

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28E-FILED on 3/21/2012

IN THE UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF CALIFORNIA
SAN JOSE DIVISION

NAZOMI COMMUNICATIONS, INC.,

Plaintiff,

v.

SAMSUNG TELECOMMUNICATIONS,
INC., et al.,

Defendants.

No. C-10-05545 RMW

ORDER DENYING MOTION FOR
SUMMARY JUDGMENT OF INVALIDITY

[Re Docket No. 144]

Defendant Kyocera Communications, Inc. ("Kyocera") moves for summary judgment that the claims of U.S. Patent No. 6,338,160 (the "'160 Patent") are invalid under 35 U.S.C. § 101. Plaintiff opposes the motion. On March 16, 2012, the court held a hearing to consider Kyocera's motion. Having considered the papers submitted by the parties and the arguments of counsel, and for the reasons set forth below, the court denies the motion.

I. BACKGROUND

The '160 Patent, titled "Constant Pool Reference Resolution Method," is assigned to plaintiff Nazomi Communications, Inc. ("Nazomi"). The abstract provides:

An implementation of Java is disclosed in which references to the constant pool are implemented by using a Data Resolution Field within the constant pool entry. The Data Resolution Field acts as an index to a jump table to jump to resolve the reference or to perform the bytecode instruction. When the reference is resolved, the

1 contents of the Data Resolution Field in the constant pool entry are modified so that
2 the next time the bytecode is run, the resolution steps need not be done.

3 As explained in the specification, Java is a programming language that is portable across different
4 platforms and operating systems. '160 Patent at 1:11-14. This portability is achieved by use of
5 bytecodes, which are instructions that are not specific to a particular processor. *Id.* at 1:30-34.
6 When a Java program is executed, a bytecode interpreter, part of the Java Virtual Machine, converts
7 the bytecodes into instructions for the processor on which it is running. *Id.* at 1:34-43. Because
8 Java is run from the Virtual Machine, it is possible to operate programs without having all references
9 to objects resolved, for example before the program is completely downloaded from the internet. *Id.*
10 at 1:45-52. However, this creates a problem in that it may be time-consuming to resolve a reference
11 each time an instruction, such as an invoke instruction, is run. *Id.* at 1:53-59.

12 The '160 Patent describes the invention of a Resolution Data Field, which is used in constant
13 pool entries. *Id.* at 2:19-21. A constant pool is a data structure. *Id.* at 7:41-43. When a bytecode
14 references a constant pool, data is obtained from the constant pool, including data from a resolution
15 data field. *Id.* at 7:63-67. The data from the resolution data field is used as an index to a jump table.
16 *Id.* at 7:67-8:1. The jump table causes the system to behave differently depending on whether the
17 reference is resolved. *Id.* at 8:1-8. If the reference is not resolved, the system checks whether the
18 object referred to is in memory and, if not, loads it into memory. *Id.* at 8:5-8. The data in the
19 constant pool is modified to indicate the location of the found or loaded reference, and the resolution
20 data field is set so as to indicate that the reference is resolved. *Id.* at 8:8-13. The next time the
21 bytecode is operated on, the system will jump to normal operation of the resolved reference. *Id.* at
22 2:33-37, 8:3-4. Thus, after the first operation of the bytecode, the system need not search for the
23 object in memory. *Id.* at 2:37-39.

24 The '160 Patent compares the disclosed invention to an earlier version of the Java Virtual
25 Machine that used quick variants of bytecodes. When using quick variants, each standard bytecode
26 would resolve the specified item in the constant pool and turn itself into the quick variant of itself.
27 *Id.* at 1:65-2:2. However, because the bytecode must be overwritten by the quick variant, the
28 bytecodes cannot be run from read-only memory. *Id.* at 2:10-12. The invention of the '160 Patent
does not modify the bytecode itself, and so the bytecode can be run from read-only memory. *Id.* at

1 2:40-43. The ability to run a program from read-only memory is useful in certain circumstances,
2 particularly for embedded systems. *Id.* at 2:12-14.

3 The claims of the '160 Patent are all directed to "a method of executing an instruction."

4 Claim 1 recites:

5 1. A method of executing an instruction comprising:

6 obtaining from an instruction storage location, an instruction that references a data
7 structure, the data structure storing an indication of a reference that may need
8 resolution;

9 obtaining data from the data structure including data from a resolution data field;

10 using data from resolution data field as an index to a jump table to determine whether
11 to do a resolving step; and

12 thereafter, if the data in the data resolution field indicates that the reference was not
13 resolved, resolving the reference and, thereafter, modifying the data in the data
14 structure including modifying the data in the resolution data field to indicate that the
15 reference is resolved, wherein the data in the instruction storage location is not
16 modified.

17 Claim 11, the only other independent claim, is substantially the same as Claim 1, except it refers to a
18 "constant pool entry" instead of a "data structure."

19 II. ANALYSIS

20 Kyocera moves for summary judgment that all claims of the '160 Patent are invalid under 35
21 U.S.C. § 101. Section 101 provides for the patenting of "any new and useful process, machine,
22 manufacture, or composition of matter," subject to the conditions and requirements of Title 35.
23 Supreme Court precedents provide "three specific exceptions to § 101's broad patent-eligibility
24 principles: 'laws of nature, physical phenomena, and abstract ideas.'" *Bilski v. Kappos*, 130 S. Ct.
25 3218, 3225 (2010) (quoting *Diamond v. Chakrabarty*, 447 U.S. 303, 309 (1980)).

26 Kyocera argues that the claims of the '160 Patent cover abstract ideas and are analogous to
27 the claim held invalid in *Gottschalk v. Benson*, 409 U.S. 63 (1972). In *Benson*, the applicant sought
28 to patent a method of converting numbers from binary-coded decimal (BCD) form to pure binary
form. The Court noted that "[t]he claims were not limited to any particular art or technology, to any
particular apparatus or machinery, or to any particular end use. They purported to cover any use of
the claimed method in a general-purpose digital computer of any type." *Id.* at 64. After reviewing
principles of patent-eligibility, the Court concluded that the claims were invalid because "one may

1 not patent an idea" and "in practical effect that would be the result if the formula for converting
2 BCD numerals to pure binary numerals were patented in this case." *Id.* at 71. The Court found that
3 "[t]he mathematical formula involved here has no substantial practical application except in
4 connection with a digital computer," so a patent "would wholly pre-empt the mathematical formula
5 and in practical effect would be a patent on the algorithm itself." *Id.* at 71-72.

6 Kyocera argues that the '160 Patent claims "cover the abstract steps of obtaining a piece of
7 data, using said data to determine a next step to be taken, and then potentially modifying said piece
8 of data." Mot. at 8. Like the claim in *Benson*, Kyocera argues, the claims are so abstract and
9 sweeping that they cover both known and unknown uses through existing or future-devised
10 machinery, or without any apparatus. Nazomi responds that the '160 Patent claims, when read in
11 light of the specification, are clearly directed toward execution of computer instructions and thus are
12 not abstract. Nazomi also argues that Kyocera's motion is premature because the court has yet to
13 construe the claims, and claim construction will clarify the subject matter involved.

14 The claims here are clearly more specific than Kyocera's generalized description: the claims
15 specify that the data obtained must include data from a resolution data field, that the next step to be
16 taken is a resolving step, and that the modification is to indicate that a reference is resolved. The
17 specification teaches that "[m]ost computer languages, such as C, are compiled languages. All
18 references to objects are resolved before running of the program." '160 Patent at 1:45-47. These
19 languages, unlike Java, would not present the situation of needing to resolve a reference during
20 execution. In addition, the '160 Patent distinguishes the invention from prior art that required
21 rewriting the bytecode; instead, the claims recite modifying the data in the data structure or constant
22 pool entry, including the data in the resolution data field. The claims also recite specific
23 implementation details, such as using data from the resolution data field as an index to a jump table.¹
24 Thus, it appears that the claims are directed to a specific application and implementation of general
25 principles, rather than broadly pre-empting the use of an abstract idea.

26
27
28 ¹ The specification explains that, by using a jump table, the data in the resolution data field need
only be added to a base value rather than compared to a value; in most current processors, adding is
faster than comparing. '160 Patent at 2:43-47.

1 Kyocera argues that, even adopting Nazomi's interpretation of the claims as limited to
2 executing computer instructions, the claims are invalid. Kyocera argues that, under Nazomi's
3 interpretation, the claims define a "programming technique" and neither the Federal Circuit nor the
4 Supreme Court have found a programming technique to satisfy section 101. However, the thrust of
5 both courts' decisions on subject matter eligibility make clear that section 101 is a permissive
6 provision with limited exceptions. *See, e.g., Bilski*, 130 S. Ct. at 3225. Thus, it is not sufficient to
7 show that a type of claim has never been specifically upheld; rather, Kyocera must demonstrate that
8 the claims fall within one of the exceptions to "§ 101's broad patent-eligibility principles," *id.*
9 Kyocera has not identified any categorical exclusion of programming techniques, and the claims are
10 directed to methods, which are processes within the meaning of section 101.

11 Kyocera also argues, relying on *Benson*, that simply requiring the use of a general purpose
12 computer does not impose a meaningful limitation on claim scope. However, as discussed above,
13 the court in *Benson* concluded that allowing the claim would in practical effect allow a patent on the
14 algorithm itself because the formula for converting BCD to pure binary "has no substantial practical
15 application except in connection with a digital computer." 409 U.S. at 71-72. Here, Kyocera has
16 made no showing that the '160 Patent claims involve ideas that have no substantial practical
17 application except in connection with computer instructions.

18 Moreover, Kyocera has not clearly identified the abstract idea that the '160 Patent claims
19 allegedly pre-empt. In *Benson*, the Court characterized the claimed invention as both a
20 mathematical formula and an algorithm, which it defined as "[a] procedure for solving a given type
21 of mathematical problem." The '160 Patent claims do not appear to involve either a mathematical
22 formula or solving a type of mathematical problem. In other cases that have rejected claims under
23 section 101 for claiming abstract ideas, courts have distilled the claims at issue to general concepts.
24 *See Bilski*, 130 S. Ct. at 3231 ("Claims 1 and 4 in petitioners' application explain the basic concept
25 of hedging, or protecting against risk"); *Dealertrack, Inc. v. Huber*, --- F.3d ----, 2012 WL
26 164439 at *16 (Fed. Cir. 2012) ("The claim 'explain[s] the basic concept' of processing information
27 through a clearinghouse" (quoting *Bilski*, 130 S. Ct. at 3231)). The court does not discern any
28 "basic concept" that is explained by the claims of the '160 Patent. *See also Research Corp. Techs.*,

1 *Inc. v. Microsoft Corp.*, 627 F.3d 859, 868 (Fed. Cir. 2010) (declining "to define 'abstract' beyond
2 the recognition that this disqualifying characteristic should exhibit itself so manifestly as to override
3 the broad statutory categories of eligible subject matter and the statutory context that directs primary
4 attention on the patentability criteria of the rest of the Patent Act" and "perceiv[ing] nothing abstract
5 in the subject matter of the processes claimed" for rendering a halftone image).

6 At the hearing, Kyocera likened the basic concept behind the '160 Patent to writing down the
7 location of something, so that in the future the location is readily at hand. A bare claim on that
8 concept might well be considered unpatentably abstract. However, as discussed above, the '160
9 Patent claims recite what appear to be fairly specific steps, and the specification distinguishes the
10 invention from a prior art technique that operates on the same principle of not resolving a reference
11 more than once. The invention, a resolution data field, has the advantage of not requiring
12 overwriting the bytecode and thus allowing bytecodes to be run from read-only memory. Thus, the
13 claims appear to be directed to a narrow improvement useful in specific contexts. *See Research*
14 *Corp.*, 627 F.3d at 869 (noting "inventions with specific applications or improvements to
15 technologies in the marketplace are not likely to be so abstract that they override the statutory
16 language and framework of the Patent Act").

17 This court's conclusion that the '160 Patent claims are not unpatentably abstract is reinforced
18 by the Supreme Court's most recent decision in *Mayo Collaborative Services v. Prometheus*
19 *Laboratories, Inc.*, No. 10-1150, 566 U.S. ---, 2012 WL 912952 (2012). In *Mayo*, the Court
20 recognized that "all inventions at some level embody, use, reflect, rest upon, or apply laws of nature,
21 natural phenomena, or abstract ideas." Slip op. at 2. In distinguishing between processes that are
22 patent eligible and those that are impermissibly broad, the Court focused on whether the process
23 contains additional steps that "transform[] the process" from one that pre-empts all use of a natural
24 law "into an inventive application of the formula." *See id.* at 3, 11-12; *see also id.* at 14-15
25 (discussing favorably English case in which "the claimed process included not only a law of nature
26 but also several unconventional steps . . . that confined the claims to a particular, useful application
27 of the principle"). The Court rejected the claims at issue because the claims did little more than
28 recite a law of nature and add the instruction "apply the law." *See id.* at 8-11. The Court found that

1 the same reasoning applied in *Benson*, as the claim there "did not differ significantly from a claim
2 that just said 'apply the algorithm.'" *Id.* at 16. Here, the claims of the '160 Patent do more than recite
3 an abstract idea and say "apply it." Rather, they recite specific steps that confine the claims to a
4 specific, useful application.

5 **III. ORDER**

6 For the foregoing reasons, the court denies Kyocera's motion for summary judgment.

7
8
9 DATED: March 21, 2012


10 RONALD M. WHYTE
11 United States District Judge