

# **INTEROPERABILITY IN THE PACIFIC RIM: REVERSAL OF FORTUNES IN SINGAPORE AND AUSTRALIA**

**Jonathan Band and Taro Isshiki, Morrison & Foerster LLP<sup>1</sup>**

## **INTRODUCTION**

In the fall of 1995, the Singapore High Court in *Aztech Systems Pte. Ltd. v. Creative Technology Ltd.*, No. 93-688 (High Court of the Rep. of Singapore Oct. 24, 1995) (as reprinted in Fleet Street Reports (1996)), found “fair dealing” in the copying of software during the course of reverse engineering in order to achieve compatibility. In early 1996, the trial judge in the Australian Federal Court in *Data Access Corp. v. Powerflex Services Pty. Ltd.*, No. 93-VG473 (Federal Ct. Austl. Feb. 9, 1996), held that command names were copyrightable as computer programs and that compatibility concerns would not negate the protectability of program elements. For the developers of interoperable software, *Aztech* was beneficial while *Data Access* represented a major set-back.<sup>2</sup>

Both decisions have now been reversed on appeal. In Singapore, the appellate court disagreed with almost all of the trial court’s rulings and held that fair dealing under Singapore law applied only to non-commercial research. In Australia, the Full Court held that command names were uncopyrightable. This article discusses these appellate decisions and their implications for the development of interoperable computer software.

***CREATIVE TECHNOLOGY LTD. v. AZTECH SYSTEMS PTE LTD.*  
No. 95-181 (Court of Appeal of the Republic of Sing. Nov. 12 1996)**

## **Facts and The High Court’s Ruling**

The interoperability aspect of this case centered on the defendant’s copying of software into random access memory (RAM) during the course of reverse engineering. Creative Technology developed the “Sound Blaster” sound cards for use with personal computers. Creative packaged some computer software, including a specific application program known as TEST-SBC, with its sound cards. Aztech began developing a sound card which would interoperate with applications designed for use with a variety of different sound cards, including the Sound Blaster sound card. As part of the development process, Aztech copied Creative’s TEST-SBC program into the RAM of its computers to study the manner in which Test-SBC communicated instructions to the Sound Blaster sound card.

---

<sup>1</sup> Jonathan Band is a partner in the Washington, D.C. office of Morrison & Foerster LLP. Taro Isshiki is a third-year law student at the National Law Center at George Washington University.

<sup>2</sup> See Jonathan Band and Noah Levine, Conflict in Commonwealth Courts: A Review of Recent Software Interoperability Decisions in Singapore and Australia, *Journal of Proprietary Rights* (July 1996), for a more detailed discussion of these cases.

Aztech claimed to have made only RAM copies in test runs, and not to have engaged in the decompilation of Creative's program from object code into a higher level. The issue before the court was whether Aztech's copying of the Creative's computer program into RAM as part of its hardware development process qualified as "fair dealing" under the Singapore Copyright Act (SCA).

The four enumerated factors in Singapore's fair dealing provision are essentially identical to those under the fair use section of the U.S. Copyright Act, 17 U.S.C. § 107. In addition, the Singapore's fair dealing section provides a safe haven for certain kinds of research and private study under Section 35(1): "A fair dealing ... for the purpose of research or private study shall not constitute an infringement of copyright." However, the Singapore's statute has a restrictive clause which, on its face, excludes any research carried out by business entity. Section 35(5) defines the scope of "research," as excluding "commercial research, research carried out by bodies corporate ... or bodies of persons carrying on a business."

Despite the explicit exclusion of commercial research, the Singapore High Court found that the use by Aztech nevertheless constituted "private study" within the intended meaning of the statute. In the court's opinion, 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." *Aztech* at 62. Upon finding that Section 35 of SCA would not necessarily prohibit commercially-motivated study, the High Court weighed the four enumerated "fair dealing" factors set forth in Section 35 as well as the public interests advanced by Aztech's acts. The court concluded that the balance of these considerations required a finding that Aztech's use of Creative's program constituted fair dealing under the SCA.

### **Court of Appeal's Ruling**

#### ***"Fair Dealing" Defense***

The Court of Appeal disagreed with the trial judge and decided that Section 35(1) excludes commercial research as well as private study for commercial purposes. The court said that in order to come within the "private study" exception, the copying must be undertaken by the student himself. *Creative* at 33. The court was influenced by the argument that if it were to adopt a broader construction of "private study" to extend to "private study for commercial purposes," it effectively would render meaningless the specific exclusion of commercial research under Section 35(5). Since the court concluded that Aztech's admitted copying of TEST-SBC did not qualify as "research or private study," the fair dealing defense was unavailable to Aztech.

#### ***"Essential Step" Defense***

Section 39(3) of SCA, which is derived from Section 117 of the U.S. Copyright Act, allows the owner of a computer program to copy or adapt that computer program as an essential step in the utilization of the program in conjunction with a machine. The appellate court considered whether the essential steps in using a computer program could include copying it into the RAM for the purpose of studying the underlying ideas and concepts of the program. Relying on interpretations of Section 117, including *Apple Computer, Inc. v. Formula*

*International, Inc.*, 594 F. Supp. 617 (C.D. Cal. 1984), *aff'd*, 725 F.2d 521 (9th Cir. 1984), and *Allen-Myland, Inc. v. IBM Corp.*, 746 F. Supp. 520 (E.D. Pa. 1990), *vacated and remanded*, 33 F.3d 194 (3d Cir. 1994), *cert. denied*, 513 U.S. 1066 (1994), for guidance, the court agreed with Creative that Section 39(3) was enacted for the limited purpose of allowing the rightful owner of the program to load and use it in his computer. *Creative* at 44. In the court's view, Section 39(3) did not allow copying or adaptation for the creation of a compatible product. Therefore, Aztech's RAM copy of TEST-SBC could not be deemed an essential step in the utilization of the program. In reaching this conclusion, the court explicitly rejected the Fifth Circuit's decision in *Vault Corp. v. Quaid Software Ltd.*, 847 F.2d 255 (5th Cir. 1988), which held that Section 117 permitted RAM copying during the course of reverse engineering.

### ***Implied License***

Aztech argued that when it purchased TEST-SBC, it obtained along with its physical ownership the right to use it for a reasonable purpose. Aztech relied on a 1871 British patent case, *Betts v. Wilmont*, which held that the purchaser of a patented article has an implied license to sell the article and to use it for any reasonable purpose, absent some clear and explicit agreement to the contrary. *Creative* at 35-36.

Aztech argued that its use was for a reasonable purpose; the copy of TEST-SBC was made to ascertain functionality with the object of building a non-infringing compatible product. The trial judge made no finding on whether such use was indeed reasonable, but was persuaded that Aztech merely exercised an inherent right of ownership conferred by the purchase of TEST-SBC.

The Court of Appeal disagreed with the trial judge and held that the proposition in *Betts* is inapplicable in the Singapore copyright context. The court noted that the exclusive rights granted to the patent owner differ materially from those accorded to the copyright owner and that to uphold such an implied license would run contrary to the provisions of the SCA.

### **Disassembly of Firmware**

At trial and on appeal, Creative alleged that Aztech had disassembled a substantial portion of the firmware embedded in the Sound Blaster microprocessor. The trial court decided as a factual matter that Aztech did not disassemble the firmware.

The Court of Appeal disagreed. After an extensive review of the facts, the court determined that Aztech had the means, motive and opportunity to disassemble Creative's firmware, and the literal similarities between Aztech's and Creative's firmware raised the "irresistible inference that the chances of independent development on the part of Aztech were low." *Creative* at 24. However, because only 4% or less of Aztech code was identical to Creative's, the court held that Aztech's copying did not amount to a substantial taking.

Although the appellate court found no liability for the disassembly, it nonetheless seemed disturbed by the act of disassembly itself: "This, however, in no way prejudices our finding of disassembly, which involves a degree of reproduction and adaptation having a greater impact in terms of revealing the ideas and interfaces of a copyright holder's program, insights

which would not otherwise have been obtained by independent development or empirical observation within a given time frame.” *Creative* at 25. In other words, the court viewed the disassembly negatively precisely because it revealed features not protected by copyright.

The court’s hostile view toward disassembly provides a clear contrast to the Ninth Circuit’s decision in *Sega Enterprise Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992). In *Sega*, the court decided that disassembly in order to gain access to the unprotectable elements of the program, when no other means of access is available, is a fair use. The *Sega* court did not view decompilation as an independent wrong, but as a legitimate means to study the ideas of the computer program in order to create a non-infringing product. (Interestingly, the Court of Appeal followed *Sega*’s holding that Section 117 did not permit decompilation. *Creative* at 45.)

After the *Creative* decision, no commercially-motivated research or study is entitled to the fair dealing defense in Singapore. Thus, interoperable developers cannot use the fair dealing defense to excuse the interim copies made during disassembly and other forms of reverse engineering. Additionally, the developers of interoperable software cannot rely on “essential step” or “implied license” arguments to justify their interim copying.

***POWERFLEX SERVICES PTY. LTD. v. DATA ACCESS CORPORATION*  
No. 96-VG295 (Full Court of the Federal Ct. Austl. Jun. 4, 1997)**

**Facts and Federal Court’s Ruling**

This Australian case focused on infringement by the end product, not infringement during the development process. Data Access Corporation developed Dataflex, an application development system which allows the user to program customized database applications. Along with the development component, Dataflex includes a runtime program, which permits the user-written database applications to run on a computer.

The defendant, Dr. David Bennett, created a competing application development system, called PFXplus, using the same commands, file structure and function keys as Dataflex. This allowed Dataflex users to switch to PFXplus without having to learn a new language and to run their Dataflex applications on PFXplus.

The undisputed facts indicated that in creating PFXplus, Bennett studied both the documentation and operation of the Dataflex program. However, Dataflex’s source code was quite different from the PFXplus source code, and Data Access did not allege any similarity in the object code.

The portion of Data Access’ infringement suit related to interoperability focused on two aspects of Bennett’s software: the commands and the Huffman compression tables. The trial court held that each of the commands in the Dataflex language was an expression of a set instructions falling within the definition of “computer program” in the Copyright Act. Bennett used 192 of the 254 commands in Dataflex lexicon, to perform the same function as in Dataflex. Many of the commands, such as “save” and “display,” performed similar functions in other programs, while at least 55 words, *e.g.* “end\$of\$report” and “keyproc,” were unique to Dataflex.

Bennett presented a merger defense to the copying of the words of the Dataflex language. He based this defense on *Baker v. Selden*, 101 U.S. 99 (1879), followed in the Australian case *Autodesk v. Dyason*, 173 C.L.R. 330 (Austl. 1992). Bennett argued that it was necessary 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. The trial court rejected Bennett's merger argument largely because the Australian copyright statute has no section comparable to § 102(b) of the U.S. Act. The court ruled that the command names are copyrightable in Australia so long as there are "numerous other ways of expressing the non-copyrightable idea." *Data Access* at 12. The court cited Judge Keeton's decision in *Lotus Development Corp. v. Paperback Software International*, 740 F. Supp. 37 (D. Mass. 1990), as support for its conclusion. See *Data Access* at 12-13.

The trial judge also found copyright infringement in Bennett's intentional reproduction of the Huffman compression table used in Dataflex's runtime program. Huffman compression is a method of storing data in a smaller amount of space. The compression table in essence allows the program to save space by employing shorthand for frequently used commands. The PFXplus compression table had to be identical to the Dataflex compression table in order to run an application written on the Dataflex system; a different table would misinterpret the shorthand in the compressed Dataflex program. Because there is only one table that will properly match the commands with the shorthand, Bennett argued, the expression merged with the idea, and could not receive copyright protection.

This argument was rejected by the court, with reference once again to Judge Keeton's *Paperback* opinion. *Data Access* at 17. The court noted that Bennett could have achieved compression—indeed Huffman compression—several different ways. The court was not persuaded by the fact that compression achieved these other ways might render the PFXplus runtime program incompatible with the Dataflex applications.

## **Full Court's Ruling**

### ***Command Names***

In an unanimous decision, the Full Court overturned the ruling that copyright subsisted in the names of a computer program's commands. The Full Court rejected the trial court's approach of characterizing commands as elements of copyrightable expression. Noting that the statute defines "computer program" as an expression of a *set* of instructions, the Full Court held that the words used to label the commands were not themselves computer programs. *Powerflex* at 16.<sup>3</sup> The court stated,

Each of the words in the so-called Dataflex language is but a cipher. The underlying program is the set of instructions which directs the computer what to do when the cipher is in fact used, for example by being typed on to the screen. . . . The cipher or command is not an expression of the set of instructions, although it appears in that set of

---

<sup>3</sup> *Powerflex Services Pty. Ltd. v. Data Access Corp.*, No. 96-VG295 (Federal Ct. Austl. June 4, 1997), as taken off the Internet at [http://www.austlii.edu.au/au/cases/cth/federal\\_ct/1997/490.html](http://www.austlii.edu.au/au/cases/cth/federal_ct/1997/490.html).

instructions. It is the trigger for the set of instructions to be given effect to by the computer.

*Id.*

The court was of the view that while both source code and object code were “computer programs,” an individual command was not the proper object of protection because it was not the expression of the underlying set of instructions. The Full Court cited *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), as support for its decision. *Borland* held that by reason of § 102 (b), copyright did not exist in the set of commands used in a program. The Australian Full Court adopted the rationale of *Borland* by stating that while Australian law does not provide a comparable section to the U.S. Copyright Act’s § 102(b), the idea/expression distinction is inherent in the law of Australia. *Powerflex* at 17-18.

The Full Court also rejected Data Access’ alternative argument that, even if the individual commands were not the subject of copyright, the entire set was a protectable compilation. The court gave three reasons. First, some of the Dataflex commands are ordinary English words, and this is not a case where disconnected words are used in a particular order so that the order becomes the linchpin for copyright. Data Access was seeking copyright in words themselves. Second, Data Access conceded that some of the commands are common to all software languages and did not claim copyright in those commands. Third, the court was of the view that, irrespective of their significance, 254 words alone could not be a substantial part of the Dataflex program. *See Powerflex* at 18-19.

### ***The Huffman Table***

With regard to Bennett’s intentional reproduction of the Huffman compression table, the Full Court agreed with the trial court’s result, but not necessarily its reasoning. The Full Court held that the Huffman table was copyrightable as a compilation and that Bennett unlawfully reproduced Dataflex’s table. While acknowledging that the compression table comprises only about 0.25% of the PFXplus’ total source code, the court found infringement because Bennett reproduced the entire table. *Powerflex* at 22.

Unlike the trial court, the Full Court did not squarely address the compatibility issue. This may be because the Full Court found that Bennett could have derived virtually the same Huffman table *independently* without copying the table in the Dataflex program. Huffman tables are created by performing a statistical analysis of the programs one wants to compress to determine the frequency with which each command is used. The court concluded that Bennett could have performed his own statistical analysis and thereby created his own Huffman table which would have been close enough, if not identical, to the Dataflex table. From the court’s discussion of this issue, one can infer that the court would have excused Bennett’s copying of the Dataflex table under a merger or fair dealing theory, had he not been able to derive it independently. Accordingly, the court’s ruling on the Huffman table may not be as hostile to interoperability as it appears at first blush.

## CONCLUSION

These cases reflect foreign courts' efforts to reconcile their statutes with U.S. software copyright decisions. In Singapore, the appellate court was aware of the *Sega* decision, but unlike the trial court, concluded that the specific wording of the Singapore fair dealing statute prevented it from following *Sega*. It relied on U.S. precedent, however, to interpret the Section 117 equivalent as not applying to RAM copying during the course of decompilation.

In Australia, the appellate court found that the idea/expression dichotomy embodied in 17 U.S.C. § 102(b) was inherent in Australian copyright law notwithstanding the absence of a statutory equivalent. This allowed it to issue a decision consistent with *Lotus v. Borland*. Although the Huffman table ruling appears to have been influenced by the fact that the defendant could have derived the table independently, a U.S. court following *Computer Associates International, Inc. v. Altai Inc.*, 982 F.2d 693 (2d Cir. 1992) probably would have found the table unprotected because it was dictated by an external factor—the compatibility requirements of programs written in the Dataflex language. In other words, while software copyright law in Australia is headed in the same direction as U.S. software copyright law, it has not yet traveled as far. In Singapore, by contrast, the law is headed in the opposite direction, unless the legislature intervenes and amends the SCA.