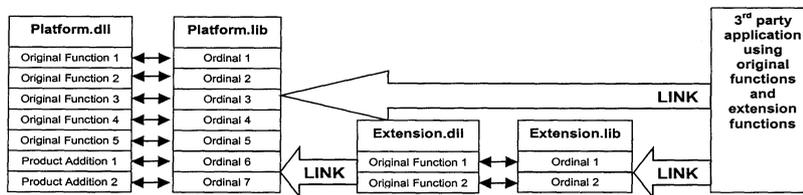


5 The invention is concerned with the operation of a computing device that has a DLL containing functions that can be accessed by an executable program where each function is associated with an ordinal number. The DLL has two parts, a first part and a second part, with each part containing a number of functions. The executable program directly links to the functions in the first part through the ordinal numbers. To access the functions in the second part, the executable program links to the DLL indirectly through a “further library” or an “interface”. I have reproduced Figure 4 from the specification below, which shows this in a diagrammatic form.



4/5

Fig. 4.

6 As currently amended, there are three independent claims (claims 1, 8 and 9). Claim 1 reads:

A method of operating a computing device having an operating system and a dynamic link library containing a plurality of functions accessible by an executable program, each function in the dynamic link library being associated with an ordinal number, the method comprising:

Providing the dynamic link library as a first part and an extension part each containing one or more of the plurality of functions;

Causing the executable program to link to functions in the first part directly by means of the associated ordinal numbers; and

Causing the executable program to link to functions in the extension part indirectly via a further library containing additional functions

7 Claim 8 relates to computer software arranged to cause a computing device to operate in accordance with the above method steps; and claim 9 relates to a computing device comprising an operating system, a dynamic link library and an

interface which causes the executable program to link to the library to carry out the above method steps.

8 Three amended sets of claims were submitted for discussion at the hearing (entitled main, first auxiliary and second auxiliary claims). In summary:

(i) the main claims are the same as the claims above except that: (i) the reference to an operating system has been deleted from the independent claims and (ii) the term “ a further library containing additional functions” has been replaced in the independent claims with the term “an interface”;

(ii) the first auxiliary claims include the amendments made in the main claims with the following additional amendments: (iii) the independent claims specify that, once a function is requested by the executable program, and the function is linked according to its position within the dynamic link library (that is, according to whether it is held in the first or the extension part), the function is subsequently performed; and

(iii) the second auxiliary claims include the amendments made in the main claims and the first auxiliary claims, with the following additional amendments: (iv) the independent method claim is defined as a method of managing resources of a computing device; (v) the independent claims specify that the plurality of functions are suitable for managing the resources of the device; and (vi) the independent claims specify that the execution of the requested function is such as to manage the resources of the device.

The full text of the independent claims for each set of amended claims is set out in the Annex attached to this decision.

The law and its interpretation

9 The relevant parts of Section 1(2) read (emphasis added)

“it is hereby declared that the following (among other things) are not inventions for the purposes of this Act, that is to say, anything which consists of –

- (a) a discovery, scientific theory or mathematical method;
- (b) a literary, dramatic, musical or artistic work or any other aesthetic creation whatsoever;
- (c) a scheme, rule or method for performing a mental act, playing a game or doing business or **a program for a computer;**
- (d) the presentation of information;

but the foregoing provision shall prevent anything from being treated as an invention for the purposes of this Act **only to the extent that a patent or application for a patent relates to that thing as such.**

10 The correct approach for assessing an application was handed down by Jacob LJ

in *Aerotel Ltd v Telco Holdings Ltd (and Others) and Macrossan's Application* [2006] EWCA Civ 1371. In this case the Court reviewed the case law on the interpretation of Section 1(2) and approved a new four-step test for the assessment of patentability:

- 1) Properly construe the claim;
- 2) Identify the actual contribution;
- 3) Ask whether it falls solely within the excluded matter;
- 4) Check whether the contribution is actually technical in nature.

- 11 As stated at paragraphs 45 – 47 of the judgment, reconciling the new test with the earlier judgments of the Court of Appeal in *Merrill Lynch*¹ and *Fujitsu*², the fourth step of checking whether the contribution is technical may not be necessary because the third step – asking whether the contribution is solely of excluded matter – should have covered the point. This part of the test is in effect a longstop to be invoked where the invention passes the first three steps.

Arguments and Analysis

Added subject matter

- 12 At the outset, I should draw attention to the replacement of the phrase “ a further library containing additional functions” with the term “an interface” in the proposed claim sets. I put it to Miss Harper that the use of this phrase was not supported by the description as filed. Miss Harper disagreed and considered that the amendment was allowable based on the embodiment described at page 10, lines 18-29. However, this is not something I need to dwell on as I do not consider it impinges on the main issue before me and I make no finding on the matter. Suffice to say, should I find the claimed invention to be patentable, it will be an issue for the Examiner to consider during the continuing examination process.

Applying the Aerotel/Macrossan test

- 13 I shall focus my attention on the method claim 1 of each set of claims. It is convenient to deal with the current, main and first auxiliary claims together since it seems to me that the substance of these claims which relate to a method of operating a computer device is broadly the same even though the claim language differs between the various claim sets. However, the second auxiliary claims are directed to a method of managing resources in a computing device and I shall deal with this as a separate issue.

Construing the claims

- 14 I shall start by applying the first step of the test. It is clear that the current, main and first auxiliary claims all relate to a method of operating a computing device to cause an executable program to access a DLL through an interface. In the case

¹ [1989] RPC 561

² [1997] RPC 608

of the second auxiliary claim, the same method of accessing the DLL is used but claim is worded as a method of managing resources in a computing device. There was some discussion at the hearing of what was meant by “resources” in this context. I understood Miss Harper to mean that this term was intended to cover both the hardware and software resources within the computing device. This is consistent with the description given in the application eg at page 1, and I am therefore content to accept Miss Harper’s explanation.

Contribution made by the invention

- 15 In paragraph 43 of the *Aerotel/Macrossan* judgment, the Court restated its previous findings that in identifying the actual contribution it is substance that matters rather than the form of claim. The judgment says “What has the inventor really added to human knowledge perhaps best sums up this exercise”.
- 16 Miss Harper presented the contribution made by the invention in the current, main and first auxiliary claims as “the improved reliability of a computing device, enabled by the provision of a novel interface.” This interface enabled an executable program to reliably access the functionality available on the computing device, regardless of any additions or amendments made to the available functionality.
- 17 Miss Harper explained that this improved reliability came about because

“.. the insight that the inventor here had was to provide a mechanism for controlling what happens when the application requests functionality and ensuring that the request from the application, the core of the functionality coming from the application is intercepted by an interface; and the interface keeps a record of exactly what functionality is at what ordinal number, so that interface can then forward on the call to the appropriate ordinal number to ensure that the correct functionality is carried out by the device in accordance with what the application is requesting. So what is provided here is an interface between the executable program that is requesting the functionality and the functionality itself.”
- 18 Turning to the contribution made by the invention claimed in the second auxiliary claims, Miss Harper stated that the problem of reliability was solved through control of a physical process, namely through managing the resources of the computer device. To this end, it was known for an operating system to have an Application Programming Interface (API) that allowed an application supplied by a third party to use the computing device. In a conventional system, an executable program would issue a call to the API in order to control some aspect of the device hardware. However, a problem arose when the device manufacturer or operating system provider changed the API by adding or updating its functionality. The third party application would then not necessarily function correctly since the API could no longer be relied upon to provide the correct functionality. The solution to this problem lay in the provision of the interface that allowed the application calls to the API to be intercepted and where necessary mapped onto new functionality.
- 19 I pressed Miss Harper on the nature of the physical process that was being controlled via the interface. In reply, she used the hypothetical example of a camera application requesting the device to take an image. Here the third party application would use the API to cause the device to take the image. If however,

the API had changed then this would result in unreliable functioning of the device. The use of the “interface” of the claimed invention would ensure that this could not happen as the call made by the application would be routed by the interface to the correct function within the API. The claimed invention therefore achieved this reliability by better control of the apparatus.

- 20 Whilst the use of the “interface” may well lead to improved reliability, it seems to me that this is an advantage of the method claimed rather than the actual contribution in the sense set out in paragraph 43 of the *Aerotel/Macrossan* judgement. In my view, the contribution made by the invention as claimed in all the claim sets lies in the interface which enables an executable program to access the functionality available on the computing device, regardless of any additions or amendments made to the available functionality by a third party.

Does the contribution fall solely within the excluded subject matter?

- 21 Miss Harper argued that contribution provided by the new interface solved a technical problem and hence went beyond a computer program as such. To support her view, she took me through recent Office and EPO decisions that had been decided or reviewed under the *Aerotel/Macrossan* judgment. In each case Miss Harper sought to show me that claims in which a computer program formed an integral part of the invention had been held to be patentable. She also reminded me of the *Aerotel/Macrossan* decision which warns of the potential trap in saying “Well, the claim involves the use of a computer program so it must be excluded”.
- 22 Miss Harper began with the *WesternGeco*³ decision in which a claim to a data processing technique to produce an improved seismic image was allowed as going beyond a mere mathematical method. Drawing a parallel with the present case, she argued that the interface of the present invention enabled the functionality of the device to be reliably located and executed resulting in an improved computer program routing mechanism which produces as its result the reliable performance of a device.
- 23 Miss Harper drew my attention to paragraph 34 of the decision in which the Hearing Officer commented that “it is fairly obvious to me that the contributions made do not relate to specific instruction [*sic*] for processing data within a computer program processor” and concludes that the relevant claims are thus not excluded as computer programs as such. In the present case, the claims involve no specific instruction for processing data within a computer program processor. Instead, she said, the invention involved a more fundamental, structural improvement on the state of the art, in providing a new interface enabling reliable linking between programs and required functionality. Finally, on *WesternGeco*, Miss Harper referred to paragraph 26 which discusses the concept of inventors’ impetus for making improvements. It is commented that in the *Vicom*⁴ case the impetus came “not from a desire to improve a mathematical method for the sake of it but to improve the resolution of a physical image; this, it seems to me, points to the contribution as extending beyond mere improvement of a mathematical

³ BL O/135/07

⁴ EPO Board of Appeal case T 208/84

method *per se*". She argued that, in the present case, it could be seen from the description of the prior art that the inventor's impetus came from a desire to improve the technical operation of real devices: it was not an abstract wish to improve computer programs. The contribution here (she said) was tied to the problem of ensuring device functionality when an executable program requests the functionality. This, in her view pointed to patentability of the claimed invention.

- 24 Turning to the Office practice notice on patentability⁵ and the case study on the *Touch Clarity*⁶ decision, Miss Harper noted that the contribution was found to extend beyond a computer program because the relevant claims included the limitation of controlling a robot, which was not solely in the field of computer programming. In the present case, she argued performing the required functionality of the device similarly went beyond computer programming.
- 25 Miss Harper also argued the claimed invention was at least analogous to *Fisher-Rosemount*⁷ where the contribution was held to go beyond a computer program because it "includes control of a physical process" (paragraph 30). Specifically, in the present case, physical actions were performed by the computing device which, as a result of the novel interface, could execute its full functionality as required.
- 26 Miss Harper then highlighted paragraph 6 of *Vicom*, where the Technical Board of Appeal stated its opinion that "even if the idea underlying an invention may be considered to reside in a mathematical method a claim directed to a technical process in which the method is used does not seek protection for the mathematical method as such". At paragraph 12, it is further stated that "a claim directed to a technical process which process is carried out under the control of a program (be this implemented in hardware or in software) cannot be regarded as relating to a computer program as such ...". At least in the first and second alternative claim sets of the present case, what was claimed in claim 1 was a technical process involving an arrangement of computer programs and *also* involving the physical step of performing functions on the computing device. In her view, this meant the contribution was not a computer program as such.
- 27 Finally, Miss Harper drew my attention to paragraph 92 of *Aerotel/Macrossan* which summarises the Court of Appeal's judgment in *Gale's Application*, and observes that "it was decided that the computer program exclusion extends not merely to the code constituting a program, but that code as embodied on a physical medium which causes a computer to operate in accordance with that code. More is needed before one is outside the exclusion – as for instance a change in the speed with which the computer works". In the current case, she submitted, the requisite "more" was present in the claims, at least for the first and second alternative versions: there was a change in the very fact that the computer works, never mind its speed. In her view, the present claims must therefore be outside the exclusion.

⁵ <http://www.ipo.gov.uk/patent/p-decisionmaking/p-law/p-law-notice/p-law-notice-subjectmatter.htm>

⁶ BL O198/06

⁷ BL 0147/07

- 28 It seems to me that if the contribution made by the invention, considered as a matter of substance rather than the form of claim (see paragraph 43 of *Aerotel/Macrossan*), consists solely of a program for a computer, then the invention will be excluded under section 1(2) and will not be saved by reference to a possible technical effect. I should not now give the applicant benefit of any doubt as to whether the invention arguably covers patentable subject-matter, as paragraph 5 of the judgment makes clear. Nevertheless, it bears emphasising that the exclusion of section 1(2) applies only where the invention relates to excluded matter as such. I must therefore be satisfied that the contribution lies solely in a computer program before finding against the applicant. I observe that Office decisions are not binding on me and, as I said at the hearing, I must make my decision based on the facts of this case.
- 29 I have carefully considered Miss Harper's submissions which, if I have understood her correctly, were cited by way of analogy to show that claims to a technical process involving an arrangement of computer programs and *also* involving the physical step of performing functions on the computing device were patentable.
- 30 So, does the contribution fall solely within the excluded subject matter? As stated above, in my view, the contribution is the interface which enables the executable program to link to the functions in the DLL, where the DLL is the API through which control of the computer hardware is enabled. I cannot see that this control is changed by the claimed invention – it seems to me that the application merely describes a different way in which I can call the functionality held within the DLL and not the way the API operates to control the resources of the computer.
- 31 The application also makes it clear that only applications that require access to the additional functionality of an updated DLL will be handled by the interface. The actual role of the DLL has not been changed by the claimed invention - it still provides exactly the same functionality as it had before. What has changed is the manner in which the DLL has been accessed – it is now done by an additional piece of software in the form of the interface. I therefore find that the contribution made by the claims on file, the main claims and the first auxiliary claims boils down to nothing more than a computer program and hence is excluded from patentability.
- 32 Turning now to the second auxiliary claims. Despite Miss Harper's valiant efforts to persuade me otherwise, I cannot agree that managing the resources of a computing device is a physical process in the sense of *Touch Clarity* or *Fisher-Rosemount* referred to earlier in this decision. It's what the operating system of a computer does. The interface provided by the claimed invention does not alter this fact; it only alters the routing of calls to the DLL. It may provide a different method of accessing functions in the DLL related to the resources on the device, but it does not provide a method of managing resources – that is done by the DLL as it was before.
- 33 I would add one further comment to reinforce this point. There will, as envisaged by the claims be situations where the executable program does not need to operate through the interface. (Fig 4 provides a good diagrammatic example of this) In these cases it is clear that the DLL functions exactly as it did before the

invention and this shows that it is the way in which the DLL is accessed and not the DLL itself that forms the contribution. In my view, this interface is nothing more than an executable application talking to a DLL through another DLL and is clearly a computer program. Indeed Miss Harper stated during the hearing:

"I think what the interface controls is the access by one computer program (the executable program, as it is called in the claims) – the access by the executable program of functionality. The interface provides the route for the call from the executable program to the functionality. So I guess it is controlling how one computer program (the executable program) interacts with another (the functionality in the API). But our invention goes beyond that, in controlling the device on which all this is implemented. The functions in the API control the operation of the device"

- 34 I therefore find that the contribution made by the second auxiliary claims is nothing more than a computer program for allowing an executable program to link to a DLL through a further computer program interface and hence sits squarely within the computer program exclusion.

Check the contribution is actually technical

- 35 I do not need to apply the fourth step of the test as the contribution has failed the third step.

Conclusion

- 36 I therefore find the invention as claimed in the current, main, first and second auxiliary claims is excluded under Section 1(2) in that it relates to a computer program as such. At the hearing Miss Harper suggested that she may wish to file a further set of claims. However, I do not see any possible saving amendment and I therefore refuse the application.

Appeal

- 37 Under the Practice Direction to Part 52 of the Civil Procedure Rules, any appeal must be lodged within 28 days.

MRS S E CHALMERS

Deputy Director acting for the Comptroller

Annex

Main Claims

Claim 1:

A method of operating a computing device having a dynamic link library containing a plurality of functions accessible by an executable program, each function in the dynamic link library being associated with an ordinal number, the method comprising:

providing the dynamic link library as a first part and an extension part, the first part and the extension part each containing one or more of the plurality of functions;

causing the executable program to link to functions in the first part directly by means of the associated ordinal numbers; and

causing the executable program to link to functions in the extension part indirectly via an interface.

Claim 9:

Computer software arranged to cause a computing device to operate in accordance with a method according to any one of claims 1 to 8.

Claim 10:

A computing device comprising:

a dynamic link library containing a plurality of functions accessible by an executable program, each function in the dynamic link library being associated with an ordinal number; and

an interface;

wherein the dynamic link library is provided as a first part and an extension part, the first part and the extension part each containing one or more of the plurality of functions; and

wherein the computing device causes the executable program to link to functions in the extension part indirectly via the interface, while the executable program is able to link to functions in the first part directly by means of the associated ordinal numbers.

First Auxiliary Claims

Claim 1:

A method of operating a computing device having a dynamic link library containing a plurality of functions accessible by an executable program, each function in the dynamic link library being associated with an ordinal number, the method comprising:

providing the dynamic link library as a first part and an extension part, the first part and the extension part each containing one or more of the plurality of functions; and

if the executable program requests use of a first function, contained in the first part, causing the executable program to link to the said first function directly by means of the associated ordinal number, and subsequently performing the said first function; and

if the executable program requests use of a second function, contained in the extension part, causing the executable program to link to the said second function indirectly via an interface, and subsequently performing the said second function.

Claim 9:

Computer software arranged to cause a computing device to operate in accordance with a method according to any one of claims 1 to 8.

Claim 10:

A computing device comprising:

a dynamic link library containing a plurality of functions accessible by an executable program, each function in the dynamic link library being associated with an ordinal number; and
an interface;

wherein the dynamic link library is provided as a first part and an extension part, the first part and the extension part each containing one or more of the plurality of functions; and
the computing device being arranged to:

if the executable program requests use of a first function, contained in the first part, cause the executable program to link to the said first function directly by means of the associated ordinal number, and subsequently perform the said first function; and

if the executable program requests use of a second function, contained in the extension part, cause the executable program to link to the said second function indirectly via the interface, and subsequently perform the said second function.

Second Auxiliary Claims

Claim 1:

A method of managing resources of a computing device having a dynamic link library, the dynamic link library containing a plurality of functions accessible by an executable program and suitable for managing the resources of the device, each function in the dynamic link library being associated with an ordinal number, the method comprising:

providing the dynamic link library as a first part and an extension part, the first part and the extension part each containing one or more of the plurality of functions; and

if the executable program requests use of a first function, contained in the first part, causing the executable program to link to the said first function directly by means of the associated ordinal number, and subsequently performing the said first function so as to manage the resources of the device; and

if the executable program requests use of a second function, contained in the extension part, causing the executable program to link to the said second function indirectly via an interface, and subsequently performing the said second function so as to manage the resources of the device.

Claim 9:

Computer software arranged to cause a computing device to operate in accordance with a method according to any one of claims 1 to 8.

Claim 10:

A computing device comprising:

a plurality of resources

a dynamic link library containing a plurality of functions accessible by an executable program and suitable for managing the resources of the device, each function in the dynamic link library being associated with an ordinal number; and

an interface;

wherein the dynamic link library is provided as a first part and an extension part, the first part and the extension part each containing one or more of the plurality of functions;

the computing device being arranged to:

if the executable program requests use of a first function, contained in the first part, cause the executable program to link to the said first function directly by means of the associated ordinal number, and subsequently perform the said first function so as to manage the resources of the device; and

if the executable program requests use of a second function, contained in the extension part, cause the executable program to link to the said second function indirectly via the interface, and subsequently perform the said second function so as to manage the resources of the device.