

ECIS Seminar on SAS v WPL before the CJEU

Comment on Papers by Pam Samuelson and Thomas Vinje

On Tuesday this week many of us read Advocate General Bot's Opinion in the SAS case with a mighty sense of relief. In it he gave us a firm and full line drawing of the correct answers to Arold J's questions. He has left the Court itself to colour in the segments between his lines. We can only hope that they – the Grand Chamber -- will choose the reds and blues, the golds and silvers that bring out the Adv-General's scrupulous designs. Of course we cannot be sure that they will. That is why the three of us are hard at work on article urging them to do so.

In this short comment, my objective is to place the arguments about software copyright raised in the SAS case in a broader perspective of IP as a whole.

Pam Samuelson has reminded us that a computer is a machine made to function partly by text – text that has been christened 'literary work' by the Software Copyright Directive. But still it is at root a machine rendered functional by electronic hardware and software operating together. It follows that we need to consider how far software can be protected by patents, in comparison with the copyright that its programming supplies.

Software patenting has had a chequered history, particularly in Europe. In the mainframe world of the 1960s and early 1970s, the computing industry's leaders argued against unlocking the gate so as to admit computer software as patentable subject-matter; and this the European Patent Convention of 1973 provided, keeping out at the same time patents for discoveries, scientific theories and mathematical methods, schemes rules and methods for performing mental acts, doing business and presentations of information. But it added that each of these categories of subject-matter was excluded only "as such" – a weasel-like phrase which has caused much bother.

As microprocessing bloomed the search for some form of IP to use against software pirates began in earnest. Some advocated a sui generis form of protection against copying the programming, starting anew from scratch. But copyright became the preferred goal, partly because international relationships were needed for rapidly globalising markets and international legal measures were already set up for literary and artistic works by the Berne Convention and the Universal Copyright Conventions. For the EEC the copyright solution was concretised in the Software Directive, which, as we have been hearing, did a great deal to settle limits on this literary copyright that would allow for interconnectivity between programs, including the use of computer languages and interfaces needed to secure interoperability.

By then, however, litigation in the US and other producer countries was beginning to deal with patents for programs testing the limits of that system under various legislative and judicial formulae. A decade ago, the European Commission decided that the single market needed a Directive to Member States on software patenting. Its Draft, however, was bitterly disputed by a lobby which had an 'interesting' array of justifications for opposing this initiative, and in the end the Commission threw in its towel.

Triumph this may have seemed to the protesters, but what it ignored was that the towel landed back at the feet of courts and tribunals in the EPC system. Pursuing an expansionist vision, Technical Tribunals of the European Patent Office proceeded to rule that the exclusion of computer programs as such from the scope of EPC system did not apply to any claim to a program for use in a computer, and it stuck to this line despite the view of the English Court of Appeal that the express exclusion of computer programs in the EPC could not be ignored by judicial say-so. [This counter-argument led to one EPO Tribunal accusing the UK Court of Appeal of lack of good faith! Hostilities for a time became naked]. Despite such posturing and sabre-rattling, what both the EPO and the English courts accepted was that software patents must in any case be for inventions; and this required showing that the claimed subject-matter was both novel and obvious by considering only the *technical effects* that it produced. Novelty and inventive step are of course the same preconditions of 'patentability' that apply to all

forms of technology. So the EPO's interpretation of the EPC on the subject honours the TRIPs requirement (in Art 27(1)) that patents should be available in all fields of technology without discrimination.

Why does inventiveness continue to lie at the nub of the patent system? Why do modern patent systems test for the necessary qualities of novelty and inventive step by search and examination before any grant is made? Why may the same bases of objection be raised against the patent once granted throughout its life? Essentially because the competitive basis of free economies includes the ability to take ideas from rivals and other providers of information without having to seek their authorisation – to read the publications of others in order to glean for ideas that they may put to their own advantage, to engage in reverse engineering to the same end; and so on. Intellectual property rights create exceptions in the form of market monopolies and that is a vital reason why they should be circumscribed. All too much experience from the 19th century onwards has revealed how a cheap, unregulated patent system becomes prey to the shark-like activities of those who simply masqueraded as inventors.

The patent is the type of IP most likely to prevent industrial competition occurring in any version of a successful product or process. The potential economic impact of a patent can thus be far wider than the copying of literary text or the designing of a mass-made article to give it a particular appearance. Computers that are programmed to function so as to produce particular results should rightly not be subject to program patents that have the effect of monopolising those results, however the program achieves them, unless the capacities of the defined invention are technical, as is required for other machines and processes.

What then of the scope of copyright in programs under the Software Directive? The Directive builds up limits upon copyright in computer programs for the purpose of allowing others to exploit freely the knowledge that they have derived from observing how an existing program behaves in operation. These exclusions are not confined to the writing of programs that will not compete, or will not compete directly, with the program copied. So

they are intended to apply to the ordinary rough-and-tumble of commercial imitation. Pam and Thomas have taken you through the details of the Directive which assist in this process of drawing the copyright line. Counsel for SAS have argued (joined to some extent by the Commission) that the Directive should be read as including within the scope of software copyright 'copying' solely of the functional behaviour of a program which has occurred without access to its object or source code, where the only subject-matter taken is a program language necessary to secure interoperability and interfaces necessary to procure interoperability.

To go so far would be to use copyright to perform the task that the patent system very wisely eschews, save for its carefully circumscribed conditions relating to technical inventiveness. Copyright would be available for a great range of computing outcomes, subject only to the requirement that the program be the result of original work arising from the programmers' intellectual effort. It would exist without any need to define what its function is or to substantiate that it has these basic qualities. Copyright could well become the IPR of choice for all innovative computer products without considering whether their behaviour embodies a function by technical invention. Why bother then with strictures involved in obtaining and then maintaining a patent? Copyright would have tentacles far more embracing than the patent system could provide. If there is support enough within the EU to reverse the limitations clearly intended by the Software Directive and accepted by the Advocate-general in his opinion, this should only be done by a revision of the Directive's terms, and not by decision of the Court of Justice. The current Directive should protect against the misappropriation of actual coding or of the detailed steps built into the program as expressed. It should not give exclusive rights in, for example, all ways of analysing stock exchange transactions or of listing replacement supplies needed by a business, just because this is the object of the program alleged to have been 'copied', or indeed by the taking of coding language or interface protocols.

I have dwelt on this larger aspect of SAS, because, as you have heard, the answers to the questions put to the Court by Arnold J concentrate almost entirely on the meaning to be given to the terms of the Software Directive.

We may hope and expect that the Court will separate ‘ideas and principles’ from ‘expression of the writer’s own intellectual creation’ in words that add meaning through examples to what the Directive itself states. But such positivist legal reasoning will scarcely be satisfactory if the purpose behind it is not also stated as clearly as possible. The Court should acknowledge specifically the dangers of stretching IP for the functions of software in computers beyond the constraints set by the patent system into the realm of any function at all that can be shown to have been copied.

At this point in time we should cheer the Advocate-general for his appreciative and clearly expressed Opinion and urge the Court to be even more explicit about the purposive justifications for coming to the same answers as the Advocate-General to the questions posed by Arnold J.