

# INTEROPERABILITY IN THE PACIFIC RIM: RECENT DECISIONS IN SINGAPORE AND AUSTRALIA

by Jonathan Band and Noah Levine\*

## I. INTRODUCTION

Recent copyright decisions in United States courts have evinced a trend towards narrowing the scope of protection in computer software in order to facilitate interoperability. In *Bateman v. Mnemonics, Inc.*, 79 F.3d 1532 (11th Cir. 1996), *Lotus Development Corp. v. Borland International, Inc.*, 49 F.3d 807 (1st Cir. 1995), *aff'd by equally divided Court*, 116 S. Ct. 804 (1996), and *Compaq Computer Corp. v. Procom Technology, Inc.*, 908 F. Supp. 1409 (S.D. Tex. 1995), U.S. courts have found that external interoperability concerns *could* operate to deny copyright protection to certain aspects of computer software. These decisions helped to alleviate the concerns of many developers regarding their exposure to copyright infringement liability for their interoperable products.

The other primary concern of developers of interoperable software is that of potential infringement in the *development* stages of that software. An understanding of the manner in which other programs operate, which is necessary in order to develop interoperable software, can sometimes be gained only through the decompilation of that program's machine readable object code into a higher level, human readable form. Some dominant software firms, however, have argued that the translation inherent in the act of decompilation infringes their exclusive rights. The Ninth Circuit eliminated many of these concerns in *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), which held that

---

\* Mr. Band is a partner in the Washington, D.C. office of Morrison & Foerster. Mr. Levine is a summer associate in the Washington, D.C. and San Francisco offices of Morrison & Foerster, and a third-year student at the Columbia University School of Law.

decompilation in order to gain access to the unprotectable elements of the program, when no other means of access is available, is a fair use.

During the past year, two copyright decisions with special ramifications for both of these interoperability concerns have been handed down by courts in Commonwealth countries in the Pacific Rim. The most recent of the two, and perhaps the most beneficial for international developers of interoperable software, was decided by the High Court of the Republic of Singapore in a case concerning a more limited form of reverse engineering than the decompilation at issue in the *Sega* decision. The second, and the more harmful for software developers concerned with interoperability, was decided earlier by a federal court in Australia in a case involving the protectability of functional components of a computer program dictated by external compatibility concerns. In Singapore, the cause of interoperability was advanced by a decision approving of limited forms of reverse engineering. In Australia, the cause was reversed, with the court deciding that compatibility concerns could not negate the protectability of program elements.

What makes these decisions particularly interesting is their use and interpretation of United States copyright law. While the Singapore court largely followed the copyright decisions of U.S. courts, the Australian court was much more particular about which U.S. decisions it agreed with, and which it did not. The different approaches by the two courts naturally resulted in the decisions of contrasting benefit for developers of computer software.

**II. AZTECH SYSTEMS PTE LTD. V. CREATIVE TECHNOLOGY LTD.  
No. 93-688 (High Court of the Republic of Sing. Oct. 24, 1995)**

**A. Facts**

The important part of this case, as far as interoperability is concerned, centered on the copying of software into random access memory (RAM) by the defendant. Creative Technology developed the “Sound Blaster” sound cards for use with personal computers. Along with their sound cards, the company packaged some computer software, including a specific program known as TEST-SBC. Aztech, somewhat later in time, began its own project of developing a sound card which would interoperate with applications designed for use with either the Sound Blaster or other standard sound cards in the industry. As part of the process of developing its own sound cards, Aztech copied Creative Technology’s TEST-SBC program into the RAM of its own computers. This was done to allow the Aztech research and development team to run the computer program and test it by running it along with other programs. Through this process, Aztech’s researchers hoped to study the manner in which the Creative Technology program communicated directions to the Sound Blaster sound card.

Thus, the reverse engineering was limited only to running the actual program in tests, *i.e.*, “black box reverse engineering,” not the decompilation of Creative Technology’s program from object code into a higher level form. (Black box reverse analysis does not involve the translation of the object code of the program under study, but may involve “techniques such as test runs, communication line traces, storage media dumps, and screen displays of hexadecimal object code.” Thomas C. Vinje, *The Development of Interoperable Products Under the EC Software Directive*, Computer Law., Nov. 1991, at 3.) Therefore,

the legal issue presented was whether Aztech’s copying of the computer program into memory, in order to run the program as part of its hardware development process, qualified as “fair dealing”—the Commonwealth equivalent to the United States’ fair use doctrine. The court evidently assumed that a transitory RAM copy was a copy within the meaning of Singapore’s copyright law.

## **B. The High Court’s Ruling**

### **1. Fair Dealing**

The four enumerated factors applied in determining whether a use should be considered “fair dealing” under the Singapore Copyright Act are remarkably similar to those under the U.S. Act. However, Singapore’s statute is more restrictive in that it also requires that the use of a literary work be “for the purpose of research or private study.” *Aztech* at 54. The statute also explicitly excludes from its definition of “research” industrial research or research carried out by companies or other business groups. *See id.* However, even under this restrictive definition of fair dealing, the Singapore High Court found the use by Aztech to constitute “private study” within the intended meaning of the statute. The judge stated: “It seems to me that a study is *private* if the study and the information and knowledge acquired through it are kept or removed from public knowledge or observation and this is so even if the purpose may be of a commercial nature.” *Id.* at 56–57. This aspect of the decision is important since a failure to construe “private study” to reach commercially-motivated study would necessarily have excluded all forms of reverse analysis of computer software from the fair dealing defense under Singapore law. It is also notable in that it shows the court’s stretching of the traditionally restrictive doctrine to accommodate the analysis practiced by Aztech.

In its evaluation of the four enumerated fair dealing factors, the court stressed the fact that Aztech copied the Creative Technology program into the memory of its own computers only in order to run the program. The judge wrote that the program

was copied to the memory of the computer. That is the ordinary way of running the program and it was undoubtedly the way Creative intended it to be run. It was not copied in any other way nor in any other form. It was not disassembled or printed out. No copies of it in any form were made for distribution or which could be distributed. I think this weighs in favour of fairness.

*Id.* at 57. This was important to the first fair dealing factor—the purpose and character of the dealing—because it indicated that the *copying* of the computer program was for the intended purpose of running the program. It was also important to the court’s determinations on the second and third factors—the nature of the work and the amount and substantiality of the part copied—since the nature of a computer program requires that it be copied in its entirety into the computer’s memory in order for the program to be run efficiently.

A significant distinction emphasized by the court several times during its fair dealing determination was that Aztech was attempting to develop a competing sound card, not a software program that would compete with Creative Technology’s TEST-SBC program. “I think it has to be borne in mind though that the product that was developed and marketed by Aztech was the *sound card*. It was not a *software program* that emulated TEST-SBC or the instructions in it.” *Id.* at 58. The court found this to be important in evaluating not only the commercial purpose of the dealing, but also the effect upon the potential market or value of the work, since the development of a competing hardware product would not compete directly, or perhaps at all, with the analyzed software.

Finally, the court also considered, as an independent and additional factor in its fair dealing determination, the public interests advanced by the dealing at issue here. Like the Ninth Circuit in *Sega*, 977 F.2d at 1523, 1527, the Singapore High Court found that the increased competition in the market “to more than just Creative and those licensed by it” would be a benefit “in consonance with the purpose of the Act.” *Aztech* at 59–60. The court then concluded that the balance of the considerations required a finding that Aztech’s use of Creative Technology’s program constituted fair dealing under the Act and therefore succeeded as a defense to infringement.

## **2. Two Important Caveats**

The limited extent of reverse engineering reviewed in this case must be noted. All the High Court considered was Aztech’s copying of the TEST-SBC program into RAM in order to study a specific command of the program. The court specifically observed that Aztech did not decompile Creative Technology’s software. This observation creates the implication that the High Court may have ruled differently in a case involving decompilation of a computer program’s object code.

A second potential limitation suggested by the High Court’s opinion is that reverse analysis of computer programs for the purpose of developing competing *software* may not be construed as fair dealing. Here, Aztech studied the Creative Technology program in order to develop competing *hardware*. As stated above, this was a crucial factual distinction in the court’s evaluation of two of the four statutory fair dealing factors. The presence of direct competition may have altered the court’s ultimate finding of fair dealing.

### 3. Use of Foreign Law

It is interesting that the High Court considered U.S. copyright decisions regarding reverse engineering specifically, and fair use generally, and yet left open the possibility of the limitations discussed above. The court first justified its examination of U.S. decisions because the provisions for fair dealing in Singapore and for fair use in the United States “are in many respects similar.” *Aztech* at 61. The U.S. decision elaborated upon most heavily by the court was the Ninth Circuit’s *Sega* decision. However, neither of the limitations implicitly imposed by the High Court—that the decompilation of object code may not be a fair use/dealing, or that in order to be fair, subsequent competition with the analyzed software may have to be indirect—were present in the *Sega* decision. The Singapore High Court instead turned to the *Sega* decision rather warily, embracing only its recognition of the public interest as a relevant factor in fair use determinations:

While I am conscious of the need to approach the American authorities with caution in view of the development in that jurisdiction of the distinction between *ideas and functional concepts* which are not protected by copyright and the *expression* of those unprotected elements which is protected it is at least comforting to know that the broader public interest is a factor to be taken into consideration albeit in respect of the “purpose and character” of the use although I would prefer to treat it as a separate matter to which regard is to be had.

*Aztech* at 63.

Of course, perhaps the potential limitations in *Aztech* are just that—potential. The judgment of the court was largely determined by the facts of the case as presented. In contrast to the *Sega* case, the facts in *Aztech* presented neither the issue of reverse engineering by decompilation nor that of undertaking such study in order to develop a product that would compete directly with the software analyzed. Moreover, to the extent that the *Sega* decision recognized that copying for the purposes of reverse engineering was in

some form acceptable, the *Aztech* decision followed it substantially. In this respect, international developers of interoperable software have reason to celebrate the High Court's decision.

The court also investigated the applicability of an 1871 British patent case, *Betts v. Wilmont*. There the British court held that a purchaser of goods patented by another party has control of the goods such that, absent a clear and explicit agreement to the contrary, the purchaser may sell or use the articles however he pleases. *Aztech* argued that under *Betts v. Wilmont*, its purchase of the software from Creative Technology gave it the "right to use it for a reasonable purpose." *Aztech* at 68. The court agreed, finding that

When a man buys a Sound Blaster sound card and with it comes TEST-SBC he expects to have and to exercise his rights of ownership over it. He can use it. That is one of the rights of ownership. He can run the software in his PC. That is what it is for. To run it the program has to be copied to the PC's memory. He can run the program in as many PCs and as often as he pleases. He can study it to see what it is doing and he can experiment with it. That is exercising his right as an owner to use it. But he cannot make and distribute copies of it. That is not using it. That is not using the software he has bought.

*Id.* at 75. Most importantly, the court located the above-described right in *Aztech*'s ownership of the software, not in the reasonableness of the use for which the software was employed. *See id.* at 76. Thus, even though the court suggested potential limitations on the fair dealing defense, it provided a fairly broad and independent defense based on the right of ownership as recognized in *Betts v. Wilmont*.

### **III. DATA ACCESS CORPORATION V. POWERFLEX SERVICES PTY, LTD.**

**No. 93-VG473 (Federal Ct. Austl. Feb. 9, 1996).**

#### **A. Facts**

This Australian case focused on infringement by the end product, not infringement due to copying during the development process. The Data Access Corporation developed

DataFlex, a compilation of programs described by the Australian federal court “as an application development system.” *Data Access* at 2. Specifically, the DataFlex programs are used for not only the creation of databases, but also for the development of database application programs which can in turn be used to create and work with databases. As such, the DataFlex system provides users with a programming language in which they can write their own programs. The DataFlex system also incorporates a “run time” program which allows one to operate the applications created through the use of the DataFlex language. The defendants in the copyright infringement suit were Dr. David Bennett, his wife, and the company they incorporated to sell the allegedly infringing product, PFXplus. Bennett, according to the court, “aspired to create an application development system which would be highly compatible with the DataFlex computer language.” *Data Access* at 3.

As far as issues of interoperability were concerned, the infringement suit focused on two aspects of Bennett’s software. First, the PFXplus language used 192 of the 225 instruction words in the DataFlex language. The use of the same words in the source code of either the DataFlex or PFXplus languages caused the computer to perform the same functions. The program code implementing each of these functions, however, was completely different.

The second interoperability issue arose in the context of Bennett’s intentional reproduction, in identical form, of certain compression tables used in one of the DataFlex programs. These tables, which were part of the DataFlex “run time” program, merely allowed the software user to save storage space through the compression of the program into smaller data strings. This raised an interoperability issue because of Bennett’s assertion in

his defense that it was necessary to use the same compression tables in his programs in order to achieve compatibility.

## **B. The Federal Court's Ruling**

### **1. Merger of Idea and Expression**

Bennett presented a merger defense to the copying of the words of the DataFlex language. His merger defense was based on that language in *Baker v. Selden*, 101 U.S. 99 (1879), as affirmed in the Australian case, *Autodesk v. Dyason*, 173 C.L.R. 330 (Austl. 1992), holding that when an expression of an idea is inseparable from its function, it is part of the idea itself and therefore unprotected by copyright. Specifically, Bennett appealed to the logic of the First Circuit's *Borland* decision, although not specifically to the "method of operation" argument, since the Australian copyright statute has no companion to section 102 (b) of the U.S. Act. Although not argued specifically by the parties, it appears that as in *Borland*, it was necessary for Bennett to use the same words in order to make PFXplus compatible with DataFlex and desirable to customers who had learned the DataFlex language and did not want to learn a new set of commands. *Cf. Borland*, 49 F.3d at 810.

The Australian court chose to disregard the ruling in *Borland* and instead turned to Judge Keeton's district court decision in *Lotus Development Corp. v. Paperback Software International*, 740 F. Supp. 37 (D. Mass. 1990), which as a practical matter had been overruled by the First Circuit *Borland* decision. The Australian court quoted extensively from the *Paperback* decision, and seemed to place great importance on Judge Keeton's decision that if the expression of an idea goes beyond the functional elements within that idea and beyond the obvious, *and* if there are numerous other ways of expressing the same idea, that form of expression is copyrightable. *See Data Access* at 12–13. The Australian

judge found that the choice of words here went “beyond the functional elements of the ideas they express, and beyond the obvious, and are . . . elements of expression, original and substantial, and therefore copyrightable.” *Id.* at 13. Thus, the Australian court rejected *Borland’s* notion that elements of a computer program so “essential to operating something” should be outside the scope of copyright protection. *Borland*, 49 F.3d at 816.

## 2. Compatibility as an External Factor

The Australian court more starkly demonstrated its position on what impact compatibility concerns could have on the protectability of program elements in its findings regarding Bennett’s copying of the compression tables. Bennett once again presented a merger defense, arguing that in order for a person using the DataFlex application development system to use a PFXplus program, the PFXplus compression table must be identical to that of DataFlex. He further contended that this meant there was but one manner in which to express the idea of “function through compatibility,” and therefore the idea and expression merge, making the compression table unprotectable under copyright. *Data Access* at 17.

This argument was rejected by the court, with reference once again to Judge Keeton’s *Paperback* opinion. The Australian court held that compatibility concerns *could not* negate the protectability of a program element:

The function of compression by means of the Hoffman method may be, and has been, performed by any one of very many different expressions in integer code. The expression given in the DataFlex table is but one of the many possible expressions. The conclusion is established that the DataFlex table is copyright [sic] before consideration is given to the PFXplus table. *The desire of Dr. Bennett for the compatibility he achieved by reproduction of the DataFlex table, not any inseparability of function and expression of the Hoffman compression method, constrained him to merge function and expression.*

*Data Access* at 17 (emphasis added).

Viewed in this light, the Australian decision is very near the opposite side of the spectrum—in terms of protection offered to interface specifications—from recent cases in the United States. Perhaps the clearest contrast can be discerned when comparing the *Data Access* decision to the Eleventh Circuit’s ruling in *Bateman v. Mnemonics, Inc.* 79 F.3d 1532 (11th Cir. 1996). In the latter case, the court found that, although interface specifications were not *per se* uncopyrightable, external factors dictated by compatibility concerns would typically negate the protectability of certain elements of computer programs.

By distinguishing constraints dictated by compatibility from constraints flowing from the inseparability of function and expression, the Australian court basically decided on a *per se* basis that no elements of computer programs dictated by external compatibility concerns could be determined to be unprotected. In light of the state of U.S. law as represented by *Borland* and *Bateman*, the Australian court’s reliance on the opinion of Judge Keeton in *Paperback* as an example of U.S. legal authority seems even more puzzling. Perhaps more mystifying is the Australian court’s failure to consider an even more obvious U.S. case relevant to the questions before the court—*Computer Associates International, Inc. v. Altai, Inc.*, 982 F.2d 693 (2d Cir. 1992), which established the abstraction-filtration-comparison test for questions of infringement by nonliteral copying of computer programs.

Interestingly, the court made a completely contrary finding with respect to another portion of the facts presented, thereby casting doubt on the above-described finding. Bennett also copied a substantial portion of the error text table contained in the DataFlex programs. When an error occurred during use of the DataFlex software, the program referred to an error table which contained numbered errors corresponding to specific lines of text to be displayed

on the computer screen, describing the error to the user. Bennett's error text lines were substantially similar to those provided in the DataFlex table. However, in these circumstances, the court found that the "expression of the idea is inseparable from its function and is not copyrightable." *Data Access* at 15. Although the facts are not elaborated upon in greater particularity than presented here, it is at least questionable why the court would so easily apply the merger doctrine in this circumstance, but not do so with respect to the words of the DataFlex language. However mixed the message of the court may be, it certainly cannot be interpreted as favorable to developers of interoperable software.

#### **IV. CONCLUSION**

These two cases show that the courts of Pacific Rim countries are moving in different directions when considering interoperability as a factor in their determinations of the scope of copyright protection in computer software. Although U.S. copyright law seemed to gain some form of extraterritorial application in both cases, the manner in which it was interpreted led to drastically different results. In Singapore, the U.S. law as applied in *Sega* received cautious support, thereby bestowing upon developers of interoperable software a fair dealing defense to claims of infringement arising from copies made during a reverse analysis process. However, in the Australian decision, U.S. cases such as *Borland* were not adopted, thereby indicating that interoperability concerns may not assume as significant a role in Australian decisions pertaining to the scope of copyright protection for computer software.

A final observation: Although both courts used (and in *Data Access*, arguably misused) U.S. authorities, neither referred to any of the four recent software copyright decisions in other Commonwealth countries: two in Canada and two in the United

Kingdom. See Jonathan Band, *Computer Associates Crosses the Atlantic and Lake Ontario: Richardson v. Flanders and Delrina v. Triolet*, 1 Int'l Computer Law. 2 (June 1993); *Matrox Electronic Systems v. Gaudreau*, 6 Eur. Intell. Prop. Rev. D-138 (1994). It seems that now the sun never sets on the American Empire.

DC39106