United States District Court,
N.D. California.

Andrew PICKHOLTZ, an individual residing in California, and 353 Patent LLC, a Delaware Limited Liability Company,
Plaintiffs.
v.
ALADDIN KNOWLEDGE SYSTEMS LTD., an Israeli Corporation, and Aladdin Knowledge Systems, Inc., a New York Corporation,
Defendants.

No. C 04-01979 WHA

Feb. 24, 2005.

Alex Eugene Potente, Lucas W. Huizar, Roderick M. Thompson, Farella Braun & Martel LLP, San Francisco, CA, for Plaintiffs.

Andrew Pickholtz, San Francisco, CA, pro se.

CLAIM-CONSTRUCTION ORDER

WILLIAM ALSUP**, District Judge.**

INTRODUCTION

This is the claim-construction order for United States Patent No. 4,493,353, ("the '353 patent"), entitled "Software Protection Method and Apparatus." The following claim terms are construed herein: (1) "computer software," (2) "execution of," (3) "selected data locations" and (4) "external memory device." A technology tutorial, a full round of briefing, and a *Markman* hearing preceded this order.

STATEMENT

Plaintiff Pickholtz was the inventor and sole owner of the '353 patent until it expired on June 3, 2003. As part of a settlement agreement in a prior infringement suit against Rainbow Technologies, Inc., ("Rainbow"), the already-expired patent was assigned to Rainbow, but Pickholtz retained the right of first refusal to pursue patent infringement claims for infringement of the ' 353 patent against other past infringers. Plaintiff 353 Patent LLC is the assignee of all of Rainbow's remaining interests in the patent. Defendants Aladdin Knowledge Systems, Ltd. and Aladdin Knowledge Systems, Inc., (collectively "Aladdin"), are an Israeli corporation and its wholly-owned subsidiary incorporated in New York, respectively. Aladdin is accused of selling infringing "dongles," (*i.e.*, small devices that connect to a computer port and act as hardware keys to limit the unauthorized use of software), during the patent term.

The '353 patent is the sole patent in suit. The disclosed invention sought to improve upon methods of

preventing software piracy (*i.e.,* unauthorized software use or reproduction). The existing software protection systems at the time had various limitations. For example, some methods prevented unauthorized use, but also made it impossible for authorized users to make backup copies. Other methods were "either not sufficiently secure against cryptanalyst attack, require[d] too much space on the microprocessor chip or [we]re too slow." Still other hardware-based systems required "a separate hardware module for each software package that, of course, substantially increase[d] costs and decrease [d] versatility" ('353 patent at cols. 1:30-2:23) (provided as Potente Decl. Exh. A).

In the patented invention, proprietary software was protected with a hardware module containing a pseudorandom number generator and two authorization codes recorded on an external memory device. The two authorization codes were read prior to program execution. The first authorization code was fed into the pseudorandom number generator device, (*i.e.,* the "dongle"), that used a particular algorithm "unique to the authorized data processing system." The resulting product was then compared with the second code. If there was a match, then an execute enable signal was generated to enable the software to be executed (id. at col. 2:26-42).

ANALYSIS

This order construes four disputed terms in the patent: (1) "computer software," (2) "execution of," (3) "selected data locations" and (4) "external memory device." The '353 patent has only two claims, which are reproduced below. The disputed phrases are italicized.

1. A software protection apparatus using first and second authorization codes and a pseudorandom number, said software protection apparatus for use with a computer, comprising:

an *external memory device* having *computer software* and a first authorization code and a second authorization code at *selected data locations,* wherein said second authorization code is part of a pseudorandom sequence;

means for reading said *external memory device,* said reading means located in the computer;

pseudorandom number generator device located in the computer and coupled to said reading means, for generating a pseudorandom number in response to said reading means reading said first authorization code from said *external memory device,* said first authorization code being read prior to *execution of* said *computer software,* said pseudorandom number generator device including a sealed casing, thereby preventing identification of the pseudorandom number generator algorthim [sic];

processing means located in the computer and coupled to said reading means and said pseudorandom number generator device, for comparing the pseudorandom number generated by said psudorandom number generator device with the second authorization code read from *selected data locations* in said *external memory device,* said processing means generating an enable signal in response to a positive comparison of the pseudorandom number with the second authorization code for enabling *execution of* the *computer software* stored in said *external memory device*.

2. The software protection apparatus in claim 1 wherein said *external memory device* includes a floppy disc ('353 patent at col. 6:2-34).

During the prior litigation against Rainbow, some of the claim terms were construed by the Federal Circuit. Pickholtz v. Rainbow Techs., Inc., 284 F.3d 1365, 1373-74 (Fed.Cir.2002) (defining "computer" and "located in the computer") (provided as Potente Decl. Exh. E). Those particular claim terms, however, have not been raised by the parties to the present action. Judge Charles Breyer of this district had also construed six additional terms, including "computer software" and "external memory device," which are in dispute here. *Pickholtz v. Rainbow Techs., Inc.,* 2000 U.S. Dist. LEXIS 21945 (N.D. Cal., April 28, 2000) (provided as Potente Decl. Exh. C). His analyses of these phrases are discussed below.

1. "computer software"

Plaintiffs argue that "computer software" means "a set of instructions for execution on a computer, and may include associated data." Plaintiffs further contend that "computer software" may exist as a subset of a larger computer program or other file structure containing additional instructions and/or data. Defendants propose the construction "a set of instructions for execution by a computer, that does not exist as a subset of a larger computer program or other file structure containing additional instructions and/or data, and that does not include associated data." The additional limitations proposed by defendants are rejected. This order holds that "computer software" means "a set of instructions for execution on a computer, which may include associated documentation or data." This set of instructions must be complete, meaning that it performs a desired task when executed, but may also exist as a subset of a larger computer program or other file structure that performs more than one task.

Plaintiffs contend that Judge Breyer was correct in construing this phrase to mean "a set of instructions for execution on a computer, and may include associated data." In the prior litigation, however, whether computer software may include associated data was not disputed. Consequently, that issue was not specifically addressed by Judge Breyer. *Pickholtz,* 2000 U.S. Dist. LEXIS 21945 at *13-14. Thus, this order may not simply adopt that construction without independently determining the meaning of the phrase.

A long line of decisions holds that intrinsic evidence is the primary source for determining claim meaning. Vitronics Corp. v. Conceptronic, Inc., 90 F.3d 1576, 1582 (Fed.Cir.1996). Under this approach, extrinsic evidence should not be consulted, although dictionaries may be consulted, "so long as the dictionary definition does not contradict any definition found in or ascertained by a reading of the patent documents. Id. at 1584 n. 6. An alternative approach suggests that the intrinsic record should only be consulted after the ordinary and customary meaning of claim terms to persons skilled in the art, which is presumed to govern, is determined. Texas Digital Sys. v. Telegenix, Inc., 308 F.3d 1193, 1204 (Fed.Cir.2002). The presumption may be rebutted if the patentee "has clearly set forth an explicit definition of the term different from its ordinary meaning" or "has disavowed or disclaimed scope of coverage, by using words or expressions of manifest exclusion or restriction." *Ibid*. Otherwise, the ordinary meaning should be adopted. Using either approach, reliance on dictionary definitions is appropriate if the Court determines that the disputed phrase retains its plain meaning.

Here, nothing in the patent itself or in its prosecution history indicates that the inventor intended to deviate from the plain and ordinary meaning of the phrase "computer software." The intrinsic evidence is silent as to the principal points of disagreement, namely (1) whether "computer software" may be a subset or portion of a larger program or file and (2) whether associated data may be included. As to the first issue, Judge Breyer previously held that "[i]t is therefore possible for the protected "computer software" of Claim 1 to exist as a subset of a larger computer program or other file structure that contains additional instructions and/or data." Pickholtz v. Rainbow Techs., Inc., 260 F.Supp.2d 980, 986 (N.D.Cal.2003) (provided as Potente Decl. Exh.

F). The Court agrees.

Aladdin's primary argument revolves around the contention that "software" and "program" were used interchangeably in the patent and its prosecution history. Defendants argue that a "program" must be a *complete* set of instructions, that when executed on a computer will perform a required task. Even accepting this premise, however, it only sets the *lower* boundary of what "software" is. It would not preclude such "software" from existing as a subset of a larger file structure. In other words, a set of instructions (with or without associated data) would not constitute "software" if it was incomplete, such that when executed, no task was performed. But, there is no upper limit on how many complete sets of instructions "computer software" may contain, or how many different tasks it may perform.

A complex software program may consist of many sub-programs, each of which could theoretically stand alone as a program of its own. A spell-checker, for example, would be "computer software" that normally exists as a subset of a larger "word-processor" application. In describing an alternative embodiment of the invention wherein the pseudorandom number generator is implemented using software, the patent itself acknowledges that sub-programs may exist within a larger software program. The specification indicates that "it is possible to render the algorithm reasonably inaccessible by essentially 'burying' the routine within the software; the software routines which implement the steps in FIG. 2 [a flow chart explaining the steps of the invention] can also be 'buried' with the software" ('353 patent at col. 5:30-34).

At oral argument, Aladdin argued that the patentee had disclaimed the alternative embodiment described above by canceling application claim 9, which had been copied from claim 2 of United States Patent No. 4,446,519 ("Thomas," provided as Potente Decl. Exh. G at PIC 000148-153). The Court is not persuaded that the cancellation of a claim copied during an interference proceeding qualifies as a clear disclaimer of scope, particularly as the examiner later found application claims 9 and 10 were "patentable over Thomas" (id. at PIC 000174).FN*

FN* There was apparently a numbering discrepancy in the file history, wherein application claims 7 and 8 (Potente Decl. Exh. G at PIC 000154-55) were re-numbered 9 and 10, while application claim 9 (id. at PIC 000163-64) was re-numbered 11. The re-numbered claims 9 and 10 were ultimately allowed, with the examiner's suggested amendments, and issued as Claims 1 and 2 of the '353 patent (id. at PIC 000175-76).
With regard to the second point of dispute, the parties have proffered numerous dictionary definitions, all of which suggest that software *may* include the data processed by programs (Potente Decl. Exh. H at PIC 000545, Exh. K at PIC 000410, Exh. L at PIC 000396, Exh. M at 000431, Qualey Decl. Exh. 11 at AKS 2465-66, Exh. 12 at AKS 2472). Significantly, while some of these are silent on the issue, none of the proffered definitions *preclude* software from including associated data.

Defendants' argument that associated data may not be included as part of "computer software" is misguided. The opposition brief focuses on the fact that the authorization codes are examples of "associated data," which they allege is separate from "computer software." From this assumption that the protected "computer software" may not include the authorization codes, defendants draw the conclusion that "computer software," as it is used in the '353 patent, may not include *any* associated data (Opp.15). Even assuming *arguendo* that defendants' assumption is correct, the conclusion does not follow. Software programmers have long and routinely included data to inform the code of any number of variable values. It would be a rare program that completely lacked associated data, even if some of its sub-programs did not have any. At most, defendants' argument merely supports the finding that a definition of "computer software" which *required* the inclusion of all associated data would be improper.

In summary, "computer software" means "a set of instructions for execution on a computer, which may include associated documentation or data." This set of instructions may also exist as a subset of a larger computer program or other file structure that performs more than one task. Thus, while Aladdin argues that the '353 patent requires the reading of the first authorization code and the second authorization code before execution of *any* computer software, (*i.e.,* any lines of code in the overall program), this order finds that the authorization codes need only be read prior to the execution of the *protected* computer software, which may be a particular sub-program within a larger computer program.

2. "execution of"

Plaintiffs proffer the claim construction "performing, running, or carrying out an instruction by a computer." Aladdin argues that this phrase means "loading into main memory, and performing, running, or carrying out at least one instruction of a set of instructions by a computer." This order adopts plaintiffs' construction that "execution of" means "performing, running, or carrying out one or more instructions by a computer."

As with the first phrase, the intrinsic evidence is silent on whether "execution of" first requires "loading into main memory" the instruction before it is performed. The portions of the specification cited by defendants do not support their argument. The quoted sentences merely describe the path in Figure 1 by which instructions *could* be loaded onto the main memory, namely from a disc **18** to main memory **14** over channel **20** ('353 patent at col. 3:1-3, 29-32). Nowhere is it implied that this path *must* be taken whenever "an execute enable signal is generated to enable the software to be executed in the system" (id. at col. 1:40-42).

Similarly, the prosecution history does not support defendants' proffered construction. Aladdin argues that the '353 patent was distinguished from Thomas, which described a computer program that had to be "read into a working memory of the computer to generate coded interrogation signals" (Potente Decl. Exh. G at PIC 000157). Aladdin also points out that the examiner requested an amendment to insert the words "said first authorization code being read prior to execution of said computer software" (id. at PIC 000176). From this, Aladdin deduces that "execution of" requires "loading into main memory" (Opp.22). It is unclear to the Court how Aladdin reached this conclusion when the patentee distinguished Thomas on a variety of grounds, none of which related to whether the first authorization code is read before *loading* the protected software into main memory. Indeed, the Federal Circuit noted that Pickholtz had "argued that Thomas lacked (1) a PRN generator device 'located in the computer;' (2) a sealed casing on the PRN generator device; and (3) a second authorization code stored on an external memory." Pickholtz, 284 F.3d at 1369.

Moreover, the extrinsic evidence cited by defendants only further reinforces that "loading" and "executing" are separate steps. Aladdin quotes a website stating that "[d]ata from external memory will be transferred to the main memory before the CPU can operate on it" and "[t]o process data that resides in external memory, the CPU must first transfer it to main memory." First, the Court notes that citation of this evidence regarding *data,* in support of an argument regarding execution of *software,* appears to be inconsistent with defendants' position above that software may not include data. Second, that loading precedes execution does not necessarily mean that "execution of" encompasses the step of "loading."

For the reasons stated above, this order holds that "execution of" means "performing, running, or carrying out one or more instructions by a computer."

3. "selected data locations"

Plaintiffs argue that this phrase should be construed to mean "designated areas in a memory device." Defendants contend that the proper construction would be "*separate* designated data storage areas in a memory device." This order rejects the additional limitation proposed by defendants and holds that "selected data locations" means "designated areas in a memory device."

Initially, the Court notes that it is unclear from the claim language exactly what is required to be at selected data locations. Claim 1 reads, for example, "an external memory device having computer software and a first authorization code and a second authorization code at selected data locations" ('353 patent at col. 6:6-8). There is some ambiguity as to whether (1)(a) the computer software, (b) the first and (c) the second authorization codes or (2)(a) the software and (b) the two authorization codes or (3) only (a) the first and (b) the second authorization codes must be at selected data locations. Of course, the distinction between the three scenarios only matters if the word "selected" requires these designated data locations to be "separate" from each other, which will be discussed in further detail below.

Defendants focus on scenario (2) described above, arguing that the use of the plural word "locations" implies that the software must be at a different location from the authorization codes. Plaintiffs' response, that the first and second authorization codes need not be separate from each other, assumes scenario (3) is applicable. Where there is genuine ambiguity, the Court must resolve it against the drafter of the patent and adopt the more restrictive meaning. Athletic Alternatives v. Prince Mfg., Inc., 73 F.3d 1573, 1581 (Fed.Cir.1996). Accordingly, this order disagrees with both parties and finds that scenario (1) governs. Thus, the computer software, the first authorization code and the second authorization code must all be at selected data locations.

As for whether the designated data locations must be "separate," this order finds insufficient evidence in the intrinsic evidence to compel a deviation from the ordinary meaning of "selected," which the parties agree is synonymous with "designated." With regard to the patent itself, the words "separate" or "distinct" do not appear in the claim language or in the specification in the context of describing where data is located. Defendants point to the following language in the specification as evidence of separate storage on the external memory device: "stored on disc **18** together with the proprietary software are a first authorization code and a second authorization code" ('353 patent at col. 3:44-46). Yet, while this sentence appears to indicate that the software and the two authorization codes must both be stored on the same disc, there is no suggestion that they must be stored at separate data locations thereon.

Defendants' interpretation of the prosecution history is equally unsupported. It is true that the original patent application claims used the phrase "storing first and second authorization codes together with the sequence of program instructions on a storage medium" (Potente Decl. Exh. G at PIC 000014). Aladdin argues that the patentee was forced to cancel these claims because the examiner noted that the prior art- *i.e.,* United States Patent Nos. 3,609,697 and 3,996,449 (Blevins and Attanasio, provided at Potente Decl. Exh. G PIC 000024-30 and PIC 000082-90, respectively) as well as Thomas-already disclosed the use of stored security codes *within* the protected software (Opp.18-19). This is incorrect.

The patentee distinguished the prior art by arguing that the invention required *both* a first and a second authorization code stored on an external memory device, whereas the prior art lacked either the *first* or the *second* code on the external memory device (*see e.g.,* Potente Decl. Exh. G at PIC 000131, 000157-58). The inventor did *not* argue that the invention was patentable over the prior art because the authorization

codes were stored in locations separate from, as opposed to within, the protected software.

In summary, the portions of the prosecution history cited by defendants are unpersuasive because the prior art was distinguished on other grounds. Although reading the first authorization code and reading the second authorization code are described as separate steps, it does not necessarily mean that the two codes are read from separate locations. In addition, while this order recognizes that the first and second authorization codes are presumably not stored at precisely the *same* location unless they are identical codes, the ' 353 patent does not require that the two codes be stored at non-overlapping locations. Likewise, as discussed above, none of the intrinsic evidence suggested a clear disavowal of claim scope to exclude systems where the authorization codes are stored within the protected software.

For these reasons, "selected data locations" means "designated areas in a memory device." Accordingly, the computer software, the first authorization code and the second authorization code must all be at designated areas in a memory device, which are not required to be separate and distinct from each other.

4. "external memory device"

The parties agree that "external memory device" should mean "storage that is accessible by a computer through input/output channels or buses, including hard drives, floppy disk drives, CD drives, DVD drives, and flash memory sticks." Even where a definition is stipulated, however, "the trial judge has an independent obligation to determine the meaning of the claims, notwithstanding the views asserted by the adversary parties." (Exxon Chem. Patents Inc. v. Lubrizol Corp., 64 F.3d 1553,1555 (Fed.Cir.1995) (holding that the district court erred in merely choosing between the definitions proffered by the parties).

In the previous litigation, Judge Breyer relied upon a dictionary definition and held that this phrase meant "storage that is accessible by a computer only through input-output channels." *Pickholtz,* 2000 U.S. Dist. LEXIS 21945 at *16. The parties have essentially adopted this construction, but have added particular examples.

The Court agrees with Judge Breyer's holding that this phrase retains its ordinary and customary meaning (*see* Potente Decl. Exh. H at PIC 000536). Accordingly, this order holds that "external memory device" means "storage that is accessible by a computer only through input-output channels." As the parties correctly point out, this would include hard drives, floppy disk drives, CD drives, DVD drives and flash memory sticks. The stipulated construction is rejected because the word "buses" adds nothing to the definition, as a "bus" is a type of input-output channel.

CONCLUSION

The foregoing claim-construction ruling shall govern all subsequent proceedings herein.

IT IS SO ORDERED.

N.D.Cal.,2005.
Pickholtz v. Aladdin Knowledge Systems Ltd.

Produced by Sans Paper, LLC.