

# A DEFECTIVE DIRECTIVE?

Stephen Carter looks at the reasons behind the European Union's rejection of rules that would have protected software codes and its consequences

**"N**o Directive is better than a bad Directive". Or at least so thought the European Parliament when it rejected outright the proposed European Directive on the Patentability of Computer Implemented Inventions by 648 to 14 in a July vote. This was expressed both by those in favour of patents for software as well as those against, although their definitions of a "bad Directive" will have been very different.

The aim of the Directive was to harmonise the approach taken by the patent offices and, more importantly, the courts of the EU countries in relation to the extent to which "computer implemented inventions" can be protected by patents. The European

Commission, which originally proposed the Directive, made it clear before Parliament's vote that if the Directive was rejected it would not seek to make any new proposals. Despite suggestions to the contrary in the immediate aftermath of the vote, there are no signs at the time of writing of any move to start down the path towards a Directive once more. This is unsurprising when one considers that it has taken more than three and a half years of involved debate to get to the current situation.

The rejection of the Directive does not mean that software cannot be patented in Europe. The European Patent Office (and other patent offices in Europe) has been granting patents for inventions involving software for many years and it will continue



THE PROBLEM IS IN DRAWING THE LINE BETWEEN TECHNICAL, AND THEREFORE PATENTABLE, SOFTWARE INNOVATIONS AND NON-TECHNICAL SOFTWARE INNOVATIONS FOR WHICH PATENTS SHOULD NOT BE GRANTED

to do so. The independent announcements by open source technology provider Red Hat and Open Source Developer Labs (OSDL) in August 2005 that they will be taking steps to help open source developers to obtain and share software patents suggests a recognition, even within the open source community, that software patents are here to stay. Two recent decisions of the UK Courts also emphasise that the “right sort” of software inventions will continue to be enforced in the UK.

Irrespective of any emotional, ethical or political stance in relation to software patents, it is important for businesses to understand the extent to which software can be patented in Europe (and elsewhere in the world) and for them to devise strategies that acknowledge this while being in line with their overall business aims and principles.

#### European Patent Office

One might rightly question why so many patents for software-related innovations (in excess of 30,000) have been granted by the European Patent Office (EPO) when the law they are applying states that “programs for computers (as such)” (along with “schemes rules and methods for doing

thing (not an assumption that everyone agrees with), one could argue that it is wrong to exclude protection for a technical innovation simply because it is implemented in software. Why, for example, should a person be able to secure patent protection for improved electronic circuitry for deploying the air bag in a car but not for the same improvement obtained with new software code?

This does not mean to say that the EPO will grant patents for any new software code. Far from it. The Boards of Appeal at the EPO have made it very clear that software innovations that do not exhibit the necessary technical advance, for example where the advance is merely in the field of economics or business, will not benefit from patent protection. The problem, however, is in drawing the line between technical, and therefore patentable, software innovations and non-technical software innovations for which patents should not be granted.

The approach now adopted by the EPO in deciding which side of the line a particular case falls follows the reasoning of an Appeal Board decision from April 2004 (the *Hitachi* decision) that considered the patentability of claims to a computerised “Dutch auction”. The innovation sought to

## THOSE BUSINESSES THAT HAVE SOUGHT TO PATENT THEIR SOFTWARE DEVELOPMENTS IN EUROPE SHOULD NOT BE PUT OFF DOING SO BY THE EU PARLIAMENT’S REJECTION OF THE DIRECTIVE

business” and other more abstract things) are not considered inventions that can be protected by patents.

The answer lies in a long line of decisions by the Boards of Appeal at the EPO, starting in the mid-1980s, that maintained that the exclusion did not apply to software innovations that were essentially “technical” in character (because their technical nature made them something more than programs for computers “as such”). So, for example, claims to a method of image enhancement, where the innovation was in the algorithm used, have been allowed, as have claims to x-ray apparatus where the only difference from what was already known was the software that controlled the apparatus, because the overall contribution to society was a technically improved apparatus.

A great many granted software patents clearly fall into this “technical category”. If one assumes that monopoly protection for technical innovations in general is a good

address a problem with existing online auctions, namely the fact that network delays meant that bids did not necessarily arrive at an auction site in the order they were placed by bidders at different remote locations. This problem was solved by adopting an approach in which each bidder submitted a “desired price” and a “maximum price” at the outset and the auction was then completed based on these submitted prices, avoiding the need to transmit further bids during the course of the auction. The patent application claimed an automated auction method carried out at a server computer operating a series of steps (that were specified in the claim) and a computerised auction apparatus for conducting an auction, via a network, following those steps.

The Appeal Board ultimately reached the conclusion that the claimed invention was not patentable because, in its view, it did not provide a technical solution to the problem being addressed. Rather, the solution was



based entirely on modifications to the auction process.

The Board reached this conclusion by first asking whether the claimed method and system included technical features, which they did as they referred to a “server computer”, “client computers” and a “network”. This meant the system and method did not fall foul of the exclusion for computer programs as such. The Board went on, however, to consider whether the method and system involved an inventive step, a necessary requirement for the grant of a patent, in the sense that the method and system were not obvious over what was already known in the relevant field.

Importantly, in answering this question the Board only paid attention to the technical features of the method and system. Once the non-technical auction-related features were stripped away all that was left were conventional technical features and there was therefore no inventive step. In effect, what

the Board was looking for was a technical solution to a technical problem. It did not find one and the application was rejected.

There is no complete answer as to what might lend the necessary non-obvious technical character to a software related invention where the application of the invention is in a field of economics or business. However, based on the patents that have been granted in this field by the EPO, it is possible to suggest types of advance that are likely to be considered sufficiently technical. These include improved user interfaces, more efficient use of resources such as memory, displays and communications links, improved reliability, quicker operation, improved control and improved access to or management of data. It was also suggested in the “Dutch auction” case that new steps in a business process that are included specifically in order that the process is suited for performance on a computer might lend the process the necessary technical character. These advances all have in common that they relate to the technical operation of the computer system and are to a large extent independent of the specific commercial application of the software.

#### The UK

Two decisions from the UK’s High Court, both handed down on the same day in July 2005 hot on the heels of the European Parliament’s rejection of the Directive, confirm that patents are available to protect software innovations that have the necessary technical character.

The case of *Halliburton Energy Services Inc v. Smith International (North Sea) Ltd and others* related to a computer system for designing drill bits for drilling in rock. It was held that the design process alone was too abstract to be considered sufficiently technical in character for a patent, but that a process including manufacture of a drill bit to the design would have been suitable subject matter for a patent even though the innovation would lie essentially in the design process (implemented in software). The judge confirmed that patents could be granted for software innovation but only if the contribution made by the inventor lies in a technical effect and not merely in one of the excluded (non-technical) fields.

The other case – *CFPH LLC’s Application* – concerned two UK patent applications of CFPH, a Cantor Fitzgerald company. The applications related to networked (meaning “internet”) wagering methods with an emphasis on reducing processing delays to a minimum so that in cases where prices are

changing continuously a client is provided with the most up to date information before placing a bet. The UK Patent Office rejected the applications, arguing that the claimed inventions related to non-technical changes to a business method in order to overcome the delay-related problems in previous systems. CFPH appealed to the High Court where the judge upheld the decision of the patent office. In doing so, he explained that the way to approach the question of deciding whether something is a patentable invention is to "identify what is the advance in the art that is said to be new and non-obvious"; and "then to determine whether it is both new and not obvious under the description 'an invention'".

There is no definition of "an invention" given by the relevant statute, but there is a list of things that are said not to be inventions, including programs for computers and business methods as such.

The judge pointed out that, especially in those non-borderline cases where something clearly is or clearly is not "technology", it will often be possible to short-cut the two steps above by asking "Is this a new and non-obvious advance in technology?"

Considering business methods, and to use the judge's words again, "a new advance in business methods, of itself, cannot supply that element of novelty and non-obviousness that is required to support a patent claim." Significantly, however, the judge also made it clear that a patent claim is not excluded merely because it relates to an invention in the field of commerce. Moreover, he said that "if it is possible that the claim is capable of being supported on other grounds [meaning that the advance that is said to be new and non-obvious can be considered "an invention"], the business context is not irrelevant. It may well be relevant background on obviousness." So, the specific business context of a technical invention, in particular the common practices and prejudices in that field, may be very relevant to deciding whether a patent is granted or not.

On computer programs, the judge also made it clear that the mere presence of a computer program as an element of a claimed invention does not of itself preclude the grant of a patent. He said that "there are many artefacts that operate under computer control (such as the automatic pilot of an aircraft) and there are many processes that operate under computer control (such as making canned soup). A better way of doing those things ought, in principle, to be patentable."

The judge went on to suggest that the

question to ask was: "is it (the artefact or process) new and non-obvious because there is a computer program (in which case it is excluded)? Or would it still be new and non-obvious in principle even if the same decisions and commands could somehow be taken and issued by a little man at a control panel, operating under the same rules? For if the answer to the latter question is 'Yes' it becomes apparent that the computer program is merely a tool, and the invention is not about computer programming at all. It is about better rules for governing an automatic pilot or better rules for conducting the manufacture of canned soup."

Of course, as the judge points out, "if it were better rules for running a business the

**TWO DECISIONS FROM THE UK'S HIGH COURT CONFIRM THAT PATENTS ARE AVAILABLE TO PROTECT SOFTWARE INNOVATIONS THAT HAVE THE NECESSARY TECHNICAL CHARACTER**

idea would not be patentable." Soon after these two decisions, the UK Patent Office issued a Practice Notice stating that its practice would be changed to follow the rulings in the decisions. Examiners at the Patent Office are now applying the two-step test set out in the *CFPH* judgement and the suggested "shortcut".

#### Practical strategies

Given the still far from clear backdrop to patenting software related inventions in Europe, what strategies should businesses adopt?

First, those businesses that have sought to patent their software developments in Europe should not be put off doing so by the EU Parliament's rejection of the Directive. Patents continue to be available for such developments, so long as the advance is not purely in one of the excluded fields (such as business methods). Care is needed, however, in the drafting and prosecution of patent applications in this field to ensure that a good, technical invention that is implemented in software does not inadvertently fall to the wrong side of the tests that are now being applied by the EPO and UK Patent Office.

Second, for those businesses that have not historically sought to protect their software

innovations with patents, now might be the time to re-visit their strategy in this regard. As the moves by Red Hat and OSDL referred to above demonstrate, patent portfolios can have uses other than the aggressive pursuit of competitors. A portfolio of patents can be a useful shield against others and for a small, growing business can be invaluable in helping to attract investment and gives the potential to create revenue streams at an early stage through licensing.

Of course, patents are expensive and, particularly for small and medium-sized enterprises, budgets will not necessarily stretch to building large portfolios of patent applications. Therefore, thought also needs to be given to other defensive strategies such as publication of development work to spoil the chances of others later obtaining patents that hinder commercial enterprises and, particularly in those cases where commercial sensitivities preclude publication, it may be prudent to watch the patent activities of at least close competitors so that possible problems can be flagged up and proactive steps taken to mitigate them as early as possible.

#### The future?

It seems unlikely that there will be any appetite in the near future for legislators to grasp once more the thorny issues surrounding the question of whether or not patents should be available for software innovation.

However, the number of patents being filed and granted in Europe is likely to continue to increase, probably at a rapid rate. As a result, there may be more patent disputes, but it is only ever a very small percentage of granted patents that are enforced.

It is also possible that there might be more sharing of patents following the model of OSDL or defensive portfolios being built in the manner Red Hat are supporting. Moreover, there may be more patents being "donated" to the open source community as IBM, Novell, Sun Microsystems and Nokia have done during the course of this year.

Whatever the future brings, patents for software are here to stay and businesses need strategies that acknowledge this reality. In a world where software is critical and in which software patents are becoming more prevalent in Europe – with no prospect of that changing in the short term – a strategy that involves burying your head in the sand is not one to be recommended. IP

*Stephen Carter is a partner in the patents practice of Mewburn Ellis LLP. He can be contacted at [stephen.carter@mewburn.com](mailto:stephen.carter@mewburn.com)*