

## **INTEROPERABILITY AFTER *LOTUS V. BORLAND*: THE BALL IS IN THE LOWER COURTS AGAIN**

By Jonathan Band  
*Morrison & Foerster LLP, Washington, D.C.*

On January 16, 1996, the U.S. Supreme Court affirmed the First Circuit's decision in *Lotus v. Borland* by a two sentence *per curiam* order of an equally divided Court. Many had expected the Supreme Court in its first software copyright case to issue a clear pronouncement resolving once and for all the debate on the appropriate scope of protection for computer programs. The *per curiam* affirmance, however, left the *Lotus* decision as binding authority only in the First Circuit.

One of the issues in the *Lotus* case was the protectability under copyright of elements necessary to achieve interoperability. Judge Keeton of the U.S. District Court for Massachusetts had ruled that such elements did receive protection. The First Circuit, however, reversed Judge Keeton on this point and withheld protection from the key reader file which allowed Lotus 1-2-3 compatible macros to run on Borland's spreadsheet program. Although Lotus did not specifically appeal the First Circuit's ruling on the key reader file, Borland and its *amici* stressed to the High Court that interoperability was at the core of the case.

Had the Court settled the matter, then two lower court decisions concerning interoperability issued in December, 1995, one by the Eleventh Circuit and the other by the U.S. District Court for the Southern District of Texas, would have been at most footnotes in the history of software copyright law. But because the Supreme Court in essence punted, these two decisions take on much greater significance. They continue the lower courts' current pro-interoperability tilt, but do so in a way that does not endanger copyright protection for those program elements that deserve it under law.

## **I. BATEMAN V. MNEMONICS**

### **A. Facts and Issues On Appeal**

Bateman developed a single board computer with an operating system for use in automated parking systems. A subsidiary of Mnemonics purchased the Bateman computers and developed compatible application programs. After doing business together for several years, Mnemonics began experiencing difficulties with the Bateman computers. When Bateman could not correct the problem, Mnemonics disassembled<sup>1</sup> the Bateman operating system to discern the interfaces necessary for compatibility with the existing application programs, and then developed its own compatible operating system. Bateman filed suit for, *inter alia*, copyright infringement of its operating system. After trial, a jury found for Bateman.

The appeal centered on the jury instructions. Mnemonics claimed that the district court committed reversible error by (1) instructing the jury to filter-out only nonliteral similarities when applying the Second Circuit’s “abstraction-filtration-comparison” test; and (2) failing to instruct the jury on the legal consequences of copying elements dictated by compatibility requirements.

### **B. The Eleventh Circuit’s Ruling**

#### ***1. Two Interesting Footnotes***

Before reaching the issues on appeal, the Eleventh Circuit in a footnote discussed the lawfulness of disassembly. Although disassembly was not before the Eleventh Circuit, and indeed it appears that disassembly had not been an issue at trial, the Eleventh Circuit elected to provide the district court guidance because the lawfulness of disassembly “likely will appear on remand.” Slip Op. at 12, n.18. After briefly discussing the Ninth Circuit’s decision in *Sega v. Accolade*, 977 F.2d 1510 (9th Cir. 1992) and the Federal Circuit’s decision in *Atari v. Nintendo*, 975 F.2d 832 (Fed. Cir. 1992), the Eleventh Circuit stated that “[w]e find the *Sega* opinion persuasive in view of the principal purpose of copyright — the advancement of science and the arts.” Slip Op.

---

<sup>1</sup> Disassembly is a form of software reverse engineering that involves translating machine readable object code into a higher level, human readable form.

at 12, n.18. In other words, the Eleventh Circuit in effect directed the district court to find Mnemonic's disassembly for purposes of achieving interoperability to be a fair use.

In another footnote, the Bateman court explained the critical significance of separating idea from expression in the context of computer programs:

It is particularly important to exclude methods of operation and processes from the scope of copyright in computer programs because much of the contents of computer programs is patentable. Were we to permit an author to claim copyright protection for those elements of the work that should be the province of patent law, we would be undermining the competitive principles that are fundamental to the patent system.

Slip Op. at 17, n.21. In support of this proposition, the court cited the Supreme Court's decisions in *Baker v. Selden*, 101 U.S. 99 (1879) and *Bonito Boats v. Thunder Craft Boats*, 489 U.S. 141 (1989), and the Federal Circuit's decision in *Atari v. Nintendo*. In these two footnotes, the Eleventh Circuit signaled that it viewed computer programs as utilitarian literary works which should receive a "thinner" scope of copyright protection than more expressive literary works such as novels and plays.

## ***2. Application of Abstraction-Filtration-Comparison Test to Literal Elements***

Mnemonics had asked the trial judge to instruct the jury to filter out unprotected features appearing in literal as well as nonliteral elements of the Bateman program before comparing it to the Mnemonics program.<sup>2</sup> The trial judge, however, instructed the jury to perform filtration only on nonliteral elements, thus allowing comparison of unprotected literal elements. According to Mnemonics, this "effectively rendered futile [Mnemonics'] efforts to rebut Bateman's evidence of literal similarity with regard to elements of the work for which copyright was claimed, thereby nullifying several of [Mnemonics'] key defenses, including compatibility, efficiency, and standard programming techniques." Slip Op. at 24.

The Eleventh Circuit acknowledged that the Second Circuit in *Computer Associates v. Altai*, 982 F.2d 693 (2nd Cir. 1992), had fashioned the abstraction-filtration-comparison methodology to address the copying of nonliteral elements, and that the First

---

<sup>2</sup> In computer programs, literal elements refer to the source and object code, while nonliteral elements refer to the program's structure and organization.

Circuit in *Lotus v. Borland*, 49 F.3d 807, 815 (1st Cir. 1995), *affd*, 1996 U.S. LEXIS 470 (1996), had suggested that this methodology might be of “little help” when considering the copying of literal elements. The Eleventh Circuit observed, however, that this disagreement was “more a matter of semantics than substance.” Slip Op. at 27. The Eleventh Circuit stated that:

[e]ven if the *Altai* test is limited to nonliteral copying . . . a parallel type of analysis must be undertaken in examining alleged instances of literal copying of computer code .... Whether one chooses to call the consideration of such generally recognized challenges to literal code copying as merger and efficiency ‘filtration’ is of little consequence; what matters is that these well-established ‘defenses’ are considered.

*Id.* at 27-28. The judge in essence instructed the jury not to consider these defenses with respect to the instances of literal copying. This was “a manifest distortion and misstatement of the law,” *Id.* at 27, and sufficient grounds for a new trial given that much of Bateman’s evidence concerned literal copying.

### ***3. The Legal Consequences of Copying Elements Dictated by Compatibility Requirements***

The trial judge had instructed the jury that computer programs are utilitarian articles that contain elements that may be dictated by external factors, “such as compatibility requirements.” Mnemonics on appeal argued that this instruction, while technically correct, did not go far enough because it failed to direct the jury to filter out those portions of the Bateman operating system dictated by the interface with Mnemonics application program.

The Eleventh Circuit concluded that if Mnemonics was arguing that interface specifications are not copyrightable as a matter of law, it was wrong: “[i]t is an incorrect statement of the law that interface specifications are not copyrightable as a matter of law.” Slip Op. at 33. The Eleventh Circuit hastened to add, however, that the trial court had erred “in not instructing the jury on the legal consequences of copying dictated by compatibility requirements.” *Id.* The appellate court then explained what it meant. It reviewed the decisions from other circuits, including *Sega v. Accolade*, *Computer Associates, v. Altai*, *EDI v. SSI*, 26 F.3d 1335 (5th Cir. 1994) and *Atari v. Nintendo*,

which found “that external factors such as compatibility may work to deny copyright protection to certain portions of a computer program.” Slip Op. at 34. Next, it stated that “[w]hether the protection is unavailable because these factors render the expression unoriginal, nonexpressive per 17 U.S.C. § 102(b), or whether these factors compel a finding of fair use, copyright estoppel, or misuse, the result is to deny copyright protection to portions of the computer program.” Slip Op. at 35.

The Eleventh Circuit in yet another footnote emphasized that it was only holding that external factors such as compatibility “*may* negate a finding of infringement.” Slip Op. at 35, n.33 (emphasis supplied). It noted that “[s]uch a finding will depend on the particular facts of a case, and thus it would be unwise for us to formulate a bright line rule to address this issue, given the importance of the factual nuances of each case.” *Id.*

In sum, the *Bateman* court refused to issue a *per se* rule that interface specifications can not receive copyright protection. At the same time, the court provided no less than five different theories under which such specifications would not receive protection.<sup>3</sup> Further, by ordering the district court in the new trial to instruct the jury on “the legal consequences of copying dictated by compatibility requirements,” the Eleventh Circuit presumably insisted that the district court enumerate these theories to the jury. Thus, while the Eleventh Circuit refused to draw a permanent bright line around interface specifications, it certainly penciled a line in.

The *Bateman* court’s reluctance to categorically exclude interface specifications from copyright protection makes sense when one recognizes that “interface specification” is a programming term, not a legal term of art. The *Bateman* decision does not define “interface specification,” and also refers to “interface commands” and “operating system interface” in the same section. The Eleventh Circuit correctly recognized that it would be reckless for it to attach legal results to terms whose meaning and technological significance may change over time. Instead, it provided a framework within which elements related to interoperability should be examined. The wisdom of retaining the freedom to look at “the factual nuances of each case” is demonstrated by the following decision.

---

<sup>3</sup> By including copyright misuse in this list, the Eleventh Circuit breathed new life into this theory, which until now had not gained much legitimacy outside of the Fourth Circuit.

## II. *COMPAQ v. PROCOM*

### A. Facts

Each Compaq server typically holds several hard drives. The server contains the Compaq Insight Manager program (CIM), which, *inter alia*, monitors the operation of the hard drives and generates a prefailure warning when a hard drive's performance falls below a certain level. Compaq then replaces the hard drive free of charge. The CIM thus helps effectuate Compaq's warranty to its customers.

What the CIM actually monitors are five parameters contained in the Monitor and Performance (M&P) partition of the hard drive. The M&P partition appears to be firmware. For each parameter there is a threshold value, and if that threshold value is exceeded, the CIM generates a prefailure warning.

Procom sells Compaq compatible hard drives. The Procom hard drives could interoperate with the Compaq server even if they did not contain any of the parameters in the M&P partition that interacted with the server's CIM; the CIM simply would never issue a prefailure warning. Moreover, the CIM could function if the Procom parameters contained different threshold values; the CIM would then just issue its prefailure warning at a different point in time from when it would have a Compaq hard drive. The CIM could not work properly, however, if the M&P partition contained different parameters, *i.e.*, monitored different functions.

According to the decision, Procom never completely understood how the M&P partition worked. Thus, it simply copied the parameters and the threshold values from the Compaq M&P partition. Compaq sued for infringement.

### B. The District Court's Ruling

The court ruled on a motion for preliminary injunction that the five threshold values constituted a protectable compilation. According to the court, the values are not empirically verifiable facts, but rather the result of Compaq's decision making process. First, Compaq had to make a prediction when the disc drive would actually fail. Then, Compaq had to make a business decision as to when prior to that actual point of failure it was willing to replace the hard drive under its warranty:

In making this decision, Compaq must weigh several considerations such as the cost of replacing drives too early in their life versus the risk of waiting too long to replace the drive and having it fail while in use. It seems unlikely that other drive manufacturers, facing different economic considerations and different customer expectations, would choose the exact same point in time to replace a drive that Compaq chose.

1995 U.S. Dist. Lexis 18535 at \*18. In other words, the court did not view the parameter values as functionally dictated. That court acknowledged, however, that merger would be implicated if the values were solely engineering predictions of when the drives would fail.

While the court extended copyright protection to the particular parameter values established by Compaq, it refused to protect the parameters themselves,

to obtain prefailure warnings through CIM, the drive must have the five numbers representing the five parameters monitored by the program. A third party attempting to gain access to CIM has no choice but to also select those five parameters for observation. If a third party selected other parameters, then any warnings that CIM issued would be meaningless.

*Id.* at \*19-20. The court likewise recognized that the CIM would not function properly if the parameters appeared in the M&P partition in a different sequence. Accordingly, the parameters and their sequence were unprotected as a system or under the merger and *scenes a faire* doctrines.

The court also rejected Procom's fair use argument on the basis that Procom had no real need to copy the actual values of the parameters. The court distinguished this case from *Sega* by emphasizing that in *Sega*, Accolade sought to understand how the Sega product worked and took only what was necessary to achieve interoperability, while Procom never really understood the Compaq product and thus took more than was necessary.<sup>4</sup>

This decision demonstrates the importance of courts understanding the "factual nuances" of the cases before them. Had it reviewed the facts in a cursory manner, it

---

<sup>4</sup> It is worth noting that the *Procom* court interpreted *Sega* as permitting disassembly for purposes of developing a product that competed with the target of the disassembly. *Id.* at \*24. Some commentators have incorrectly suggested that *Sega* permitted disassembly only for purposes of developing products that *attached* to the target of the disassembly.

might have concluded that because business judgments went into the selection of the parameter values, the values, as well as the parameters, should receive copyright protection. Alternatively, the court might have decided that both the parameters and the values related to interoperability and thus neither deserve protection. By closely examining the facts, however, the court succeeded in drawing a line between interface specifications and implementations: the parameters and their sequence are the specifications; the values are the implementations. The court, therefore, could withhold copyright protection from the interface specifications under section 102(b) but still give meaningful copyright protection to the implementations.

Further, because the Procom hard drive could function in the Compaq server even if the CIM did not function properly, the court could have concluded that the parameters were not really necessary for interoperability. The court understood, however, that Procom hard drives would not be as desirable to customers if they were not CIM compatible. In other words, the court recognized that there were degrees of interoperability, and allowed Procom to perform the copying necessary to achieve complete interoperability.

Of course, one may question the court's rulings about the protectability of the values as a compilation -- does the exercise of business judgment, as opposed to scientific judgment, lead to expression? Under the facts of this case, this seems to be the right result.

### **III. CONCLUSION**

These two cases manifest a continuation of the recent trend of the federal courts to narrow the scope of copyright protection to facilitate software interoperability. At the same time, these cases show that courts will not treat defendants' invocation of interoperability as a talisman. Rather, defendants will have to demonstrate that a particular element is truly necessary for interoperability before a court will permit it to be copied.