Moore: Viewpoint

Source – compiler - object is largely nonsense

Most software licence agreements are drafted on the basis that modern software is written in the same way as it was 30 years ago. More worryingly, most software copyright litigation proceeds on the same flawed principle.

Lawyers are still taught that computer software is produced in three simple-to-understand steps:

- First, you write your “human readable” source code;
- then you put your source code through a compiler;
- and you end up with object or machine code which is the low level stuff which runs on computers.

The foregoing process is now rarely the complete picture and the development of computer software has become a highly complex, compound process. If we take, for example, a typical modern software program and break it down into its constituent elements, we are likely to find the following:

- The program is itself written in a highly sophisticated program; and often in more than one. Visual Basic code may, for example, be coupled with Visual C code;
- Some modern software is not compiled at all and not all versions of all development programs produce compiled code – for example, only Visual Basic version 5 onwards is actually compiled, earlier versions were not;
- The finished software program which is produced in a modern development environment usually requires access to files (for example “DLLs”, which may be substantial) which ship with the development environment and which the software developer will have to distribute with their programs in order for those programs to work properly;
- Very few modern programs are written from scratch and the use of “stock code” and “add-ins” developed by third parties (which may be entire programs in themselves licensed on commercial terms) is common; and
- Open source code is finding it’s way into many modern programs, bringing with it some, often, fairly ambiguous licence terms.

Against this backdrop, modern software agreements tend to be overly simplistic. Take, for example, a typical customer request that the copyright in the resulting program should be owned by them. Many software houses routinely sign up to such provisions. Not only is this a pretty dangerous unqualified provision to agree to, and may not actually be needed by the customer, but the software house is frequently immediately put in breach of contract. Why? Well examine the typical elements of a modern software program set out above and it will be immediately clear that the software house simply does not own significant proportions of what it is providing to the customer. In fact, It is only normally in a position to sell (assign) what it created bespoke for this project, and even then, it can prove to be a very difficult task to identify exactly what falls into that category.

When it comes to litigation, the problems can get even worse. The “copyright work” which may have alledgedly been copied cannot easily be viewed in the same way as a self-contained novel and commonly suffers by this over-simplistic comparison.

Moore
Enterprise House
4 Yeomans Grange
Sutton Coldfield
B75 7TP
UK

Telephone: +44 (0) 121 314 8018

Fax: +44 (0) 121 314 8022

E-Mail: enquiries@moorelegal.co.uk

Both the original software and the alledged copy can be clouded by the complexity in their production methods (as set out above). This is not an insurmountable problem but it does require the advisors and the courts to be very clear as to exactly what constitutes the copyright work in question.