United States District Court
For the Northern District of California

IN THE UNITED STATES DISTRICT COURT

FOR THE NORTHERN DISTRICT OF CALIFORNIA

ORACLE AMERICA, INC., Plaintiff, v. GOOGLE INC., Defendant. _______________________________________/	No. C 10-03561 WHA **ORDER PARTIALLY GRANTING AND PARTIALLY DENYING SECOND MOTION TO STRIKE PORTIONS OF MITCHELL REPORT AND VACATING HEARING**

INTRODUCTION

In this patent and copyright infringement action involving features of Java and Android, defendant moves to strike portions of opening and reply expert reports regarding patent infringement. For the reasons stated below, the motion is **GRANTED IN PART AND DENIED IN PART.**

STATEMENT

The factual background of this action has been set forth in previous orders (*see* Dkt. Nos. 137, 230, 433). In August 2011, defendant Google Inc. sought leave to file a motion to strike portions of the opening expert report of John C. Mitchell regarding patent infringement. Google asserted that many aspects of the report were not supported by plaintiff Oracle America, Inc.'s infringement contentions as required by the Patent Local Rules (Dkt. No. 377). Over Oracle's objections, Google was allowed to file a motion limited to three points of critique (Dkt. Nos. 380, 382). The motion was granted in part and denied in part. Because Google raised meritorious critiques, Google was allowed to file the instant additional motion to strike (Dkt. No. 464). This order follows expedited briefing.

ANALYSIS

Our Patent Local Rule 3-1 requires detailed disclosure of a party's patent infringement contentions. *See* Patent L.R. 3-1. Oracle initially disclosed its infringement contentions in December 2010, and Google challenged their sufficiency two months later (Dkt. No. 79). Oracle agreed to supplement its initial infringement contentions and in fact did so twice. Oracle ultimately chose to stand on its April 2011 second supplemental disclosures, despite Google's position that they still were inadequate. Oracle was warned that if its disclosures later proved inadequate to support its infringement theories, there would be no second chance to cure the defects (Dkt. No. 110 at 10–11).

Google now asserts that the opening and reply expert reports of John C. Mitchell contain infringement theories that were not disclosed in Oracle's April 2011 infringement contentions. Google also asserts that the reply report contains new material that should have been disclosed in the opening report. Oracle objects to the attacks on the reply report, because "Google did not obtain leave to move to strike any part of the Reply Report, as this is beyond the scope of its August 30, 2011 précis and the Court's September 26, 2011 Order, which concerned only the [opening] Report" (Opp. 4–5). This objection is overruled. The Mitchell reply report had not yet been served at the time of Google's original précis request. Google's critiques of the reply report are in the same vein as its critiques of the opening report, and all of Google's critiques have been fully briefed. These issues will be resolved now, without further procedural ado.

1. FUNCTIONS *VFORK()* AND *CLONE()* ('720 PATENT).

Patent Local Rule 3-1(c) required Oracle's disclosure of infringement contentions to include a "chart identifying specifically where each limitation of each asserted claim is found within each Accused Instrumentality." Google asserts that the functions *vfork()* and *clone()* were not identified in Oracle's claim charts, but that the Mitchell opening report now relies on them as satisfying certain limitations of claims 1 and 6 of United States Patent Number 7,426,720. Google seeks to strike a portion of paragraph 606 and all of paragraphs 622–27 from the Mitchell report "because they improperly refer to and rely upon the *vfork()* and *clone()* mechanisms" (Br. 1–2).

Claim 1 of the '720 patent covers a "system for dynamic preloading of classes through memory space cloning of a master runtime system process," comprising multiple components including "a copy-on-write process cloning mechanism" whose function is described in detail in the claim. Claim 6 covers a "system according to claim 1, further comprising . . . a process cloning mechanism" whose function also is described in detail (col. 10:2–40). The "copy-on-write process cloning mechanism" of claim 1 and the "process cloning mechanism" of claim 6 are similar, but not identical.

Paragraph 606 of the Mitchell report states: "The Linux fork executed by Android provides the 'copy-on-write process cloning mechanism' in its *fork()* system call. Linux provides additional 'process cloning mechanisms' in its *vfork()* and *clone()* system calls." The paragraph then quotes at length from a source explaining these three system calls, or "functions" (Banner Exh. A at 301–06).

Google's instant motion does not challenge Mitchell's theory that the *fork()* function satisfies the "copy-on-write process cloning mechanism" limitation of claims 1 and 6. Indeed, the Mitchell report also identifies several other items as satisfying the "copy-on-write process cloning mechanism," and Google does not contend that any of those theories was not timely disclosed.

On the other hand, Mitchell's theory that the *vfork()* and *clone()* functions satisfy the "process cloning mechanism" of claim 6 is where the battle is joined. Paragraphs 622–27 of the Mitchell report focus largely on the *clone()* function as an example of a "process cloning mechanism" for claim 6 purposes (Banner Exh. A at 131–15). Oracle's operative infringement contentions did not identify the *vfork()* or *clone()* functions as satisfying the "process cloning mechanism" limitation of claim 6. This is not disputed. Oracle therefore may not now rely on those functions as satisfying that claim limitation.

Oracle states that in both its infringement contentions and in the Mitchell report, its "'720 patent infringement theory focuses on a function present in Android called *do_fork()*." According to Oracle, the report references *vfork()* and *clone()* only to explain where *do_fork()* is found and how *do_fork* is called (Opp. 1–3). True, the Mitchell report identifies the function *do_fork()* as "provid[ing] the 'process cloning mechanism' of claim 6" in one of the very paragraphs Google seeks to strike (Banner Exh. A at 314). Google does not challenge this theory in its instant motion. The report, however, also seems to identify the functions *vfork()* and *clone()* as satisfying that limitation, as described above. Such theories would be improper.

As to the functions *vfork()* and *clone()*, Google's motion is **GRANTED IN PART AND DENIED IN PART** as follows. Google has not shown that the Mitchell report identifies *vfork()* or *clone()* as satisfying the "copy-on-write process cloning mechanism" limitation of claim 1 and dependent claim 6. This order therefore does not address whether such theories were adequately disclosed in Oracle's infringement contentions. On the other hand, the Mitchell report does identify *vfork()* and *clone()* as satisfying the "process cloning mechanism" limitation of claim 6. Those theories were not disclosed in Oracle's infringement contentions. Accordingly, Mr. Mitchell may not offer direct testimony that the *vfork()* or *clone()* function itself satisfies the "process cloning mechanism" limitation of claim 6. This ruling, however, does not preclude any and all mention of those functions on direct. For example, they may be referenced to describe where and how the *do_fork* function is called if indeed Oracle contends that *do_fork* itself satisfies the "process cloning mechanism" limitation.

2. "mBS MOBILE" APPLICATION ('476 PATENT).

Again, the Patent Local Rules required disclosure of a chart identifying where each limitation of each asserted claim was found within each accused item. Google asserts that no application called "mBS Mobile" was disclosed in Oracle's operative infringement contentions, but that the Mitchell opening report now identifies such an application as providing the security functionality claimed in United States Patent Number 6,192,476. Google seeks to strike paragraphs 70 and 77 from the Mitchell report.

Paragraphs 70 and 77 are identical. They refer to the mBS Mobile application to illustrate "[s]ome of the ways that application developers could benefit from the java.security framework" (Banner Exh. A at 21–24). It is undisputed that the mBS Mobile application was not mentioned in Oracle's infringement contentions. The Mitchell report, however, does not accuse the mBS

Mobile application of satisfying any particular limitation of the asserted claim of the ’476 patent. Rather, the java.security framework is accused, and the mBS Mobile application is cited as an example of the benefit of the alleged infringement. Paragraphs 70 and 77 do not present a new theory of infringement based on the mBS Mobile application. As explained with reference to Google’s first motion to strike portions of the Mitchell report, the Patent Local Rules required Oracle to disclose its *theories* of infringement, not to identify every evidentiary *item of proof* supporting those theories.

In its reply brief, Google characterizes the discussion in paragraphs 70 and 77 as a new theory of *indirect* infringement. Google, however, does not explain how Oracle’s disclosures concerning indirect infringement are supposedly inadequate to support the challenged paragraphs. As to the mBS Mobile application, Google’s motion is **DENIED**.

3. SECURITYMANAGER AND ACCESSCONTROLLER (’476 PATENT).

Google asserts that paragraph 115 of the Mitchell reply report “should be stricken because it represents a new infringement theory that was never previously disclosed by Oracle” (Br. 4). The entirety of challenged paragraph 115 states (Banner Exh. B at 47) (internal quotations and citation omitted):

> Even if the SecurityManager is disabled, the Java security framework may still be used. For example, the static methods in AccessController are always available to be called. Even if no SecurityManager is instantiated, Android code includes AccessController.java and the static methods associated with the class may be called. Therefore, the functionality of the Java security framework is accessible via the AccessController.

Google argues that this theory is improper because it was not disclosed in Oracle’s infringement contentions and was not presented in the opening Mitchell report. According to Google, both the infringement contentions and the opening report relied on SecurityManager rather than AccessController to show infringement of the ’476 patent (Br. 3–4).

Challenged paragraph 115 is located in a portion of the reply report that addresses claim 10 as an example, stating that the other then-asserted claims “are similar” (Banner Exh. B at 46). At present, only claim 14 of the ’476 patent remains at issue (Dkt. No. 471). The instant record includes portions of the infringement contentions and opening report that address claim 10, but not claim 14. These excerpts specifically address the "determining whether said action is authorized" limitation of claim 10. Claim 14 includes a similar, but not identical, limitation. Because both sides rely on the claim 10 portions of the infringement disclosures and opening report, this order assumes for the limited purposes of the instant motion that those claim 10 arguments apply equally to claim 14.

Oracle's infringement contentions disclose AccessController along with SecurityManager with respect to the relevant limitation (Banner Exh. D at 22). Thus, the statements about AccessController in paragraph 115 do not run afoul of the disclosure rules.

Whether the content of paragraph 115 should have been in the opening report is a separate question. The parties were warned that reply reports must be limited to true rebuttal, should be very brief, and should not add new material that should have been placed in the opening report (Dkt. No. 56 at ¶ 8). The Mitchell opening report opined that SecurityManager.java was the accused element that satisfied the "determining whether said action is authorized" claim limitation (Banner Exh. A at 362). The report further explained that it did so "by calling the AccessController.checkPermission method, which in turn calls the AccessControlContext.checkPermission method to check the permissions" (*id.* at 363). The opening report did not identify any other accused element as satisfying the limitation in question, and it did not identify any other steps by which the AccessController could be called.

The statement in paragraph 115 of the reply report that the AccessController methods may be called by something other than the SecurityManager presents a new infringement theory. This statement is not true rebuttal; it is sandbagging. Oracle had the burden to set forth all of its infringement theories in the opening report. If Oracle intended to accuse something other than the SecurityManager of infringing the claims of the '476 patent by calling the AccessController methods, then the Mitchell opening report should have said so.

Google argues that this critique is analogous to the critique regarding DvmDex.h in Google's first motion to strike, and that this critique should be rejected for the same reason. Not so. There, the infringement contentions identified specific accused functions, and the Mitchell report later identified the code files in which those functions were stored. Here, the opening report identified a series of accused steps, and the reply report then sought to replace one step in the series with a different (and unspecified) accused element. Doing so would present a new infringement theory.

As to the theory of infringement by something other than the SecurityManager calling the AccessController methods, Google's motion is **GRANTED**. Paragraph 115 of the Mitchell reply report is **STRICKEN**.

4. MULTIPLE INSTRUCTIONS AND UNSUCCESSFUL SIMULATED EXECUTION ('520 PATENT).

Google asserts that the Mitchell reply report presents two new theories of infringement of United States Patent Number 6,061,520. According to Google, Oracle's infringement contentions and the Mitchell opening report both advanced the theory that Android's dx tool infringes the asserted claims of the '520 patent "where it successfully creates a single new instruction, referred to as a fill-array-data instruction, that replaces a less efficient series of Java bytecode instructions for initializing individual elements of a static array" (Br. 4). On the other hand, Google interprets the Mitchell reply report as claiming that infringement includes "instances where more than one instruction is created" and instances where simulated execution is not successful (*id.* at 5). Google moves to strike the "one or more" phrase and the last three sentences of paragraph 92 of the Mitchell reply report.

A. Creation of Multiple Instructions.

The relevant portions of Oracle's infringement contentions and the Mitchell opening report do refer to creating a single instruction, not one or more instructions (*e.g.*, Banner Exh. E at 29, Exh. A at 277). The portion of paragraph 92 that refers to creating "one or more instructions" purports to further explain Oracle's infringement theory; it does not directly respond to any argument by the opposing expert.

Oracle does not attempt to explain or justify the reference to creating "one ore more" instructions in the Mitchell reply report. Oracle cites only to Google's interrogatory responses, which are irrelevant to the question at hand (Opp. 7). The statement in paragraph 92 that the accused elements create "one or more" instructions amounts to a new infringement theory. As such, it is improper.

B. Unsuccessful Simulated Execution.

The last three sentences of paragraph 92 address "Prof. Parr's example in his Appendix A, Section II" (Banner Exh. B at 39). This example involved a test that was run on modified, *i.e.*, not standard, Android code (Peters Exh. C). As such, it is a hypothetical example. The statements addressing this hypothetical example are true rebuttal and need not be supported by any previous disclosures. In particular, the statement that the claims of the '520 patent "do not require the simulating step to succeed" expresses a broad view on the scope of the claims as applied to the hypothetical example. It does not amount to an entirely new infringement theory, because it does not accuse any particular new item of infringement. Even if Oracle chose in its infringement contentions to accuse only devices and methods in which simulated execution is successful, Oracle's expert still may rebut a hypothetical example by opining that circumstances in which simulated execution does not succeed may also infringe. The last three sentences of paragraph 92 are proper.

As to the reply report discussion of the '520 patent, Google's motion is **GRANTED IN PART AND DENIED IN PART**. The reference to creating more than one instruction is **STRICKEN** from the third sentence of paragraph 92 of the Mitchell reply report. The last three sentences of that paragraph remain intact.

CONCLUSION

For the reasons stated above, defendant's motion to strike portions of the opening and reply expert reports of John C. Mitchell regarding patent infringement is **GRANTED IN PART AND DENIED IN PART**. As to the functions *vfork()* and *clone()*, the motion is **GRANTED IN PART AND DENIED IN PART**. Mr. Mitchell may not offer direct testimony that the *vfork()* or *clone()* function itself satisfies the "process cloning mechanism" limitation of claim 6 of the '720 patent. As to the mBS Mobile application, the motion is **DENIED**. As to the theory that the '476 patent can be infringed by something other than the SecurityManager calling the AccessController methods, the motion is **GRANTED**. Paragraph 115 of the Mitchell reply report is **STRICKEN**. As to the reply report discussion of the '520 patent, Google's motion is **GRANTED IN PART AND DENIED IN PART**. The reference to creating more than one instruction is **STRICKEN** from the third sentence of paragraph 92 of the Mitchell reply report. The last three sentences of that paragraph remain intact.

Any denial of a motion to strike does not mean that the item at issue in the motion is automatically admitted in evidence — it still must be moved in evidence, subject to any *other* objections, at trial. Similarly, any grant of a motion to strike does not exclude the item at issue in the motion under any and all circumstances; a party who prevailed on a motion to strike may open the door to the stricken material.

The hearing on the instant motion previously set for October 13, 2011, is **VACATED**. The hearing on the motion for relief from a non-dispositive pretrial order of a magistrate judge, set for the same day, will go forward as planned.

IT IS SO ORDERED.

Dated: October 11, 2011.

Wm Alsup
WILLIAM ALSUP
UNITED STATES DISTRICT JUDGE