content will be King' *Comps. & Law,* Vol. 16(6) pp. 32-35.

studying the nature of problems which may arise from its implementation and operation, it is noted how software has distinctive characteristics which the legal framework must accommodate within its structure. The extent to which the law is tested by the challenges posed by software and its supply, coupled with an assessment of the law's effectiveness to respond to those challenges, is the central theme of analysis within this paper.

Supply of software

Software is typically supplied under a licence and the scope of the licence and the contractual basis upon which it rests will need to be determined. A licence is quite simply 'formal authority to do something that would otherwise be unlawful'.[5] The terms of the software licence prescribe the permissions of use which the owner of the rights subsisting in the software is willing to grant to the user of the software. Such rights will invariably be granted in exchange for a licence fee, the payment of which will confer upon the licensee a licence which may last for a lengthy fixed period, for example, 99 years, or perhaps one with a perpetual duration. Clearly, by its nature, a perpetual licence will not end unless either party seeks to terminate on the grounds of a breach.

The traditional view is that software is supplied under a licence and therefore the focus is on the copyright in the software with the terms of licence granting permission to use the copyright-protected work. Of course, it is possible that software procurement may include an element of the law of sale, for example, if the software is supplied on a disk, the disk will be sold outright. However, the software itself will be governed under the terms of a licence, which will define the scope of permitted use but which will not confer outright ownership. Against this background, it is apparent to see why it may be preferable within a software licence agreement to use the term 'licensee' or 'end user' rather than 'customer' which may indicate rights beyond those which are being conferred. Thus the term 'user' rather than 'customer' could be seen as a more accurate reflection of the extent of the rights being conferred and as suggested by Warner, 'for the software industry there is some merit behind this piece of semantics'.[6]

In recent years there has been a move away from the use of perpetual licences towards fixed term licences operating a power unit pricing model.[7] One of the main exponents within the software industry has been Microsoft.[8] Previously Microsoft's 'common corporate user licence involved a perpetual use agreement, where users paid an annual fee for three years, at the end of which they had the choice of renewing the licence and upgrading the software or carrying on using the existing software at no extra charge'.[9] Under the new, subscription-based licensing model, 'the company guarantees to upgrade software but users do not have the right to continue using software once the agreement expires'.[10] This change in practice met with considerable resistance from users who have perceived their rights to be diminishing[11] and whilst in most circumstances users will benefit from being able to upgrade more frequently, they will nonetheless 'ultimately pay more in licensing fees'.[12]

In response Microsoft has drawn attention to the benefits which their subscription-based licensing model can confer upon users. For example, the new subscription model is advertised to: simplify the upgrade process; provide customers with more flexibility; be cost-

[5] Oxford Dictionary of Law (2003, 5th ed.)

[6] Warner (2002) 'Licence Hunting', *Comps. & Law* Vol. 13(1) p. 28

[7] For a fuller discussion of the different pricing models, see further Barker (2000) 'Whose software is it anyway?' *Computing*, available from: http://www.wnunet.com/News/1115156

[8] See further http://microsoft.com/licensing

[9] Nash (2001) 'New Microsoft licences – short-term gain, long-term pain' *Computer Weekly,* 11 May 2001, http://www.computerweekly.co.uk

[10] Nash (2001) 'New Microsoft licences – short-term gain, long-term pain' *Computer Weekly,* 11 May 2001, http://www.computerweekly.co.uk

[11] See further 'Users square up for Microsoft licence war' and 'Software users voice their fears' *Computer weekly*, 19 July 2001, http://www.computerweekly.co.uk

[12] Nash (2001) 'New Microsoft licences – short-term gain, long-term pain' *Computer Weekly,* 11 May 2001, http://www.computerweekly.co.uk

effective in the short term;[13] offer the ability to minimise cash flow with lower annual payments and; improve workplace productivity.[14] With the trend for subscription licences re-emerging in the industry[15] it is important that users enter into such licences in full awareness of the scope of their rights and permissions conferred by the licence. Thus, it must be remembered that a subscription licence, in contrast to a traditional licence bought in perpetuity with a one-off payment, will require the exercise of the renewal option and this will need to be exercised at regular intervals. As a note of caution, users must be mindful of a key disadvantage of a subscription licence that the user may have difficulties withdrawing from an agreement, i.e. if a user wishes to use a particular product, it must continue to subscribe and it cannot take a break from the agreement.[16] To counterbalance this position it is important for users to 'be mindful of support clauses' and to 'build in other remedies' into the agreement, rather than making a mistake by relying wholly on the right of termination.[17]

At a practical level the status of a user under this type of arrangement will undoubtedly change. Rather than, for example, paying for software under a three-year licence on a per-user basis and after having paid the licence fees being able to retain the software, this model will require the user either to pay for another term at the end of the licence period, or to discontinue using the software.[18]

The subscription model approach to licensing is representative of a move toward the renting of software.[19] The potential change in the status of the user from 'perpetual licensee' to 'subscription licensee', or even 'lessee' and the differences such changes will have to the contractual relationship between the supplier and the user are interesting to note. From a broad perspective, it could be suggested that a move towards software rental rather than acquisition under a perpetual licence may lead to greater transparency in the industry as a whole: it may encourage an increased awareness on the part of users as to the actual scope of their rights within the framework of the contractual relationship. To permit the use of software under terms more restrictive than a perpetual licence, may prove to be more effective in conveying the message to users that they do not become the owner of the software. For this reason it may prove more effective, at ground level, in discouraging potential violations of the intellectual property rights of the software creator and/or the supplier. For the software user, knowledge of the true contractual relationship may be more readily acquired and understood.

Moving towards a rental model for software supply, one can note the obvious advantages for software suppliers. A main benefit of renting rather than selling is the stream of revenue which may be generated. There are benefits for software users though, for as Barker notes, rental may 'remove the incentive for suppliers to push their customers onto new versions of software every couple of years'.[20] Thus, software rental may bring about 'a smoother upgrade path and more predictable financial outgoings for customer'.[21] However, a cautionary note can be added by referring to the earlier practices of IBM in the 1960s and 70s when this supplier was dominant in the market and could be seen to be in a position to 'force users to pay through the nose'. [22] Against this background, the current trend towards software rental

[13] *Ibid.*

[14] Warner (2002) 'Licence Hunting', *Comps. & Law* Vol. 13(1) p. 31, see further http://microsoft.com/licensing.

[15] Beckett (2006) 'Licensing: How buyers can flex their muscles' *Computer Weekly,* 25 July 2006, http://www.computerweekly.co.uk

[16] This is identified as a major disadvantage of a subscription licence by Russell Fowler, ICT Technical Support Manager for Surrey Police Force. See further Beckett (2006) 'Licensing: How buyers can flex their muscles' *Computer Weekly,* 25 July 2006, http://www.computerweekly.co.uk

[17] Advice from the leading IT lawyer, Kit Burden, Partner at Barlow, Lyde & Gilbert. See further Beckett (2006) 'Licensing: How buyers can flex their muscles' *Computer Weekly,* 25 July 2006, http://www.computerweekly.co.uk

[18] Warner (2002) 'Licence Hunting', *Comps. & Law* Vol. 13(1) p. 29

[19] See further, Dale (2007) 'Uncovering the Mysteries of Disclosure', *Comps. & Law* Vol. 17(5) p. 20 in which the author particularly recommends renting as a favourable option for law firms in the future.

[20] Barker (2000) 'Whose software is it anyway?' *Computing*, available from: http://www.wnunet.com/News/1115156

[21] *Ibid.*

[22] *Ibid.*

may be viewed with some scepticism. Yet, it is evident that software rental is predicted to become of greater significance in the near future and recommendations to engage in this method of software acquisition can be found emanating not only from market providers but from leading IT law practitioners too.[23]

Whichever licensing framework for supplying software is used, it is evident that a variety of types of software may be supplied under that agreement. This may range from, for example specialist bespoke programs through to general customised commercial software, to, at the far end of the spectrum, mass-marketed standard software packages. The type of software may be a factor to consider when determining the manner through which contractual liability for that software should arise. For example, bespoke software for a complex project may be governed by a detailed, extensively negotiated, individualistic agreement between the supplier and customer, whereas in the case of the supply of off-the-shelf standard software, shrink-wrap licensing may be employed to establish a direct contractual relationship between the parties.

The use of shrink-wrap licences has been a popular method for contracting with end users employed by the software industry. Here, the terms of the licence are usually printed clearly on a sticky seal, or are readable on the box which is covered with a shrink-wrap film.[24] Within the box are contained the disks and manual and the contention is that, once an acquirer has opened the shrink-wrap they have entered into a licence agreement upon the terms stated in the licence.[25] Under UK law, several questions concerning the enforceability of such licences have been raised and the legal position is far from being resolved.

In the only UK case to consider the validity of shrink-wrap licences, the Scottish Court of Session in Beta Computers v Adobe Systems[26] ruled that a shrink-wrap licence was enforceable. Here, a company, Adobe Systems telephoned a supplier, Beta Computers, and ordered a standard software package to upgrade their existing software. The software was produced by a third party. When the software was delivered, the terms of the end user licence were visible through the wrapping, although they could not be fully read. On the package it stated 'Opening the Informix S.I software package indicates your acceptance of these terms and conditions'.[27] Adobe did not want to take the risk and attempted to return the package unopened. The suppliers refused to accept its return and sued for payment of the price.

The court based its ruling on three findings. First, that the supply of proprietary software for a price was a single contract *sui generis*, containing elements of nominate contracts such as sale and the grant of a licence.[28] Second, *consensus in idem* of the contract could not be reached until the conditions of use were accepted by the parties and this could not have occurred earlier than when Beta provided Adobe with the conditions.[29] Third, that in any event,[30] as the conditions of use were not acceptable, the defenders were entitled to reject the software.[31] In reaching its decision, the court placed considerable emphasis on the rights of the customer to return the software should the licence prove unsatisfactory.[32]

[23] See Dale (2007) 'Uncovering the Mysteries of Disclosure', *Comps. & Law* Vol. 17(5) p. 20

[24] Oxford Dictionary of Computing (2004, 5th ed.)

[25] Oxford Dictionary of Computing (2004, 5th ed.)

[26] *Beta Computers (Europe) Ltd v Adobe Systems (Europe) Ltd* [1996] S.L.T. 604

[27] *Beta Computers (Europe) Ltd v Adobe Systems (Europe) Ltd* [1996] S.L.T. 604 at 605

[28] *Beta Computers (Europe) Ltd v Adobe Systems (Europe) Ltd* [1996] S.L.T. 604 at 609

[29] *Beta Computers (Europe) Ltd v Adobe Systems (Europe) Ltd* [1996] S.L.T. 604 at 610 and 611

[30] Thus, whether the tender by the supplier of software subject to conditions of use was regarded as a breach of a previously unconditional contract, or as being subject to an implied suspensive condition entitling the defenders to reject the software if the conditions of use were acceptable, or as being made when there was no concluded contract, the defenders were entitled to reject the software.

[31] *Beta Computers (Europe) Ltd v Adobe Systems (Europe) Ltd* [1996] S.L.T. 604 at 612

[32] See further Johnson, (2003) 'All wrapped up? A review of the enforceability of 'shrink-wrap' and 'click-wrap' licences in the United Kingdom and the United States', *European Intellectual Property Review,* 25(2), 98-102 at 101.

The fact that this case was decided under Scottish law, rather than having been heard in an English court is one which must be remembered. In particular, as Johnson summarises, the software producer's third party rights within the case, were held to be enforceable under the uniquely Scottish principle of *jus quaestium tertio*, which permits the creation of contractual rights for the benefit of third parties.[33] At the time of the ruling, under English law, on the basis that the software producer did not contract directly with the end user, the requirement of privity of contract would have prevented the software producer from enforcing contractual rights against them.[34]

Nonetheless, the court's ruling in Beta emphasised how important it was to the industry as a whole, that effect should be given to software licence conditions supplied in a shrink-wrap format. Following this decision, the legal enforceability of such licences under English law was found to be repeatedly questioned in an academic context but not called upon to be tested in the courts.[35]

It can be suggested that the doubts surrounding the enforceability of shrink-wrap licences have been responded to by the industry. This has been achieved through the introduction and increasingly widespread usage of click-wrap licences and web-wrap licences. These licences function by requiring a higher level of affirmative action on the part of the user, for example, users are required to click 'I accept' the end user licence terms before permission is granted to download or install the software.[36]

Yet each of the possible approaches towards licensing software (shrink-wrap, click-wrap and web-wrap) continues to generate debate as to whether such agreements are legally valid and should be enforceable. If there is apparent consent to the agreement on the part of the user[37] and if the agreement itself adheres to traditional contract law principles then requirements should be met. It can be seen that the unique nature of software and the demands placed on the legal framework within which it operates are reflected in these distinctive approaches to licensing. For this reason, these approaches have raised particular questions for the law to address. Owing to a limited number of cases which have been tested in the courts on such issues and in the absence of directly applicable legislative measures, it is suggested that the response of the law has been somewhat limited. Rather it is maintained that it is in the absence of a clear statement of the legal position in this area that it has fallen to the software industry to tackle the issue for itself. This has been attempted by the suppliers by developing the use of click-wrap and web-wrap licensing, methods which require a positive action by the user and thus may be relied upon to demonstrate an intention to accept the terms and conditions. It is hoped that in response to recent technological developments, the law will rise to the challenges presented and satisfactorily match such technological innovations with an appropriate response. This could occur through any of a number of ways. It may occur through legislative intervention and the introduction of specific measures designed to address the challenges. In addition, or alternatively, it may fall to the courts to address the situation and this could be done through the application of traditional contract law principles to the circumstances created by software transactions, on a best fit basis. On the other hand, the software industry may develop its own solutions, which through the passage of time and repeated usage could develop into customary practices to be referred to and relied upon.

[33] Johnson, (2003) 'All wrapped up? A review of the enforceability of 'shrink-wrap' and 'click-wrap' licences in the United Kingdom and the United States', *European Intellectual Property Review,* 25(2), 98-102 at 101.

[34] This is a position which has been amended since the introduction of the Contracts (Rights of Third Parties) Act 1999.

[35] See for example, Johnson, (2003) 'All wrapped up? A review of the enforceability of 'shrink-wrap' and 'click-wrap' licences in the United Kingdom and the United States', *European Intellectual Property Review,* 25(2), 98-102; Lai (1999) 'Recent developments in copyright, database protection and (online) licensing, *I.J.L.&I.T.* 1999, 7(1), 73 – 94; Gringras (1996) 'The validity of shrink-wrap licences', *I.J.L.&I.T.* 1996, 4(2), 77-111.

[36] See further Warner (2002) 'Licence Hunting', *Comps. & Law* Vol. 13(1) p. 29

[37] There are third party rights issues to be considered here also, but this is beyond the scope of the current discussion.

Nature of software defects[38]

The particular characteristics of software are also highlighted when one considers the nature of software defects. Software defects can occur either at the design stage, in which case there will be a failure in each copy of the product; or at the production stage when the defective copies will be more limited in number.[39] Although this may be in common with the supply of tangible products, an important distinguishing feature of software is that although 'program testing can be used very convincingly to show the presence of bugs, [it can] never demonstrate their absence, because the number of cases one can actually try is absolutely negligible compared with the possible number of cases'.[40] Therefore in relation to software testing, it is often cited that it is 'impossible to test even the simplest program in an exhaustive fashion'.[41] This undoubtedly has a number of consequences for the supplier, the end user, and also for the legal framework which should be in place to protect the rights and define the responsibilities of the contracting parties. Given the considerable permutations that running a computer program may generate, in the event of a problem with the software, it will be necessary for the parties to be in a position to determine whether fixing the problem falls within, or is beyond the scope of the supplier's existing responsibilities.

Here it can be noted that the issue of maintenance of software could be of considerable significance to the parties. Particularly when there is a supply of a complex bespoke system, the parties are likely to have in place a maintenance agreement where for example, error correction is performed by the supplier in accordance with a cascade of severity levels and a timeframe for the work to be carried out. This aspect of software supply emphasises another feature particular to software contracts which is that the supply of software may not be merely a one-off transaction between the parties but rather the start of a continuing working relationship. The scope and extent of the responsibility of a supplier to maintain the software and to provide software enhancements should be addressed within the contractual documentation.

Of course, it is recognised that the contractual agreement is unlikely to address every possibility and therefore it is at this stage that the law could offer greater certainty to the parties, for example, by defining the standards of performance expected. Standards of performance may be defined expressly by the parties or imposed through terms implied into their agreement. However, the question of whether such standards may be deemed appropriate remains largely unresolved at law and the debate, for example as to whether software should be categorised as goods or services, is a continuing one.[42] Following on from this, the extent to which the parties may restrict or exclude their liability for any failure to meet the required standard, is similarly an area which poses many challenges for the law to address. Although it is evident that if software is supplied in the course of development it cannot be expected to be bug-free,[43] the precise scope of a supplier's responsibility and the extent to which their liability may be limited is uncertain.[44] This matter could be discussed at great length but it is merely touched upon here for the simple purpose of identifying it as another area which presents particular issues for the law to address when responding to the demands of software. In the event, it is hoped that this brief summary conveys something of the nature of software defects and contributes towards noting the attributes which make software special.

[38] This conference paper is a work in progress and the author recognises that far more material could be included within the remit of this topic. It is hoped that at this stage, some of the main issues are identified. The author would welcome comments of other aspects which may be of interest and could be included in the general discussion of what makes software special.[Please email: rda@aber.ac.uk]

[39] Lloyd (2004, 4th ed.) *Information Technology Law* at p. 570.

[40] Dijkstra (1974) EWD 361: Programming as a discipline of mathematical nature, *Am. Math. Monthly,* 81, 6: 608 – 612.

[41] Lloyd (2004, 4th ed.) *Information Technology Law* at p. 571.

[42] See further, Poyton (2005) 'Dematerialized Goods and Liability in the Electronic Environment: The Truth is, 'There is no Spoon (Box)', *I.R.L.C.T* 2005, 19(1), 83-98; and Green & Saidov (2007) 'Software as Goods', *JBL* Mar, 161-181.

[43] *St. Albans City and District Council v. International Computers Ltd* [1996] 4 All ER 481 at 487.

[44] See further, Macdonald (2005) 'Bugs and Breaches'*, I.J.L.& I.T.*, Vol. 13(1) pp. 118-138 and Callaghan & O'Sullivan (2005) 'Who should bear the cost of software bugs?' *C.L.S.R.*, Vol. 21, 56-60.

Conclusion

As technology continues to evolve and the convergence of devices expands into new technological spheres, the fundamental question of ‘what makes software special?’ remains a relevant one. Answering the question will continue to draw awareness to the key challenges which software poses for the law. Contemporary examples which further reveal the nature of software and the range of possible methods for supply of software offer additional examples to test the law. Moving towards the future and mindful of the predicted growth in technological convergence, the need for an appropriate response from the law is increasingly necessary. The response of the law to date has been limited and on an ad hoc basis and yet it is evident that unique characteristics of software do raise particular issues and questions for the law to address. Some of these questions have been identified in this paper and it is observed that many of these remain unanswered.