THE LAST PLACE IN THE WORLD

COPYRIGHT PROTECTION FOR COMPUTER SOFTWARE IN NEW ZEALAND

*Brendan Cash**

Computer technology - whether it is hardware, software or the internet - is the headache that just won't go away for anyone grappling with the law of intellectual property. With the Copyright Act 1994 the New Zealand legislature attempted to meet some of the challenges but it largely avoided the issue, passing the headache on to the judiciary should any tricky issues arise. One such issue is copyright protection for computer programs. The Copyright Act 1994 makes it clear that computer programs are protected as a literary work but how far does that protection extend (or how far do we carry the fiction that a computer program is a literary work)? This article attempts to second guess what level of protection our courts might provide, drawing on decisions from the United Sates, the Commonwealth and New Zealand. It argues that New Zealand has shown a willingness to protect functional items and would protect the "look and feel" of a program, providing a level of protection which courts in the United States have considered undesirable.

I   INTRODUCTION

Would New Zealand's copyright law protect Lotus' 1-2-3 menu command hierarchy?[1] That the source code of Lotus 1-2-3 would receive copyright protection as a "literary" work was made clear in *IBM v Computer Imports Ltd.*[2] Moreover, the point received statutory confirmation in section 2(1) of the Copyright Act 1994, which includes computer

---

[1]   See *Lotus Development Corp v Borland International Inc* 49 F 3d 807 (1st Cir 1995) ("Borland").

[2]   [1989] 2 NZLR 395.

programs as a form of "literary works". However, the Act fails to settle whether such "non-literal" elements as Lotus' menu command hierarchy should be protected. Therefore, the scope of copyright protection under the new Act has been left to the courts.

Should New Zealand's courts extend protection to a program's non-literal elements?[3] This question raises two issues. The first is primarily legal and looks to established copyright principles to establish what level of protection to provide. The second issue is more policy-orientated and is concerned with what level of copyright protection is necessary in order to promote the development of the software industry.[4]

This article focuses on the legal issue raised by the copyrightability of a computer program's non-literal elements. In the absence of any guidelines from the New Zealand legislature the courts should look to well established principles of copyright in order to establish which aspects of a computer program should be copyrightable. The article will examine the approaches developed in the United States and in the Commonwealth and will look at their applicability in New Zealand.

This article will suggest that, while the scope of protection has been declining in the United States, New Zealand's willingness to afford protection to "functional items" and to the ideas they embody indicates that our courts may provide extensive protection to a computer program's non-literal elements.

In essence this article argues that our law is similar to that expressed in the American cases of *Whelan Associates v Jaslow Dental Laboratory Inc*[5] and *Lotus Development Corp v Paperback Software International.*[6] These two cases are generally seen as the high point of copyright protection for computer software since they provided protection to the "look and feel" of a program. A similar approach to copyright protection has been confirmed by our own Court of Appeal.[7]

This article will not directly address the policy issues raised by the copyrightability of computer programs. The policy issues in this area are complex and cannot be answered on a purely theoretical level. On the one hand computer programs may need a great deal of protection because of the ease with which they may be copied. On the other hand existing

---

[3]   No case involving non-literal copying of a computer program has yet been dealt with by New Zealand courts.

[4]   M Morgan "Trash Talking: The Protection of Intellectual Property Rights in Computer Software" (1994) 26 Ottawa LR 425, 430.

[5]   797 F 2d 1222 (3d Cir 1986) ("Whelan").

[6]   740 F Supp 37 (D Mass 1990) ("Paperback").

[7]   *Bleiman v News Media (Auckland) Ltd* [1994] 2 NZLR 673 ("Bleiman").

financial incentives may be enough to stimulate innovation, thereby reducing the need for protection.[8]

However, this article acknowledges that there are important policy issues to be addressed. The application of copyright law to computer software raises fundamental policy issues which need to be addressed by the legislature. Further study must be conducted into such matters as the relationship between copyright protection and software innovation, the needs of software developers and the interests of consumers.[9] This is just one of the myriad of issues that our legislature will need to address in the information age.

This article is divided into five main parts. Part I will give a brief outline of the nature of computer programs and the programming process. Part II will examine the decisions on this point in the United States and the tests which the courts have proposed. Part III will look at the leading decisions in the Commonwealth; the decisions in *Autodesk v Dayson*,[10] *Ibcos Computers Ltd v Barclays Mercantile and DB Poole*[11] and *Delrina Corp v Toilet Systems Inc.*[12] Part IV will outline the likely approach to the protection under New Zealand law, while, finally, Part V will conclude the article, tying the various issues together, and commenting on some of the policy issues raised by this subject.

II   COMPUTER PROGRAMS

In order to understand the issues raised by the copyrightability of computer programs it is necessary to have a basic understanding of computer programs. This section will provide a brief guide to both the process of programming and the nature of programs themselves.

Computer programs are essentially "sequences of instructions designed to manipulate a computer in a particular way".[13] In developing a program a programer goes through a series of steps. Julian Velasco identifies five programing steps.[14]

---

[8]   See J Velasco "The Copyrightability of Non-literal Elements of Computer Programs" (1994) 94 Colum LR 242, 283.

[9]   See MP Larvick "Questioning the Necessity of Copyright Protection for Software Interfaces" [1994] 1 U Ill L Rev 187.

[10]   (1992) 104 ALR 563 ("Autodesk").

[11]   [1994] FSR 275 (Ibcos").

[12]   (1993) 47 CPR (3d) 1 ("Delrina").

[13]   DWT Daniels "Learned Hand Never Played Nintendo: A Better Way to Think about the Non-literal, Non-visual Software Copyright Cases" (1994) 61 U Chicago L Rev 613, 615.

[14]   Above n 8, 245-256.

- Firstly, the programer identifies the task he or she wishes the software to perform. For example the programr may wish to create an electronic spreadsheet.

- Secondly, the programer identifies the particular functions needed to meet the user's demands. Using the example of the spreadsheet, the programer might want it to present statistical results.

- Thirdly, the programer designs the overall structure of the program by breaking down the functions into smaller and smaller subtasks or subroutines until the overall task is broken down into manageable pieces.

- Fourthly, the programer must implement the design of the program by actually writing the program's code. This is done first of all in a higher-level language such as "BASIC". This is known as the source code. The source code is then translated into binary code which the computer can read. This is known as object code.

- The final step in the process is "debugging". This is the process in which errors are identified and corrected. This is an ongoing process since errors may appear at any time during a program's life.[15]

As we can see a program's written code (source or object) makes up only one part of a program and programmng process.[16] A key part of a program is its design or structure - "the hierarchical division of tasks into subtasks, the interaction of one module with another, and the way data passes from one part of the program to another".[17] This is what we might call the non-literal elements of a program.

However, the concept of a program's non-literal elements extends beyond its basic structure. It includes its user interface - the mechanism through which the user operates a program. The user interface is an important part of any program since it determines the ease with which it can be used, which in turn helps to determine how successful a program is.[18] The menu command hierarchy on Lotus' 1-2-3 is a classic example of a successful user interface.

Computer programers clearly have a vested interest in protecting more than just a program's source or object - ie its literal elements. A key part of a program, and a large part of the programing process, is the design or structure of a program and the way it interacts with the user. Programers can, and do, copy the user interface or structure of successful

---

[15]   See n 13, 615-617, for further discussion on the nature of computer software.

[16]   Above n 8, 247.

[17]   Above n 13, 617.

[18]   Above n 8, 247.

programs without actually copying the literal code.[19] However, while an important part of programs, should the non-literal elements of a program be protected?

II THE PROTECTION OF NON-LITERAL ELEMENTS IN THE UNITED STATES

The relevant decisions in the United States have dealt with two questions. The first is whether the non-literal elements of computer programs can be protected by copyright at all. This is the broad question of copyrightability.[20] The second question is in regards to what test you adopt, having decided that the non-literal elements can be protected, for copyrightability and substantial similarity in individual cases. This is the narrow question of copyrightability.[21] This section will examine the different decisions reached in the United States in regards to both these issues.

A The Broad Question of Copyrightability

In the decision *Whelan Associates Inc v Jaslow Dental Laboratory*[22] the Third Circuit held that the non-literal elements of a computer program could be protected. This decision on the broad question of copyrightability has been followed in a number of cases. However, with the more recent decision in *Lotus Development Corp v Borland International Inc*[23] the copyrightability of a program's non-literal elements has been thrown into doubt, especially the copyrightability of user interfaces.

Whelan was the first case to tackle the broad question of copyrightability. *Whelan* was a case which involved a custom designed record-keeping program for dental laboratories, the non-literal elements of which had allegedly been copied.

The primary issue was whether the "structure (or sequence and organisation) of a program"[24] could receive copyright protection. The Court of Appeal held that since:[25]

> The copyrights of other literary works can be infringed even when there is no substantial similarity between the works, it would thus appear that the copyrights of computer programs can be infringed even absent copying of the literal elements of the program.

---

19   Above n 8, 248.

20   TP Nicolas "GUI Wars" (1994) 47 Ark L Rev 139, 141.

21   Above n 20.

22   Above n 5.

23   Above n 1.

24   Above n 5, 1224.

25   Above n 5, 1234.

Whelan's decision to extend protection beyond a computer program's literal elements was criticised. Some commentators argued that a program's structure is more akin to its idea than its expression, and therefore should not be protected.[26] However, the decision to protect a program's non-literal elements was generally seen as uncontroversial and has found widespread support.

In *Johnson Contracts Inc v Phoenix Contract Systems Inc*,[27] the Ninth Circuit asserted:[28]

> Whether the non-literal components of a program, including the structure, sequence and organisation and user interface, are protected depends on whether, on the particular facts of each case, the component in question qualifies as an expression of an idea, or an idea itself.

Lotus Development Corp v Paperback Software International[29] also confirmed the decision in *Whelan* on the broad question of copyrightability. *Paperback* involved the copying of the menu command hierarchy in Lotus' spreadsheet program 1-2-3. Therefore, unlike the court in *Whelan*, which was concerned only with the copying of a program's structure, sequence and organisation, *Paperback* was concerned with the copyrightability of a user interface.

The key issue in *Paperback* was whether Lotus' user interface, a key non-literal element of the program, could be protected by copyright. The Massachusetts District Court, adopting the reasoning in *Whelan*, said that the user interface, as a non-literal element of a program, could be protected. The court noted "that this type of copying of non-literal expression, if sufficiently extensive, has never been upheld as permissible copying; rather, it has always been viewed as copying of elements of an expression of creative originality".[30]

Computer Associates International Ltd v Altai Inc,[31] the definitive case in this area of copyright law, appeared to lay the matter to rest. Judge Walker characterised *Altai* as

---

[26]   See above n 8, 262; PS Menell "An Analysis of the Scope of Copyright Protection for Application Programs" (1989) Stan Law R 1045; SR England "Note, Idea, Process or Protected Expression?" (1990) Mich L R 866; KR Moon "Review of Legal Protection Computer and Semi Conductor Technologies" in Ministry of Commerce "Review of Industrial Property Rights Patents, Trade Marks and Designs" (Wellington).

[27]   866 F 2d 1173 (9th Cir 1989).

[28]   Above n 27, 1175.

[29]   Above n 6.

[30]   Above n 6, 51.

[31]   982 F 2d 693 (2d Cir 1992). The defendant had hired an employee of the plaintiff to adapt the defendants scheduling program to a different IBM operating system. The employee did so, making use of the plaintiff's copyrighted ADAPTER source code. After suit was filed, the

dealing with the "challenging question of whether and to what extent the non-literal aspects of a computer program, that is, those aspects not reduced to a written code, are protected by copyright".[32]

Following *Whelan*, the court in *Altai* acknowledged the powerful analogy that if the non-literal structures of literary works are protected by copyright; and if computer programs are literary works, as provided for by the legislature; then the non-literal structures of computer programs are protected by copyright.

However, while endorsing *Whelan's* approach to the broad question of copyrightability the court in *Altai* criticised the test formulated in *Whelan* to decide the narrow issue of copyrightability. In this regard the decision in *Altai* is markedly different from that in either *Whelan* or *Paperback*. *Altai* represents the first of a number of cases which reduced the level of copyright protection for computer software as a result of the test adopted for the narrow issue of copyrightability.

The scope of protection has been further narrowed by the recent decision of the First Circuit in *Borland*. *Borland* was an appeal from a District Court decision holding that *Borland* had infringed Lotus' copyright in its spreadsheet program 1-2-3 when it copied the menu command hierarchy into its spreadsheet programs Quattro and Quattro Pro. *Borland* is therefore another case dealing with the copyrightability of user interfaces. The question on appeal was whether Lotus' menu command hierarchy was copyrightable.[33]

The court's answer was that a menu command hierarchy was excluded from copyright protection since it was a method of operation.[34] It compared the menu command hierarchy of 1-2-3 to the buttons on a VCR. Like the buttons on a VCR the menu command hierarchy was the means by which the program was operated.[35] As such, regardless of how many ways a menu command hierarchy might be expressed or how original it was, it could not be protected.

This conclusion was driven by the consideration that a computer program is a functional item. Circuit Judge Boudin expressed the point that as a literary work "the

---

defendant rewrote the copied portion of OSCAR, as the program was known. The issue was whether there was still non-literal copying.

[32]   Above n 31, 696.

[33]   Above n 23, 806. See also *Lotus Development Corp v Borland* 133 L Ed 2d 610, 116 Ct 804 (1996). The court split equally on the matter and as a result the decision of the Circuit Court of Appeals was affirmed.

[34]   Above n 23, 815.

[35]   Above n 23, 817.

problem presented by computer programs is fundamentally different ... [t]he computer program is a means for causing something to happen; it has a mechanical utility, an instrumental role, in accomplishing the world's work."[36]   While the court noted that functionality did not act as a bar to copyright protection it did state that it altered the calculation (against protection).[37]

Therefore, while *Whelan, Paperback* and *Altai* are key cases which have confirmed the copyrightability of a computer program's non-literal elements, including a program's user interface, the decision in *Borland* casts doubt over whether the user interface, a key non-literal element of a program, can ever be protected by copyright. By its very nature the user interface is the way in which a computer program is operated.   Therefore, all user interfaces would appear to be "methods of operation", and hence not able to be protected. The protection available to computer programs has also been narrowed by the tests for copyrightability and substantial similarity. These tests will now be examined.

B   The Narrow Issue of Copyrightability

Whelan and *Paperback* represent the high point of copyright protection for computer software. They protected the "look and feel" of a program. However, since the decision in *Altai* the courts have adopted a process of analytic dissection which has greatly reduced the degree of protection.

In *Whelan* the court constructed the following test for distinguishing between unprotected idea and protected expression. "[T]he purpose or function of a utilitarian work would be the work's idea, and everything that is not necessary to that purpose or function would be part of the expression of that idea."[38]

Applying this test the court found that the purpose of the program in *Whelan* was to aid in the operations of a dental laboratory and that the program's structure was not essential to that purpose since similar software performed the same function with different structures.[39] As a result the structure of the program in *Whelan* was expression, rather than idea.

Altai criticised the test in *Whelan* as being too simplistic since it assumed that only one idea underlies any computer program. The reality is that programs are usually a complex

---

36   Above n 23, 819.

37   Above n 23, 819.

38   Above n 5, 1236.

39   Above n 5, 1237.

structure of sub-routines, each with its own idea.[40]   The test was also criticised as providing too much protection since whatever is not idea can be protected as expression.[41]

With its three part test for the narrow issue of copyrightability the decision in *Paperback* was an advance on *Whelan*, particularly in its adoption of Judge Hands' abstraction test. The test is as follows:[42]

> FIRST, ... the decision maker must focus on alternatives ... along the scale from the most generalised conception to the most particularised, and choose some formulation - some conception or definition of the idea - for the purpose of distinguishing between the idea and its expression ...

> SECOND, the decision maker must focus upon whether an alleged expression of the idea is limited to elements essential to expression of that idea ... or instead includes identifiable elements of expression not essential to every expression of that idea.

> THIRD, having identified elements of expression not essential to every expression of the idea, the decision maker must focus on whether these elements are a substantial part of the allegedly copyrightable work.

By adopting this test the decision in *Paperback* recognised that more than one idea could underlie a computer program.

However, while an advance on *Whelan*, the test in *Paperback* is not fundamentally different.[43] The abstractions test would not look out of place in *Whelan* since the it simply seeks to distinguish between idea and expression. It was this distinction which formed the foundation of the test in *Whelan*.[44]   Moreover, in applying its three part test the court in *Paperback* refused to dissect the allegedly infringed program and instead dealt with it as a whole. While holding that the "look and feel" concept was not significantly helpful,[45] the court did state that:[46]

> to determine copyrightability, a court need not - and indeed should not - dissect every element of the allegedly infringing work. Rather, the court need only identify those elements

---

[40]   Above n 32, 703.

[41]   Above n 8, 261.

[42]   Above n 6, 60-61.

[43]   Above n 20, 146; above n 8, 275.

[44]   Above n 8, 275-276.

[45]   Above n 6, 62-63.

[46]   Above n 6, 67.

that are copyrightable, and then determine whether those elements, considered as a whole, have been impermissibly copied.

Indeed, when applying the second part of its test to the facts the court concluded that the menu structure:[47]

> taken as a whole - including the choice of command terms, the structure and order of those terms, their presentation on the screen, and the long prompts - is an aspect of 1-2-3 that is not present in every expression of an electronic spreadsheet.

A similar approach can be found in *Whelan*. As David Daniels observes, the rule in *Whelan* also establishes a guide to the what structural elements a judge should compare for copyright purposes.[48] The guide indicates that a judge may compare structure at any level of abstraction, however high - ie the work as a whole. The court in *Whelan* states that it is "concerned with overall similarities between the programs".[49]

By examining the programs as a whole, the tests in *Whelan* and *Paperback* favour the finding of copyright protection. As a result both decisions are seen as the high-point of copyright protection for computer programs.[50] With the decision in *Altai* there was a shift towards analytical dissection with a corresponding decline in the protection for computer software.

The abstractions test, which forms a part of the test in *Paperback*, also forms the basis of the abstraction-filtration-comparison test adopted in *Altai*. However, the test is fundamentally different from that in *Paperback* as it undertakes the process of dissection which the court in *Paperback* expressly rejected.

The test in *Altai* directs that a trial court should:[51]

(1) 'break down the allegedly infringed program into its constituent structural parts' and then

(2) 'sift out all non-protectable material' so as to

---

[47]   Above n 6.

[48]   Above n 13, 629.

[49]   Above n 5, 1246.

[50]   Above n 20, 145.

[51]   See JE Brown "'Analytical Dissection' of Copyrighted Computer Software - Complicating the Simple and Confounding the Complex" (1995) 11 Computer Law and Practice 16, 19; Brown was counsel for Apple in its litigation against Microsoft and he acknowledges his bias against the process of analytic dissection which occurs in *Altai* and which was adopted, to the disadvantage of Apple, in its litigation against Microsoft.

(3) confine comparison and protection to the 'kernel, or possibl[y] kernels, of creative expression [left] after following this process of elimination'.

Under the rigours of this test the court in *Altai* found that only a few elements of the allegedly infringed program - a few parameter lists and macros - could withstand scrutiny and be protected by copyright.[52] Since these were only minor elements of the program the court held that a finding of infringement could not be maintained.[53]

The decision in *Altai* has received a mixed response. One commentator praised *Altai* as "a significant advance in establishing, for software copyright cases, a method for determining infringement that is both consistent with traditional copyright law and workable".[54] The approach of the Second Circuit in *Altai* has been applied by the Ninth Circuit in *Sega Enterprises Ltd v Accolaide Inc*,[55] by the Federal Circuit in *Atari Games Corp v Nintendo of America*[56] and by the Tenth Circuit in *Gales Publisher Co v Bando Chem Industries Ltd*.[57]

The test in *Altai* has also received approval in the most recent Court of Appeal decision on this issue. In *Borland* the First Circuit's Court of Appeal held that, while they did not find *Altai* helpful in deciding whether the "literal copying of a menu command hierarchy constitutes copyright infringement", the "*Altai* test may provide a useful framework for assessing the alleged non-literal copying of computer code".[58]

Critics of *Altai* have, however, identified the case as a "legal Chernobyl"[59] due to the drastic reduction in protection it provides for a computer program's non-literal elements. Critics argue that by "dissecting" a program and removing individually unprotected items before comparing it to the allegedly infringing work the court may in fact miss or eliminate

---

[52]   Above n 32, 714.

[53]   Above n 32, 715.

[54]   D Bender "*Computer Associates v Altai*: Rationality Prevails" [1992] The Computer Lawyer 1, 1.

[55]   24 USPQ 1561 (9th Cir. 1992); Ninth Circuit approved and applied *Altai* when considering the degree of protection which could be accorded to video games under the 'fair use' doctrine.

[56]   975 F 2d 832 (Fed Cir 1992); Approving and applying the test in *Altai* the Federal Circuit held that Atari had copied the security system in Nintendo's entertainment system.

[57]   9 F 3d 823 (10th Cir 1993); Tenth Circuit endorsed *Altai's* test as a way of determining whether menus and sorting criteria are copyrightable.

[58]   Above n 1, 814.

[59]   A L Clapes & J M els "Revenge of the Luddites: A Closer Look at *Computer Associates v Altai*" [1992] The Computer Lawyer 11, 11.

protectable elements, such as the way in which various elements are organised or combined.[60]

The narrowing of protection in *Altai* seems to arise out of the court's scepticism about the appropriateness of using copyright law to protect computer programs. The court adopts the abstraction-filtration-comparison test as a result of its characterisation of computer programs as being "highly functional utilitarian" items which can have "only one or two forms of expression", making them unworthy of copyright protection.[61]

The decision in *Altai* thus illustrates a marked shift away from the judicial attitudes illustrated in *Whelan* and *Paperback*. While in *Paperback* the court held "that elements of expression, even if embodied in useful articles, are copyrightable",[62] the court in *Altai* clearly doubts that a useful article such as a computer program can have any "copyrightable" expression.

A similar attitude permeates the trial court decision of *Apple Computer Inc v Microsoft Corp*,[63] a decision which was upheld by the Ninth Circuit Court of Appeal.[64] The court engaged in a form of "analytic dissection" based on a conception of a computer similar to that in *Altai*.[65] The court believed that computer interfaces "serve a purely functional purpose in the same way that the visual displays and user commands of the dashboard, steering wheel, gear shift, brakes, clutch and accelerator serve as the user interface of an automobile."[66]

The court dissected MacIntosh's user interface into its smallest elements and viewed each element in isolation. "Not surprisingly", Apple's counsel Jack E Brown noted, "[the court] concluded that almost every element was unprotectable as unoriginal, not an artistic expression or otherwise not protectable."[67]

---

[60]   A M Fitzgerald "Square Pegs and Round Holes" (1993) 4 J Law and Information Science 142, 147.

[61]   Above n 52, 19.

[62]   Above n 6, 54.

[63]   799 F Supp 1006 (ND Cal 1992). Apple sued Microsoft to protect the audio-visual works that constituted the graphical user interface on screens of Apple's MacIntosh computers. A key issue in the case was to what extent Apple had licensed the use of its user interface to Microsoft. As a result a process of analytic dissection was inevitable.

[64]   *Apple Computer v Microsoft Corp* 35 F 2D 1435 (9th Cir 1994).

[65]   Above n 51, 20.

[66]   Above n 63, 1023.

[67]   Above n 52, 21.

In *Microsoft* the court also dealt a blow to the concept of "look and feel" - of protecting a program as a whole, as the courts did in *Whelan* and *Paperback*. The court expressly stated that:[68]

> Without dispute, a copier may not make immaterial variations and thereby escape copyright liability ... and to this limited extent, there is some legitimacy for a "look and feel test" ... but this test should be applied only after protectable expression has been identified, not before, as Apple would have this court do.

The process of "analytic dissection" undertaken in *Microsoft* drew upon the Ninth Circuit's decision in *Brown Bag Software v Symantec Corporation*.[69] *Brown Bag* accepted the copyrightability of a program's non-literal elements and proposed a test which, while not identical to that in *Altai*, is compatible with it.[70]

Brown Bag set up a bifurcated test for "substantial similarity". The first part of the test is an extrinsic test which involves an objective analysis of expression.[71] In this phase the program is analytically dissected and unprotectable elements of expression are eliminated in order to define the scope of copyright protection.[72] The second phase of the test is the intrinsic test. It is only triggered when the court finds objective similarity in protected expression. This phase measures substantial similarity depending on the response of the ordinary person.[73]

The approach adopted in *Brown Bag*, and followed in *Microsoft*, is not fundamentally different from the test in *Altai*. All three tests dissect a program, remove the unprotectable elements, and then compare the "kernels" of original expression. Moreover, all the tests significantly reduce the protection available to computer software and arise as a result of the view that computers are functional items that do not merit protection.

In stark contrast, the earlier decisions in *Whelan* and *Paperback* extended protection to the "look and feel" of a computer program, providing extensive copyright protection to a

---

[68]   Above n 63, 1026.

[69]   960 F 2d 1465 (9th Cir 1992). An independent programmer developed an out-lining program which was sold to Brown Bag Software. The programmer then developed a competing product which was sold to Symantec. Brown Bag sued alleging copyright infringement. Accepting the copyrightability of a program's non-literal elements, the key issue was whether copyright in the programs non-literal elements had been infringed.

[70]   Above n 8, 279.

[71]   Above n 8, 267.

[72]   Above n 8, 267.

[73]   Above n 8, 267-268.

computer program's structure, sequence, organisation and user interface. These decisions were driven by the view that computer programs, regardless of their functional nature, did deserve protection where protectable expression could be separated from unprotectable idea.

C   Conclusion

With the decisions in *Whelan* and *Paperback*, the courts accepted that a program's non-literal elements - its structure through to its user interface; could be protected. These cases then adopted tests that provided protection to the "look and feel" of a program, expressly rejecting the process of analytical dissection.

However, while the courts in *Altai*, *Brown Bag* and *Microsoft* all accepted the copyrightability of a program's non-literal elements, including the user interface, they developed tests for the narrow issue of copyrightability which greatly reduced the level of protection. This shift in the level of protection arose largely out of the court's view that computer programs are utilitarian items which merit only minimal protection.

Finally, with the Court of Appeal decision in *Borland*, the functional view of computers has led effectively to the denial of protection to a computer's user interface. In answering the broad question of copyrightability (in regards to user interfaces) the court in *Borland*, in stark contrast to the court in *Paperback*, denied protection.

III   THE PROTECTION OF NON-LITERAL ELEMENTS IN THE COMMONWEALTH

In trying to establish whether and to what extent New Zealand should protect the non-literal elements of computer programs, some assistance may be derived from three decisions on this issue within the Commonwealth - the Australian High Court decision in *Autodesk Inv v Dayson*,[74] the English High Court decision in *Ibcos Computers Ltd v Barclays Mercantile and DB Poole*[75] and the decision of the Ontario General Division in *Delrina Inc v Triolet Systems Inc.*[76]

A   Autodesk Inc v Dayson

Autodesk involved the computer program AutoCAD and its accompanying hardware device called an "AutoCAD lock", a device without which the program could not operate.[77]

---

[74]   Above n 10.

[75]   Above n 11.

[76]   Above n 12.

[77]   Above n 10, 563.

The defendant, after examining the lock, produced his own version.[78] This lock incorporated a 127-bit look up table which comprised a substantial part of Widget C, a program within AutoCAD, that operated in conjunction with the lock. The owners of the copyright in AutoCAD sued alleging that the defendant had infringed their copyright in AutoCAD.

The High Court of Australia held the production by the defendant of his lock constituted an infringement of the copyright which the plaintiffs held in the AutoCAD program.[79] This was because a substantial part of Widget C was its 127-bit look up table and its reproduction in the defendant's lock was a reproduction of a substantial part of AutoCAD which was literary work in respect of which the copyright was vested in the plaintiffs.[80]

Given that the source and object code of Widget C had not been copied, but only the program's 127-bit look up table, it would appear that the court in *Autodesk* was extending protection to a program's non-literal elements, although this is not crystal clear. The court did hold, however that:[81]

> The definition of 'computer program' in s10(1) of the Act should not be understood as referring only to an actual written expression or representation of a set of instructions but it should be understood as conferring protection upon a set of instructions itself.

As a result, even though the defendant's lock had not copied the source code of the Widget C program, it had copied its 127-bit table and it therefore infringed the copyright the plaintiff held in its AutoCAD program. It would therefore appear that the High Court in *Autodesk* effectively extended copyright protection in Australia to include a program's non-literal elements.

B Ibcos Computers Ltd v Barclays Mercantile and DB Poole

In *Ibcos* the English High Court considered the substance and extent of the protection copyright provides for a computer program. Although *Ibcos* dealt primarily with the copying of source code it nonetheless commented on whether a program's non-literal elements would be protected. The court also considered the value of the test for substantial similarity developed in *Altai*.

---

[78] Above n 10, 565-570.

[79] Above n 10, 575.

[80] Above n 10, 573-74.

[81] Above n 10, 564.

The court in *Ibcos* dealt with statements from *John Richardson Computers v Flanders Chemtec Ltd* which approved the test in *Altai*. In *John Richardson* the court stated:[82]

> There is nothing in an English case which conflicts with the general approach adopted in the *Computer Associates* case ... at the stage at which the question of substantial similarity of any copying falls to be assessed ... the question which has to be asked ... is essentially the same question ... address[ed] in *Computer Associates*.

However, the court in *Ibcos* rejected the approach in *Altai* as generally unhelpful. The search for a core of protectable expression had no place in English law. In England the court would "first decide whether the plaintiff's program as a whole is entitled to copyright protection".[83] Moreover, in *Ibcos* the court noted that English copyright law draws no distinction between aesthetic and functional works. The courts are prepared to protect the "detailed" (as opposed to the "general") ideas of any work.[84]

The court in *Ibcos* did, however, agree with the decision in *John Richardson* that the consideration of copyright infringement is not restricted to the text of the code. The court stated:[85]

> Most literary copyright works involve both literal matter (the exact words of a novel or computer program) and varying levels of abstraction (plot, more or less detailed of a novel, general structure of a computer program). I therefore think it right to have regard in this case not only to what Mr Turner called the 'literal similarities' but also to what he called 'program structure' and design features.

In its examination of the program involved, the court concluded that, while the compilation of programs should receive copyright protection, other features of the program did not since they were at a level of generality that did not attract protection.[86]

C   Delrina Inc v Triolet Systems Inc

Delrina involved a former employee who, while employed by the plaintiff, had developed a performance monitoring program. After leaving the plaintiff's employment, the employee developed a competing product which was marketed by the defendant.[87] *Delrina*

---

[82]   Above n 11, 302.

[83]   Above n 11, 302.

[84]   Above n 11, 292.

[85]   Above n 11, 302.

[86]   Above n 11, 304-305.

[87]   Above n 4, 460.

dealt mainly with the issue of literal copying, however, the court did declare that the non-literal elements of a computer program could be protected. The court stated:[88]

> I conclude that under US copyright law and based on statutory provisions equivalent to those in the Canadian Copyright Act, copyright is extended to all parts of computer programs ... The attitude of the US Copyright Office and that of the US courts in *Computer Associates* and *Lotus* strengthen my conclusion that both the literal and non-literal portion of the Sysview program could be protected by copyright under the Copyright Act.

In addition to extending protection to the non-literal elements of a computer program *Delrina* adopted the test for copyrightability from *Altai* and *Microsoft*. The court stated:[89]

> Whether a Canadian court should adopt the abstraction-filtration-comparison method in deciding an action for copyright infringement or some other method, it seems clear that before a computer program or some part of it can be held to be copyrightable, some method must be found to weed out or remove from copyright protection those portions which, for the various reasons already mentioned, cannot be protected by copyright. After the portions that are not copyrightable have been filtered out, there may not be any kernels or golden nuggets left to which copyright can attach.

Delrina's reliance on the analysis in *Altai* and *Microsoft* seems to arise out of its acceptance of the American view that "even if the expression originated with the author, the expression of the idea is not copyrightable if the expression does no more than embody elements of the idea that are functional in the utilitarian sense."[90] This is a curious position for a Canadian court to take since traditionally Canadian law, like English law, has not drawn any distinction between functional and aesthetic items when dealing with copyright issues.[91]

D Conclusion

These three Commonwealth decisions are clear authority for the proposition that copyright protection can extend to a program's non-literal elements. However, the decisions do not provide the same degree of protection. The Australian High Court decision in *Autodesk* provides no clear guidelines as to what is, and what is not, protectable. In *Ibcos* the English High Court rejected the test from *Altai* and instead provided protection to a computer program "as a whole". In *Delrina* the Ontario General

---

[88] Above n 12, 32.

[89] Above n 12, 37.

[90] Above n 12, 41.

[91] Above n 12, 462.

Division adopted wholeheartedly the analysis in *Altai* and *Microsoft* that computer programs are functional items which contain only a core of protectable expression.

IV   THE PROTECTION OF NON-LITERAL ELEMENTS IN NEW ZEALAND

The protection of non-literal elements in New Zealand involves the same questions that have arisen in the United States. Firstly should copyright protection to non-literal elements be extended at all. This identified as the broad question of copyrightability. It is accepted that the non-literal elements can be protected in New Zealand, the second questions relates to what test or approach should be adopted in respect of the narrow issue of copyrightability and substantial similarity in individual cases.

Both these questions shall be answered by reference to New Zealand case law in the area of copyright. It will suggest that on the basis of the established copyright principles in New Zealand or alternatively extend copyright protection to a program's non-literal elements should be extended. Furthermore, it will suggest that, on the basis of the current approach to issues of copyrightability and substantial similarity, a very similar test to the "look and feel" approach developed in the cases of *Whelan* and *Paperback*, will be adopted.

A   The Broard Issue of Copyrightability in New Zealand

The leading decisions in the United States and in the Commonwealth have overwhelmingly concluded that the non-literal elements of a computer program can be protected by copyright law. However, as this article has noted, the decision in *Borland* casts doubt over these decisions, especially the decision in *Paperback* to extend copyright protection to the user interface. Should copyright protection be extended, or should it be declared that the non-literal elements of a program cannot be protected.

The decisions in *Autodesk*, *Ibcos* and *Delrina* are persuasive authority, from courts within the Commonwealth, that, as in the United States, New Zealand should extend protection. All three decisions were made under copyright statutes substantially similar, or almost identical to our own. Section 14(1)(a) and section 2(1) of the Copyright Act 1994 state that:[92]

> 14   Copyright in original works —
>
>> (1)   Copyright is a property right that exists ... in original works of the following descriptions:
>>
>> (a) literary ... works

---

92   Copyright Act 1994, ss 2(1) and 14(1)(a).

2. Interpretation —

    (1)   In this Act ... "literary work" means any work ... that is written, spoken or sung; and includes -

    (a) A table or compilation, and

    (b) A computer program.

Section 3(1) of The Copyright, Design and Patents Act (UK) 1988 in England contains a definition of a literary work which is identical to our own. Moreover, in addition to New Zealand Copyright Act, the English Act also fails to provide any definition of what a computer program is.

The Australian Copyright Act 1968, as amended in 1984, is more comprehensive. It includes a definition of a computer program in section 10(1) of the Act and, as this article has observed, that definition played a key role in the decision in Autodesk.[93] However, the decision in *Autodesk* gave the definition in section 10(1) a broad meaning by extending protection to a program's set of instructions.

In *Delrina* the defendants argued before the Canadian courts that the definition of a computer program in their Act as a "set of instructions embodied or stored in any manner" implied that only the literal instructions of a program could be protected.[94] However, the court in *Delrina* rejected that argument, holding that non-literal elements could be protected under their Act.[95]

Since the Copyright Act 1994 does not burden the courts in New Zealand with the definition of a computer program there is no impediment to New Zealand following the decisions in *Delrina* or *Autodesk.* Moreover, the decision to extend protection in *Ibcos* was made under a copyright statute identical to our own. Therefore, the nature of the Copyright Act 1994 allows the courts in New Zealand to follow decisions in other Commonwealth jurisdictions to extend protection beyond a program's literal elements.

The decision in cases like *Whelan* and *Paperback*, as well as in *Autodesk, Ibcos* and *Delrina*, to extend protection also accords well with New Zealand's copyright case law.

A fundamental linchpin of the decisions in *Whelan, Paperback, Altai* and *Ibcos* to extend protection was the powerful analogy that if computer programs are literary works, and if the non-literal elements of other literary works are protected by copyright then the non-

---

[93]   Above n 82.

[94]   Above n 4, 460.

[95]   Above n 91.

literal elements of computer programs must also be protected.[96] This analogy would also appear to hold true for New Zealand.

In *Wilson v BCNZ*[97] Jefferies J held that a feasibility study for a television program (outlining the original concept, characters and storylines) was a dramatic work in a material form protected by copyright.[98] In reaching its conclusion, the court in *Wilson* expressly rejected a defence argument that the feasibility study was not protected by copyright as all it represented was a mere idea.[99] The court then found that the defendant had infringed the plaintiff's copyright in the feasibility study by producing a television show which took "[a]ll of the key components of the plaintiff's concept, or format ... [of] two puberty aged children using their stick and wheelchair as weapons to fight evil and do good".[100]

Wilson gives a clear indication that the courts in New Zealand are prepared to protect more than just the copying of a work's literal elements. The courts will protect a work's concept, format or key features. It should be noted that the decision in *Wilson* is tempered by the decision of the Privy Council in *Green v BCNZ*[101] which upheld the decision of our Court of Appeal[102] not to extend protection to the format of a talent quest.

In the Court of Appeal Casey J held:[103]

I have strong reservations whether such protection can be brought within the framework of our Act to prevent piracy of the basic concept underlying a series of changing dramatic works. It is difficult to see in what way the underlying idea has been distilled out and translated into an expression in a material form.

However, the decision of both the Court of Appeal and the Privy Council was premised largely on the fact that the format was only minimal in nature, involving nothing more than the repetition of a few catch phrases in an everchanging show.[104] In comparison *Wilson*

---

[96]   Above n 33 .

[97]   [1990] 2 NZLR 565 ("Wilson").

[98]   Above n 97, 572.

[99]   Above n 97, 572.

[100]   Above n 97, 572.

[101]   [1989] 3 NZLR 18 ("Green").

[102]   [1988] 2 NZLR 490.

[103]   Above n 102, 503.

[104]   Above n 101, 20.

involved a detailed description of the characters and plots for each show. In that regard *Green* seems to be merely an illustration of the point made in *Ibcos* that the courts will protect a detailed idea, but not a general idea or concept.[105]

The recognition that a copyright work may involve a series of abstractions, the most general of which are not protected, is apparent in the High Court decision in *Plix Products Ltd v Frank M Winstone (Merchants) Ltd.*[106] In *Plix Products* Prichard J drew the distinction between the general idea or concept - which is not protected - and a second form of idea which, when reduced to concrete form, may be the subject of copyright. Prichard J observed that:[107]

> The author of a work will scarcely be able to transform the basic concept into a concrete form ... without furnishing it with details of form and shape. The novelist will think of characters, dialogue, details of plot and so forth ... when these ideas (which are essentially constructive in character) are reduced to concrete form, the forms they take are where copyright resides.

This view of copyrighted works as a series of abstractions, some which are protectable and some which are not, is also apparent in the recent decision of the Court of Appeal in *Bleiman v News Media (Auckland) Ltd.*[108] In *Bleiman* Gault J stated that in drawing this distinction between idea and expression:[109]

> [i]t is all a matter of to what degree of particularity or generality the idea is taken. Clearly for infringement of copyright in a dramatic work, a play, it is unnecessary for every and movement to be precisely as in the original ... in the case of literary works inexact reproduction frequently is held to infringe. Textual reproduction is not essential.

What is clear from these cases is that in New Zealand copyright protection extends beyond literal expression. The courts have acknowledged that in any copyright work there is a series of abstractions and that, while no copyright will reside in the general idea of a work, copyright will reside in other ideas which a work expresses. This means that

---

[105]   Above n 86.

[106]   (1984) 3 IPR 390 hereafter referred to as *Plix Products*. Two marketing organisations laid down requirements for Kiwi fruit packaging based on the plaintiffs' packaging. The defendant produced packaging working off those guidelines. The issue was whether the defendants packaging infringed the copyright which the plaintiffs held in their Kiwi fruit packaging design.

[107]   Above n 106, 419.

[108]   Above n 7.

[109]   Above n 7, 678.

the powerful analogy used in *Whelan*, *Paperback*, *Altai* and *Ibcos* to extend copyright protection applies in New Zealand.

New Zealand copyright law essentially uses the celebrated "abstractions" test first penned by Judge Hand in *Nichols v Universal Pictures Corp*.[110] Judge Hand noted that:

> Upon any work ... a great number of patterns of increasing generality will fit equally well, as more and more of the incident is left out. The last may perhaps be no more than the most general statement of what the [work] is about; ... but there is a point in this series of abstractions where they are no longer protected, since otherwise the [author] could prevent the use of his "ideas", to which, apart from their expression, his property never extended.

This test is mentioned in most of the leading United Sates decisions which have extended protection to a program's non-literal elements. It forms the foundation of the test in *Altai* for copyrightability, which is the leading test in the United States.[111] Therefore, if a computer program is a literary work protected by copyright, and if the non-literal elements (abstractions, detailed ideas etc) of other copyrighted works are protected, it would appear that there is a powerful argument that New Zealand should protect the non-literal elements of computer programs.

An impediment which might be erected to deny the protection of a computer program's non-literal elements in New Zealand is that a computer program can be distinguished from a novel or play in that it is a functional item. As described above, cases in the United States, such as *Altai* and *Microsoft*, have used this argument to reduce the level of protection.[112] More recently it has formed part of the reasoning of the Court of Appeal decision in *Borland* where the court effectively denied protection to user interfaces.[113]

However, this distinction between functional and aesthetic works was clearly rejected in *Ibcos*. The court stated:[114]

> United States case law has, ever since *Baker v Selden*, been extremely careful to keep copyright out of the functional field, either by saying there is no copyright in, or that copyright cannot be infringed by taking, the functional. In *Baker* a design of ledger sheets which had a particular function was refused copyright. I doubt that would have happened here.

---

[110]   45 F 2d 119 (2d Cir 1930).

[111]   Above n 51.

[112]   Above n 62.

[113]   Above n 36.

[114]   Above n 11, 293.

The same rule appears to apply in New Zealand. Here the category of artistic works has been used to protect industrial designs and other functional and mechanistic items. Baker J in *Bendon Industries Ltd v Presslok Industries Ltd*[115] noted that "claims for breach of copyright in this court have included such diverse objects as PVC kiwifruit trays, farm goods, steel filing cabinets, frisbees and plastic tie straps".

The extent of copyright protection to such items was restricted in the Copyright Amendment Act 1985[116] and has more recently been cut back in the Copyright Act 1994.[117] However, the courts draw no distinction between functional and aesthetic works, expect where Parliament has indicated that they should do so.

In *PS Johnson Associates Ltd v Bucko Enterprises Ltd*[118] the court was confronted with the argument that, since copyright does not protect ideas, some utilitarian articles should not receive copyright protection. The example used was that of the drawing of a ball bearing. It was contended that its shape was idea and that if copyright protection was extended to such a drawing the idea of a ball bearing would be protected.[119] However, the court rejected this argument. It contented that "possibly there was insufficient time, skill, judgement and labour used in producing a drawing of so simple a character as a ball bearing".[120] In making this reply the court made it clear as long as a work is original it may receive the protection of copyright.

In answering the question of whether copyright protection should extend to a computer program's non-literal elements it is clear that, on the basis of established copyright principles, it should.

In *Ibcos, Autodesk* and *Delrina* England, Australia and Canada have extended protection under a copyright statute similar, if not identical, to the New Zealand legislation.

---

[115]   (1982) TCLR 61, 70.

[116]   Section 20A and s 20B of the Copyright Amendment Act 1985 meant that the making of any object in three dimensions would not infringe the copyright in an artistic or literary work if it reproduced part of a patent specification or when the object of reproduction had been lawfully applied industrially in New Zealand for more than sixteen years.

[117]   Section 2(1) of the Copyright Act 1994 states that there is no copyright in an artistic work which is a layout design or an integrated circuit within the meaning of the Layout Designs Act 1994.

[118]   [1975] 1 NZLR 311.

[119]   Above n 118, 320-321.

[120]   Above n 118, 321.

An examination of New Zealand case law reveals the willingness of the courts to extend protection beyond a work's literal expression to some of the ideas it embodies. In the United States, in cases such as *Whelan* and *Altai*, the courts have indicated that should copyright protection be extended beyond the actual words of other literary works then the same should be the case for computer programs.

Finally the functional nature of computer programs does not act as a legal reason for denying or limiting the protection of computer software in New Zealand. Under our law functional items, such as PVC kiwifruit trays, have been granted copyright protection. Therefore, there appears to be no legal impediment to the extension of copyright protection to a program's non-literal elements.

B   The Narrow Question of Copyrightability in New Zealand

To which non-literal elements of a computer program will protection extend? This is the narrow question of copyrightability. If a program's non-literal elements are to be protected should the abstraction-filtration-comparison test from *Altai*, the leading test in the United States be adopted, or should New Zealand adopt a test which resembles the "look and feel" approach of earlier cases?

The abstraction-filtration-comparison test developed in *Altai* has been approved in a number of subsequent decisions and has been praised by commentators.[121] It is seen as the test which best reflects the true nature of computer programs. Therefore, as the standard test in the United States, the *Altai* test would appear to be a logical choice.

Further support for its adoption comes in the form of the recommendations made by the Australian Copyright Law Review Committee (CLRC) in its draft report "Computer Software Protection".[122] The report recommends that no amendment be made to secure the protection of a program's "look and feel". Instead the CLRC supports the test in *Altai* believing that the need for standardisation and efficient user interfaces supports an inherently underprotective approach.[123]

However, regardless of its alleged merits, the approach in *Altai* does not appear to be compatible with New Zealand's case law. *Altai's* version of Judge Hand's abstraction test calls for a program to be dissected into its constituent parts and then the rest of the test demands that the unprotectable elements be filtered out before the process of comparison

---

[121]   Above n 55.

[122]   P Walters "Copyright or Copy Software Law Reform" Australian Communications October 1993, 57.

[123]   Above n 122, 58.

takes place.[124] This process of analytical dissection which appears in *Altai*, as well as in other leading United States cases, was expressly rejected by our Court of Appeal in *Bleiman v News Media (Auckland) Ltd*.[125]

Bleiman involved a sports competition based on the results of sporting events. The plaintiff was seeking an interim injunction to prevent a rival publication from starting a similar game. The issue was whether the new competition substantially copied the features of the existing one.[126]

In addressing this issue the Court of Appeal laid out its approach to deciding whether or not copyright had been breached. Approving the judgment of the House of Lords in *Ladbroke (Football) Ltd v William Hill (Football) Ltd*,[127] the Court of Appeal stated that the first question to be considered was whether the "work as a whole" was original.[128] The court cited with approval a statement from Lord Reid that:[129]

> It may sometimes be a convenient shortcut to ask whether the part taken could by itself be the subject of copyright. But in my view ... the more correct approach is first to determine whether the plaintiff's work as a whole is original and protected by copyright and then to inquire whether the part taken is substantial. A wrong result can easily be reached if one begins by dissecting the plaintiffs work and asking, could section A be the subject of copyright if it stood by itself.

The Court of Appeal then stated that in deciding whether a substantial part of the work had been copied, the test, from *Taypar Pty Ltd v Santic*,[130] was whether the essence of the copyright work had been taken.[131]

In applying this approach to the facts the court rejected the defence's submission that the features allegedly infringed were common to any game with similar subject matter. The court stated that this was the dissectional approach rejected by Lord Reid in *Ladbroke*.[132]

---

124 Above n 52.

125 Above n 7.

126 Above n 7, 674-677.

127 [1964] 1 WLR 273.

128 Above n 7, 278.

129 Above n 7, 278.

130 (1980) 17 IPR 146.

131 Above n 7, 278.

132 Above n 7, 279.

Having decided that copyright could reside in the plaintiff's game as a whole the court concluded that it was arguable that the "whole essence of the game had been taken and retained" by the defendant.[133] The court granted an interim injunction in favour of the plaintiff.

While the court in *Bleiman* was dealing only with an interlocutory injunction it clearly lays out the approach which the Court of Appeal believed should be taken to the issue of copyrightability and substantial similarity. In *Bleiman* the Court of Appeal took the view that the first step is to look at the work as a whole to see whether it is original. This approach is very similar to that taken *Whelan* and *Paperback*. Both those cases looked at whether the program as a whole, rather than its individual elements, could be protected by copyright.

The approached adopted in *Bleiman* is therefore very close to protecting the "look and feel" of a work and, unless the courts wish to distinguish computers from other literary works, then the "look and feel" approach would appear to be the approach that is likely to be adopted.

Further support for this conclusion can be found in the decision in *John Richardson*, a case dealt with in *Ibcos*. While accepting the general approach adopted in *Altai*, Ferris J stated "that in preference to seeking the 'core of protectable expression' in the plaintiff's program an English court will first decide whether the plaintiff's program as a whole is entitled to copyright".[134]

The court in *Ibcos* accepted that statement to be a true account of the current approach to the issue of copyrightability in England.

The process of analytical dissection which is inherent in the *Altai* test for copyrightability and substantial similarity is clearly repugnant to New Zealand law. The court in *Bleiman* has expressly rejected this approach and instead it has promoted the protection of the "look and feel" of a copyrighted work. Just such an approach has been applied in *Ibcos* to the non-literal elements of a computer program. There the court found that the work, as a whole, could be protected by copyright.[135] Clearly, New Zealand law is closer to the approach adopted in *Whelan* and *Paperback*, than it is to the approach in *Altai*.

---

[133]   Above n 7, 278.

[134]   Above n 11, 302.

[135]   Above n 88.

C   Conclusion

An examination of New Zealand law reveals that on the basis of established copyright principles we should extend copyright protection to a program's non-literal elements should be extended. The Copyright Act 1994 sets up a copyright scheme which is not fundamentally different from that in other Commonwealth countries where protection has been extended. The case law reveals a willingness by New Zealand courts to protect copyrighted works from more than just literal copying. The courts are prepared to protect some of the ideas. Moreover, the courts do not distinguish between functional and aesthetic items when asking whether something can be protected by copyright.

In approaching the narrower issue of copyrightability in individual cases New Zealand law appears closer to the early "look and feel" cases in the United States - such as *Whelan* and *Paperback* - rather than the latter cases - such as *Altai* - where the tests involve the analytical dissection of a work. This may put us at odds with the current approach in the United States but with no guidelines from the legislature the courts must apply existing copyright principles. In that regard it would appear that extensive protection would be afforded to a program's non-literal elements.

V   CONCLUSION

That a Mills and Boon novel is protected by copyright as a literary work requires a leap of imagination. However, an even greater leap of imagination is required to think of computer software as a "literary work". Nonetheless, this is the bizarre fiction which we have to deal with under the Copyright Act 1994. What is more bizarre, however, is that the New Zealand legislature, in imposing this fiction, has provided no guidance as to the limits of copyright in the area of computer programs. The issue of limits has simply been left to our courts.

This article has shown that, on the basis of New Zealand law, New Zealand should provide extensive copyright protection to computer programs. However, while there may be no legal grounds to deny copyright protection to a program's non-literal elements, it should be acknowledged that there are policy grounds for doing so. Although New Zealand law does not draw any legal distinction between functional and aesthetic works many commentators argue that on policy grounds you can distinguish computer programs from other literary work.

What lies at the heart of policy concerns is the belief that computer programs are fundamentally different from other functional items protected by copyright as computer programs are not designed simply to convey information, but to cause processes to be

performed.[136]   Therefore, there is the danger that extensive copyright protection for computer programs will provide patent like protection when the program has not actually had to meet the stringent requirements of patent law.[137]

Computer programs are also fundamentally different from other literary works in that standardization and intero-perability are desirable. Literary works are valued for their variety and their uniqueness. However, variety is not a desirable goal of the software industry.[138] The ability to interact and operate with other programs and the ability to meet industry standards is the goal of most computer products. Therefore there is the danger that if strong copyright protection is provided programrs will be forced to make variations that serve no useful purpose, locking them out of the market since consumers have become familiar with the "standard product".[139]

It must also be acknowledged that copyright protection directly impacts on the level of innovation within an industry.[140] "An economic analysis of copyright protection suggests that as copyright protection increases beyond a certain level, the rate of innovation will actually decline - notwithstanding the increased protection given to developers."[141] Therefore, there is the possibility that by providing extensive copyright protection the courts may stunt innovation within New Zealand's developing software industry.

The consideration of these different policy issues has clearly influenced the decisions in the United States. *Altai, Brown Bag, Microsoft* and *Borland* identify a number of these concerns in limiting the protection available to computer programs.[142] On the other hand, the court in *Paperback* argued that concerns over the functional nature of computers, the need for standardisation and the desire that programrs be free to borrow and improve

---

[136]   Above n 4, 433.

[137]   Above n 13, 614.

[138]   Above n 4, 234.

[139]   Above n 4, 437.

[140]   Above n 9, 199.

[141]   Above n 9, 199.

[142]   Above n 1, 817. The court stated "[t]hat the Lotus menu command hierarchy is a method of operation becomes clearer when one considers program compatibility. Under Lotus's theory, if a user uses several different programs, he or she must learn how to perform the same operation in a different way ... We find this absurd".

upon others ideas were not sufficient to override an authors rights.[143] In *Whelan* the court stated that:[144]

> Although the economic implications of this rule are necessarily somewhat speculative, we nevertheless believe that the rule would advance the basic purpose underlying the 'idea/expression' distinction, 'the preservation of the balance between competition and protection reflected in patent and copyright laws'.

Clearly the courts are not capable of resolving these policy issues by themselves. Different courts can reach different conclusions by placing emphasis on different policy concerns. Nevertheless, the courts clearly recognise that serious policy issues are involved.

There is therefore a need for the legislature to return to the question of copyright protection, not simply in regards to computer programs, but in regards to the information age and the challenges it throws up (computer programs being one of only many). How our legislature meets those challenges may effect whether New Zealand has an innovative and creative economy which leads the world or an economy which dwells in the doldrums. However, until the New Zealand legislature does face that challenge software developers in New Zealand can only role their dice and hope that the numbers will fall their way should they need to test the degree of protection which the Copyright Act 1994 affords them.

---

[143]   Above n 6, 57-69.

[144]   Above n 5, 1237.