NOTE: This disposition is nonprecedential.

United States Court of Appeals for the Federal Circuit

2008-1544
(Reexamination No. 90/006,621)

IN RE REIFFIN FAMILY TRUST

Martin G. Reiffin, of Danville, California, argued for appellant. Of counsel on the brief was Joseph L. Spiegel, Spiegel, Brown, Fichera & Cote, of Poughkeepsie, New York.

Benjamin D.M. Wood, Associate Solicitor, Office of the Solicitor, United States Patent and Trademark Office, of Alexandria, Virginia, argued for the Director of the United States Patent and Trademark Office. With him on the brief were Raymond T. Chen, Solicitor, and Thomas W. Krause, Associate Solicitor.

Appealed from: United States Patent and Trademark Office
Board of Patent Appeals and Interferences

NOTE: This disposition is nonprecedential.

United States Court of Appeals for the Federal Circuit

2008-1544
(Serial No. 90/006,621)

IN RE REIFFIN FAMILY TRUST

Appeal from the United States Patent and Trademark Office,
Board of Patent Appeals and Interferences.

DECIDED: July 27, 2009

Before MAYER, BRYSON, and PROST, Circuit Judges.

BRYSON, Circuit Judge.

The Reiffin Family Trust seeks review of a decision of the United States Patent and Trademark Office's Board of Patent Appeals and Interferences rejecting, on reexamination, all the claims of an issued patent. We affirm.

I

In 1982, Martin G. Reiffin filed a patent application that disclosed an invention for "providing real-time compilation of a high-level language source program concurrently as the program is being entered or edited at the console by the programmer." The application explained that "humans write programs in a programming language [known as 'source code'] and computers execute only machine language [known as 'object code']." The application then described a problem with the prior art in which a computer programmer is required to compile the program's source code between editing sessions, during which "the programmer is compelled to wait for completion of the loading or execution steps."

To solve that problem, Mr. Reiffin described a way for the program to be compiled "in real-time as the source code is entered or edited by the programmer." In the disclosed system, the compiler function would be interrupted in favor of the editor function whenever the programmer struck a key on the keyboard. The interrupt would cause control of the computer's central processing unit ("CPU") to pass from the compiler to the editor. Mr. Reiffin explained that the interrupt "may be activated by a timer or clock instead of by a keyboard." In either case, he explained, the CPU would execute the compiler until it was interrupted, either by the programmer striking a key on the keyboard or by a timer indicating that a preset period had expired. When either of those events occurred, an interrupt routine would save the compiler information, and the editor routine would be invoked. The editor would then perform some function, such as putting the keyed letter onto a computer display screen. Control of the CPU would thereafter be returned to the compiler.

In 1990 and 1994, Mr. Reiffin submitted two continuation applications that ultimately became, respectively, U.S. Patent Nos. 5,694,603 and 5,694,604 ("the '603 and '604 patents"). The '603 patent is directed to "a novel computer memory product including program software interrupt-driven preemptive multithreading processing so as to enable, for example, providing real-time processing of language and other alphanumeric code concurrently as the code is being entered or edited at the console." The '604 patent is directed more broadly to a "preemptive multithreading computer system with clock activated interrupt." Multithreading is generally understood as a process by which a computer runs pieces of different "threads" in alternating sequence so rapidly that it appears to be running all the threads simultaneously.

In 1998, Mr. Reiffin sued Microsoft Corporation, alleging that the operating system and the spell-checking and grammar-checking features of Microsoft's Word program infringed the '604 patent. The district court ruled that the '603 patent was invalid for failing to comply with the written description requirement of 35 U.S.C. § 112, ¶ 1, and that the '604 patent was not entitled to claim priority to a filing date of 1990 or earlier. Reiffin v. Microsoft Corp., 270 F. Supp. 2d 1132, 1135 (N.D. Cal. 2003). Mr. Reiffin then voluntarily requested reexamination of the '604 patent.

In his reexamination application, Mr. Reiffin argued that the issue of the '604 patent's priority date raised a "substantial new question of patentability." Before the PTO, Mr. Reiffin asserted that he invented multithreading in 1982, as disclosed in the application he filed in that year. Because every claim in the '604 patent contains the term "multithreading," the question whether the '604 patent is entitled to a 1982 priority date turns on whether Mr. Reiffin is correct that his 1982 application teaches "multithreading." The examiner found that the 1982 application did not describe multithreading. The examiner therefore concluded that the '604 patent's priority date was in 1994, when Mr. Reiffin filed the continuation application that led to that patent. Accordingly, the examiner rejected many of the claims of the '604 patent in light of prior art references that postdated the 1982 application but predated the 1994 application.

On appeal, the Board of Patent Appeals and Interferences agreed with the examiner that Mr. Reiffin did not disclose multithreading in 1982. In so finding, the

Board relied on the definition of multithreading in the ’604 patent specification, which states:

> The term “multithreading” is used in this specification in its ordinary generally understood sense to mean the concurrent asynchronous preemptive time-sliced execution of a plurality of threads of instructions located within the same software program, controlled by a clock or timer which periodically activates the interrupt operation of the central processor. That is, each interrupt preempts an executing thread after the thread has executed at most for a brief timeslice during which the thread may have performed only a portion of its task. Control of the processor is thereby taken away from the preempted thread, and control then passes to an interrupt service routine which then passes control to another thread to invoke the latter for execution during the next timeslice. Control is thereafter returned to the preempted thread to enable the latter to resume execution at the point where it was previously interrupted. The term “multithreading” in each claim is to be understood as defined by the respective limitations reeked [sic: recited] in that particular claim.
>
> The operation termed “multithreading” provides that control of the processor is thus transferred repeatedly back and forth between the threads so rapidly that the threads are run substantially simultaneously. The threads may thus execute incrementally and piecewise with their successive task portions executed alternately in a mutually interleaved relation and with each thread executed during its respective series of spaced timeslices interleaved with the timeslices of at least one other thread.

’604 patent, col. 1, ll. 27-53. Based on that definition, the Board determined that in a multithreading system each thread must be interruptible. While concluding that the “compiler” Mr. Reiffin had disclosed in his 1982 application was interruptible, the Board found that the “editor” was not. The Board also concluded that the editor and compiler that Mr. Reiffin disclosed in 1982 do not satisfy the definition of multithreading because they do not execute concurrently and are not located within the same software program.

Based on its determination that Mr. Reiffin had not disclosed multithreading, the Board considered whether the patent’s claims were valid over art that predated 1994. The Board held most of the claims invalid as anticipated or obvious in light of the OS/2 operating system as described by Jeffrey Krantz et al., OS/2 Features, Functions and Applications (1988) ("Krantz"). The Board also determined that some claims were invalid for failing to comply with the written description requirement of 35 U.S.C. § 112, ¶ 1. Mr. Reiffin appealed.[1]

II

Mr. Reiffin's first contention is that the Board's decision that the '604 patent should not be afforded a 1982 priority date is invalid because that determination was part of a reexamination proceeding that was not based on a "substantial new question of patentability," as required by 35 U.S.C. §§ 303-304. That argument is unavailing. Mr. Reiffin voluntarily initiated the reexamination and asked the Board to find that "all of the claims of the subject patent are entitled to the patentee's 1982 filing date of the parent application." The Board granted the request to reexamine the patent. Although Mr. Reiffin is dissatisfied with the results of the proceeding, he requested the reexamination and cannot now be heard to argue that the Board should not have granted his request in the first place.

[1] Although Mr. Reiffin recently assigned the patent at issue to the Reiffin Family Trust, Mr. Reiffin was the owner of the patent throughout most of the period relevant to this case. For simplicity, we refer to the appellant as Mr. Reiffin.

Mr. Reiffin has cancelled and disclaimed 53 of the claims of the '604 patent, leaving 30 claims at issue. The Board affirmed the examiner's rejection of remaining claims 4, 10-14, 16, 22, 24, 26, 29, 34-36, 38, 69-72, 75, and 80-83 as anticipated by Krantz. It affirmed the rejection of remaining claims 41, 42, and 51 for obviousness in light of Krantz and for failing to satisfy the statutory written description requirement. The Board affirmed the rejections of claims 39, 73, and 76 solely on the ground that they failed to satisfy the written description requirement.

III

Mr. Reiffin acknowledges that in his 1982 application the "editor always preempts the compiler but the compiler never preempts the editor." He contends, however, that multithreading does not require all threads to be interruptible. Instead, he argues that the combination of an interruptible compiler with an uninterruptible editor can properly be described as "multithreading" as that term is defined in the specification of the '604 patent, and that his 1982 application therefore disclosed multithreading sufficiently to provide a basis for assigning a 1982 priority date to that patent.

According to the definition in the '604 patent, multithreading involves "preemptive time-sliced execution of a plurality of threads." '604 patent, col. 1, ll. 28-29. The most natural reading of that phrase is that each of the threads in the plurality of threads is preempted, which in this context means that each thread is interrupted. That reading is supported by the definition of multithreading in the specification, which explains that the threads "execute incrementally and piecewise with their successive task portions executed alternately in a mutually interleaved relation and with each thread executed during its respective series of spaced timeslices interleaved with the timeslices of at least one other thread." Id., col. 1, ll. 48-53. That narrative indicates that each thread is alternately interrupted after the passage of a set period of time, or timeslice. Thus, all threads that are part of the multithreading operation must be interruptible by whatever controls the duration of the timeslice, as the Board concluded.

That interpretation is further supported by the portion of the definition stating that whenever an interrupt operation is begun, the interrupt preempts the executing thread. '604 patent, col. 1, ll. 33-34 ("each interrupt preempts an executing thread"). Because the interrupt operation is controlled by a clock or timer, whichever thread happens to be running when the interrupt is initiated must necessarily be interruptible. Id., col. 1, ll. 31-33. The definition also explains that the length of a timeslice, rather than the length of time necessary for a thread to work to completion, dictates which thread is operating at any given time. That is because a thread is interrupted after it "has executed at most for a brief timeslice during which the thread may have performed only a portion of its task," and the subsequent thread is then invoked "for execution during the next timeslice." Id. col. 1, ll. 34-36, 39-40. The next phrase, "Control is thereafter returned to the preempted thread," makes clear that the subsequent thread runs only for the duration of that timeslice, as the term "thereafter" clearly refers to a time after the timeslice has ended.

Read as a whole, the specification's definition of multithreading makes clear that whenever a thread's operation would extend beyond a particular timeslice, the thread must be interrupted in favor of the next thread. Mr. Reiffin asserts that this aspect of the definition is inapplicable to his 1982 disclosure because his editor would never operate long enough for that functionality to come into play. But even if it were the case that the editor Mr. Reiffin disclosed would operate quickly enough that it would ordinarily complete its function within a single timeslice, the definition of multithreading disclosed in the '604 patent nonetheless requires any thread to be interrupted should its operation extend beyond a single timeslice. Thus, irrespective of how the invention that was the subject of Mr. Reiffin's 1982 disclosure would have functioned in practice, the '604 patent describes a multithreading system in which a plurality of threads are interruptible, and the 1982 application does not disclose a plurality of interruptible threads.

Mr. Reiffin further argues that multithreading does not require that every thread be interruptible because editors and other interactive threads in the software field are generally not interruptible. He asserts that "foreground threads are never preempted in favor of background threads." Mr. Reiffin contends that an editor is an interactive thread that involves human interaction with a machine, and as such it would never be designed to be interrupted by a background thread.

In making that argument, Mr. Reiffin is addressing the wrong question. The question is not whether an editor, or any other interactive thread, is ever interrupted or is even interruptible. Even if Mr. Reiffin is correct that editors are never interrupted, and that programming them to be the dominant thread is a common practice, the question remains whether the combination of an interruptible thread with an uninterruptible editor can properly be described as "multithreading" as that term is defined in the '604 specification. According to that definition, systems involving an uninterrupted editor and another interruptible thread would not qualify as "multithreading" systems under that definition, even though they may be prevalent in the field.

Mr. Reiffin counters by arguing that multithreading requires only a single thread to be interruptible. Relying on the fact that the phrase "an executing thread" within the definition is singular, Mr. Reiffin contends that the phrase, as applied to his 1982 system, refers only to the compiler thread. He then asserts that each reference to an interrupt within the definition refers to an interruption of the compiler in favor of the editor. Under that reading, the phrase "each interrupt preempts an executing thread after the thread has executed at most for a brief timeslice during which the thread may have performed only a portion of its task" indicates only that whenever the interrupt is activated, the single interruptible thread (the compiler) is preempted, after which the editor begins its operation. According to Mr. Reiffin, the fact that the editor returns control of the CPU to the compiler when it has completed performing its operation, rather than at the end of a fixed timeslice, does not keep the editor/compiler from satisfying the definition of multithreading; it simply means that the editor's timeslice is of unpredictable length.

The problem with that interpretation is that it ignores the definition's opening sentence, which defines multithreading as involving "preemptive time-sliced execution of a plurality of threads." In that phrase the "preemptive" aspect of the system's execution refers to each of the plurality of threads, not to just a single thread, and the process of preemption of each thread is "controlled by a clock or timer which periodically activates the interrupt operation of the central processor." The timing of the termination of each thread is thus controlled by the clock, not by the state of completion of a particular thread's function. Because we agree with the Board that the definition of multithreading requires each thread to be interruptible, we need not consider the additional grounds on which the Board concluded that the definition of multithreading was not disclosed by Mr. Reiffin's 1982 application, namely that the threads execute concurrently and are located within the same software program.[2]

[2] The Director cites our nonprecedential decision in In re Reiffin, 199 Fed. Appx. 965 (Fed. Cir. 2006), in which we upheld the PTO's rejection of a related application of Mr. Reiffin's. Although the court in that case held that Mr. Reiffin's description of the editor/compiler programs did not constitute a disclosure of "concurrent execution" of "multiple threads," the Director has not argued that our 2006 opinion collaterally estops Mr. Reiffin from contesting the Board's ruling on those issues in this case, and we do not reach that question.

IV

Based on its determination that Mr. Reiffin failed to disclose multithreading in the 1982 application, the Board concluded that the ’604 patent was entitled to a priority date no earlier than 1994. The Board then affirmed the patent examiner’s conclusion that the majority of the claims are anticipated by the OS/2 operating system as described in the Krantz prior art reference from 1988. The Board concluded that Mr. Reiffin did “not contest that Krantz anticipates if it is prior art” and therefore forfeited any such argument. Mr. Reiffin asserts that he did not forfeit that argument because he made effectively the same argument in his reply brief before the Board. However, the passage to which Mr. Reiffin refers is in a section of his brief directed to his general argument that he is entitled to a priority date in 1982. He has not pointed to any statement in that portion of his brief that would alert the Board that he was contending that even if he were not entitled to the earlier priority date, Krantz nonetheless would not anticipate the patent’s claims. In fact, the passage Mr. Reiffin directs us to does not even mention the Krantz prior art reference. Accordingly, we affirm the Board’s conclusion as to the claims it found to be anticipated by Krantz.

V

The Board also held claims 41, 42, and 51 invalid for obviousness in light of Krantz. Mr. Reiffin has not challenged the Board’s rejection of claim 51 and has therefore forfeited the right to appeal the decision as to that claim. Beyond the limitations that the Board found were anticipated by the prior art, claims 41 and 42 also comprise “control procedures including cursor movement, screen scroll and line deletion." The Board concluded that a person of ordinary skill in the art would be motivated to combine those features with the text editor disclosed in Krantz.

Mr. Reiffin's argument as to why those two claims would not have been obvious is limited to a series of challenges to findings made by the Board and the examiner that do not affect the ultimate conclusion of obviousness. Mr. Reiffin argues, for example, that a prior art reference known as the "Nguyen treatise" does not enable spell-checking and that another prior art reference, known as the "Nitta patent," is not relevant to obviousness. Both the examiner and Board clearly indicated, however, that the obviousness rejections at issue in this appeal were based entirely on Krantz.

Mr. Reiffin also argues that claims 41 and 42 contain additional limitations that the Board "overlooked." However, there is no indication that he presented that argument to the Board, and thus he has not preserved it for appeal. Finally, Mr. Reiffin argues that the "Peterson publication" demonstrates that the limitations related to cursor movement, screen control, and line deletion in claims 41 and 42 would not have been obvious to one of ordinary skill in the art. The Peterson reference was published in 1980, however, and therefore does not support his assertion that those features would not have been obvious to one of ordinary skill in the art in 1994.

VI

The Board rejected another group of claims for failure to comply with the written description requirement of 35 U.S.C. § 112, ¶ 1. Of those claims, all of which are directed to the spell-checking and grammar-checking functions, only claims 39, 41, 42, 51, 73, and 76 are pressed on appeal, and of those only claims 39, 73, and 76 were rejected solely based on lack of an adequate written description. We address this issue only insofar as it applies to those claims.

In his 1982 application, Mr. Reiffin disclosed that "the compiler must also perform lexical, syntactic and semantic analyses of the source code . . . somewhat like forming a sequence of characters into English words." He added, "These analyses are very much like parsing the words of an English sentence. If the sequence of symbols violates a syntactic or semantic rule an 'error' is said to have been committed and the compiler must so inform the programmer by emitting a visible 'error message.'" While Mr. Reiffin included that material in his 1982 application and in related applications that he filed in 1985 and 1990, he omitted it from the 1994 application, and it does not appear in the '604 patent. The Board therefore rejected claims 39, 73, and 76 for failing to disclose that his invention would apply generally to lexical analysis for checking the spelling of words in a language. Mr. Reiffin asserts that he should have been permitted to amend the patent to add the material from the 1982 application during reexamination of the '604 patent.

The Board explained that where there is a continuous chain of copending applications, matter from an earlier application can be added through amendment to a continuation-in-part application, and for priority purposes that matter will be entitled to the filing date of the parent application from which the added matter was derived. See Litton Sys., Inc. v. Whirlpool Corp., 728 F.2d 1423, 1438 (Fed. Cir. 1984) (citing 35 U.S.C. § 120). The Board held, however, that it is impermissible to add material from a previous application to an issued patent during reexamination. The Board explained:

> This omitted subject matter cannot be added back into the '604 patent to provide written description support without showing that it is inherently

> present in the 1994 application, as filed. By omitting this subject matter in the ’604 patent, Patent Owner clearly indicated his intention that it was not his invention.

The Board added that “by allowing the 1994 application to issue as the ’604 patent without the natural language subject matter, Patent Owner created a break in the chain of disclosures, which cannot be cured by amendment.” With respect to the rule in Litton on which Mr. Reiffin relies, the Board stated that “Litton refers to adding subject matter to a pending application, not an issued patent as in this case. Once the patent issues, the record is fixed and new matter may not be added.”

According to Mr. Reiffin, the principles of the Litton case and 35 U.S.C. § 120 apply to reexamination proceedings, and he therefore should have been permitted to add material from the 1982, 1985, and 1990 applications to the issued ’604 patent during reexamination. The authority he cites for that proposition is section 1.550 of the Rules of Practice in Patent Cases, 37 C.F.R. § 1.550, which states that ex parte reexamination proceedings “will be conducted in accordance with §§ 1.104 through 1.116,” and 35 U.S.C. § 305, which provides that reexamination “will be conducted according to the procedures established for initial examination” under sections 132 and 133. Because nothing in those statutes or rules expressly bars amending an issued patent to add matter from a parent application, Mr. Reiffin argues that he was entitled to amend the ’604 patent in that manner.

Although there appears to be no authority directly on point with regard to this unusual issue, we find the Board’s reasoning sound. In a continuation-in-part application, an applicant is free to add matter from earlier related applications in a chain of co-pending applications in order to reap the benefit of the full scope of the inventions disclosed in the applicant's earlier disclosure. Once the patent issues, however, the scope of the patentee's freedom to modify the patent, whether through reexamination or reissue, is limited. By statute, no new matter may be introduced into an application for reissue. 35 U.S.C. § 251. And although a patentee is permitted to amend both the claims and the specification of his patent on reexamination, see 35 U.S.C. § 305; 35 C.F.R. § 1.530(d)(1), he is not allowed to do so in a manner that has the effect of enlarging the scope of the patent's claims, see Creo Prods., Inc. v. Presstek, Inc., 305 F.3d 1337, 1344 (Fed. Cir. 2002); In re Freeman, 30 F.3d 1459, 1464 (Fed. Cir. 1994) (a claim is impermissibly enlarged on reexamination "if it includes within its scope any subject matter that would not have infringed the original patent").

To be sure, Mr. Reiffin did not directly seek to broaden the scope of claims 39, 73, and 76 on reexamination. However, the amendment he sought to make to the issued patent would have had the same effect, as it would have broadened the scope of what Mr. Reiffin described as his invention, and thus would have expanded the universe of potentially infringing products.

Moreover, Mr. Reiffin has failed to offer a satisfactory explanation for why the principle underlying the prohibition against the introduction of "new matter" by way of amendment, 35 U.S.C. § 132, should not apply in this situation. It is true that the statutory prohibition against introducing "new matter" in a patent application does not apply in the limited circumstance in which the application incorporates material from an earlier application in a chain of co-pending applications. But in this instance, as the Board observed, the chain has been broken by the issuance of the '604 patent.

An instructive opinion on this issue is Dart Industries, Inc. v. Banner, 636 F.2d 684 (D.C. Cir. 1980) (Markey, J., sitting by designation). In that case, a patentee's first application had disclosed a particular feature, but his continuation-in-part application did not. After the continuation-in-part application issued as a patent, the patentee filed a reissue application, seeking to rely on the earlier application's filing date for priority. The PTO rejected that effort because of the break in the chain of co-pending applications disclosing the disputed feature; it held that by referring to the disclosure in the earlier application the patentee sought to add new matter to the patent in violation of section 251.

The Dart court noted that under section 120, "no claimed subject matter is entitled to the benefit of the filing date of an earlier application unless that subject matter has been disclosed in every intervening application relied upon to establish a chain of copendency." 636 F.2d at 688 (emphasis in original). In that case, continuity was lost when the continuation-in-part application was filed without the disclosure in question. Therefore, the court concluded, when that application issued as a patent, the applicant's "failure to maintain a continuity of disclosure of [the feature in question] became uncorrectable." Id.

Although this case involves the reexamination statute, 35 U.S.C. § 305, rather than the reissue statute, id. § 251, the analysis in Dart is pertinent to the effect of a break in the chain of copending applications on later efforts to amend an issued patent. Here, as in Dart, the applicant's failure to maintain a continuity of disclosure with respect to the material first included in the 1982 application but omitted from the 1994 application meant that when the 1994 application issued as a patent, the failure to maintain continuity of disclosure of that material became uncorrectable. We therefore uphold the Board's decision with respect to the disposition of the claims raised on this appeal.