



7th October 2010

PATENTS ACT 1977

APPLICANT

Intuit Inc.

ISSUE

Whether patent application number
GB0917486.3 complies with section 1(2)

Mr H Jones

DECISION

Introduction

- 1 Patent application GB0917486.3 was filed on 6th October 2009 and has a priority date of 10th October 2008 derived from a pending US application (12/249429). The application has not been published nor has it been searched. The examiner has instead issued a report under section 18(3) of the Act following the guidance in paragraph 17.99 of the Manual of Patent Practice and the decision of the Court in *Rohde & Schwartz*¹. This report sets out an objection on the grounds that the application is not patentable under section 1(2) of the Act.
- 2 The applicant has been unable to convince the examiner that the application is patentable under section 1(2) and a hearing was held on 6th September 2010. The hearing was attended by Mr Simon Davies of D Young & Co as patent attorney representing the applicant.
- 3 The primary issue before me is whether the application complies with section 1(2) of the Act. There is also a secondary issue of whether the application as amended incorporates additional matter. However, in view of the fact that the application has not yet been searched or fully examined for any issues of clarity and support, I do not see the need to deal with this issue in this decision. If I find in favour of the applicant I am sure that the attorney and examiner can resolve this issue without my intervention. At the hearing, both Mr Davies and the examiner agreed with this point of view.

The application

- 4 The application comprises a method claim (claim 1) and an apparatus claim (claim 12), and is concerned with the operation of an application programming

¹ [1980] RPC 155

interface (API) between the persistence tier and middle tier of a three tier programming model. To avoid any problems of construction, the persistence tier includes the database and storage system whilst the middle tier is often referred to as the business tier and contains the business logic implemented in programs or applications. The top tier is the presentation tier and provides an interface, e.g. web pages, between the users and the application.

- 5 In operation the API is structured to receive a request to execute a command at the persistence tier. The API then determines the entity that is affected by the command and uses this to identify a function in the middle tier that is associated with both the entity and the command. An instruction is then sent to the middle tier to execute the function, and if the API receives confirmation that the function has been executed in the middle tier then the original command is executed at the persistence tier.
- 6 The application provides a useful example of the operation of the system on page 11 between lines 8 and 28. In this example a command is received at the API to update the salary record of an employee. The entity in this case is the employee table in the database. The API then identifies a function in the middle tier such as a salary range verification function or check for the authorisation status of the person issuing the command. The middle tier will then execute these functions and if the API receives a confirmation that it is an allowable change then it will execute the command. In this case it will write the new salary value into the employee table.

The law

- 7 The relevant parts of section 1(2) read as follows:

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) ...;

(b) ...;

(c) a scheme, rule or method for performing a mental act, playing a game or doing business or a program for a computer;

(d) ...;

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.

- 8 Current IPO examination practice is to use the structured approach set out by the Court of Appeal in its judgment in *Aerotel/Macrossan*² for deciding whether an invention is patentable. The test comprises four steps:

² *Aerotel Ltd v Telco Holdings Ltd and Macrossan's Application* [2006] EWCA Civ 1371

- 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.

9 More recently, the Court of Appeal in the case of *Symbian*³ confirmed that this structured approach is one means of answering the question of whether the invention reveals a technical contribution to the state of the art. In other words, *Symbian* confirmed that the four-step test is equivalent to the prior case law test of 'technical contribution', as per *Merrill Lynch, Gale* and *Fujitsu*. Mr Davies accepted that this was the correct approach that should be followed.

Arguments and analysis

10 In this particular application there appears to be little difficulty in applying the first two steps of the *Aerotel* test. Both Mr Davies and the examiner are of the same view as to claim construction and the contribution. In short, both agree that the contribution is a computer program and their disagreement lies in whether the computer program makes a technical contribution.

11 At the hearing it was also agreed that one way of identifying whether an application made a "technical contribution" was to use the signposts set out by Lewison J in *AT&T/CVON*⁴ and subsequently used by Mann J in *Gemstar v Virgin*⁵. I agree that these signposts provide a useful guide in determining whether the present invention makes a technical contribution.

12 Mr Davies presented his argument on two bases. First he argued that the application was similar in many respects to *Symbian* and that since this was allowed then so too should this case. He then went on to argue that it made a technical contribution because he felt it passed two of the *AT&T/CVON* signposts, specifically a) whether the claimed technical effect operates at the level of the architecture of the computer; that is to say whether the effect is produced irrespective of the data being processed or the applications being run (the second signpost), and b) whether the perceived problem is overcome by the claimed invention as opposed to merely being circumvented (the fifth signpost). In particular, Mr Davies argued that the computer program was part of system architecture and would therefore comply with the second signpost. He also argued that it solved a technical problem rather than circumventing one and thus fulfilled the fifth signpost.

13 Turning to his first argument, Mr Davies made the point that in *Symbian* the executed program would call functions in Dynamic Link Libraries (DLL) in the system. As he saw it, the key component of the invention was the interception of a call to a function and the subsequent mapping to another function if required,

³ [2009] RPC 1

⁴ [2009] EWHC 343 (Pat)

⁵ [2009] EWHC 3068 (CH)

which led to the better, more reliable computer that was considered by the Court to be allowable.

- 14 He saw clear similarities with the current application in that commands from the second level were intercepted and mapped to functions. I am not swayed by this view, though I do acknowledge that on the face of it there are similarities. The particular difficulty I have in accepting this view is that I cannot see how this would result in a better or more reliable computer. To my mind there is a distinct difference between this application and *Symbian*. First of all, *Symbian* dealt with the relationship between applications and functions within the underlying operating system. The maintenance of this link is clearly of fundamental importance in ensuring that the computer continues to function – if such a link were to break then the computer would be less reliable. Secondly, the system set out in *Symbian* would function for any program written to the same standards. In effect, it would always work irrespective of which application executed the call to the DLL.
- 15 In my opinion the current application lacks both these features. The application model is in three tiers, each of which may comprise many executable modules to function. However, unlike *Symbian* they interact with each other and not with any functionality of the machine or computer on which they run. The computer operates as it always has done in that each tier is supported by functions of the operating system and any interface between the two is unchanged. The application may well be highly complex involving multiple components but it remains an application running on what is to all intents and purposes a standard general purpose computer. It appears to me that what the applicant has achieved is to make a more reliable application not a more reliable computer.
- 16 It appears also that data transferred between the two tiers does not change. The command is intercepted from the second tier and is effectively stalled whilst a check is carried out. Once the required checks have been made the command is then released and the data written to the persistence tier, i.e. the underlying database. This command must be in a specific format that is intrinsically linked to the database being used. As such, the functionality of the application is entirely dependent on the applications running at the second and third level of the application. This is different to *Symbian* where any application could use the method. As such, I do not think this is the same as *Symbian*.
- 17 Mr Davies then turned his attention to the *AT&T/CVON* signposts. In his first argument he made the point that the three tier programming model can be viewed as system architecture. It has several tiers and communication occurs between the tiers, and these relationships warranted description as system architecture. There is little doubt in my mind that you could easily describe a programming model as a programming architecture. However, I am not sure that this is the meaning that Lewison J intended when deriving the signposts.
- 18 In the *AT&T* decision, Lewinson J draws heavily in paragraphs 21-34 on previous case law including the EPO Technical Board of Appeal decision in *IBM*⁶ to which Mr Davies has also referred. This case related to a “method of communication

⁶ T 0006/83

between different programs and files held at different processors within a known network". In this case, the claim was allowable because it worked irrespective of the nature of the data and it was considered to relate to the architecture of the system. In this respect, the architecture of the system is how the computational components are connected and not the applications running on the computers.

- 19 I think it is clear that what Lewison J meant in his second signpost was that the architecture of a computing system is closely related to the internal components such as the operation of the processor, how the cache memory operates, or how the bus controllers and the power supplies interoperate. Each of these will continue to operate irrespective of which application runs on the computer components and I think that is the point he was making in the rider to the signpost. In this respect he was pointing out that application programs do not intrinsically have a "technical effect" but those that allow control or operation of the internal aspects of the computer may well do.
- 20 How does this affect the current application? In my view, the application is for a computer program application and not part of the architecture of the system - there is no internal control of the architectural components of the system and there is no interaction beyond that which one would expect of a computer program running on a computer. This cannot therefore be considered a technical contribution.
- 21 Mr Davies then set out an argument on the basis of the fifth signpost that the application solved a perceived problem. In illustrating this he drew my attention to the problem that current three tier systems experience with performance and scalability. A problem exists in that each tier can be managed by individual groups of technicians who do not necessarily pass implementation details of the other tiers to each other. For example, programmers who write the business logic of the second tier may be unaware of the way the database of the third tier is implemented. Such implementation information will include, for example, field validation. In providing the solution of the application, this implementation knowledge is not necessarily required since the API will let the business logic programmers handle some of the tasks that would otherwise have been achieved by the implementation logic of the database. The overall contribution is, according to Mr Davies, an improvement in the performance and scalability of the three tier model.
- 22 This may well be the case, but I have to consider whether this is a "technical contribution". I do not believe it is. The proper, and really the only, solution to the problem he sets out is for the teams of programmers to talk with the database implementers. The solution proposed by the applicant is therefore a circumvention of this problem, and the fifth signpost suggests that the application does not possess the required technical contribution.
- 23 In summary, I have decided that the application does not share the same technical features of *Symbian* nor does it pass the signposts of *AT&T/CVON*. As a consequence, I cannot identify a technical contribution and the application is considered to be a computer program as such.

Conclusion

- 24 I find that the invention is excluded under section 1(2) because it relates to a computer program as such. I have carefully reviewed the specification and do not see any possible saving amendment. I therefore refuse the application under section 18(3). As no search has been conducted, the applicant is entitled to a full refund of the fee paid despite the considerable effort invested by the examiner in dealing with the application.

Appeal

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

H Jones

Deputy Director acting for the Comptroller

Annex

Claims as filed on 25th May 2010

Claim 1

A method for implementing a persistence application programming interface (API) comprising:

Receiving a request at the API to execute a command at a persistence tier which provides data storage for an n-tier distributed architecture;

In response to the request, determining an entity type for an entity affected by the command;

Identifying a function at a middle tier of the n-tier distributed architecture that is associated with the entity type and the command, wherein the middle tier includes the business logic;

Sending an instruction to the middle tier to execute the function; and

Upon receiving a confirmation at the API that the function executed, executing the command at the persistence tier.

Claim 12

An apparatus configured to implement a persistence application programming interface (API) comprising:

A receiving mechanism to receive request at the API to execute a command at a persistence tier which provides data storage for an n-tier distributed architecture;

A determination mechanism configured to determine an entity type for an entity affected by the command in response to the request;

An identification mechanism configured to identify a function at a middle tier of the n-tier distributed architecture that is associated with the entity type and the command, wherein the middle tier includes the business logic;

A sending mechanism configured to send an instruction to the middle tier to execute the function; and

An execution mechanism configured to execute the command at the persistence tier upon receiving a confirmation at the API that the function executed.