Exhibit 19

UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF NEW YORK

NETWORK-1 TECHNOLOGIES, INC., Plaintiff, vs. GOOGLE LLC and YOUTUBE, LLC, Defendants.	Case No. 14-cv-2396 Case No. 14-cv-9558

EXPERT REPORT OF MICHAEL MITZENMACHER, PH.D. REGARDING VALIDITY OF U.S. PATENT NOS. 8,010,988; 8,205,237; AND 8,904,464

code for running the search was downloaded onto a user's computer. Is that right? A. Correct. Q. And it's not -- A. Actually, you know what, I'm not sure about that. I actually wonder if we didn't distinguish, because we -- we gave them one piece of code. It's -- it's possible that that code existed on the Clango application. I -- I would have to check into that. Q. You just don't know? A. But it wasn't called directly, if that's the distinction you're trying to make. Q. Well, was there -- I -- I guess users didn't have -- I mean, for one thing, they didn't have the database, right? A. Correct. Q. And so they couldn't run the searches themselves? . . . Yeah, they could run searches only through that interface and not -- but the search code itself was not executed on the client side.").

c. As discussed above, Mr. Schrempp's emails to alpha and beta testers do not mention any search algorithms or search methodologies. AUDMAG0014893-94; AUDMAG00014930. Moreover, Audible Magic has *never* disclosed its indexing techniques to the public. *See* Wold Depo. at 212:15-213:6 ("Q. Now, what was the first time that you or anybody at Audible Magic made any public disclosure about any of the indexing techniques you use there? . . . I mean, in some ways, we've never made public disclosure. We -- except I -- I -- I mean, it depends on what you mean. I mean, obviously, the -- the product exists in a sense that it's a public disclosure. And I'm sure there have been casual mentions to customers or important suppliers or other people, but I -- I don't know."); *see also id.* at 213:8-214:14 (testifying that there may have been end-user license agreements in place for Clango testers, and that he did not recall the terms of those agreements); Schrempp Depo. at 60:2-62:2 (testimony concerning AUDMAG00014891-92); *id.* at 126:1-127:21 (testimony concerning AUDMAG00014870).

d. From my review of the Clango demo, it was not possible to discern how Clango was outputting the name of the song being played on Windows Media Player. In other words, it was impossible to tell from the demo what type of search algorithm, if any, was being used.

e. In short, without a description of the search algorithm or search methodologies used and/or access to the program's source code, a user would have no way of discerning whether or not any "identification" Clango Alpha or Clango Beta made is "based on a non-exhaustive search identifying a neighbor" that "is sublinear."

47. Even setting aside the undisclosed nature of the search algorithms used, the Darrell Report does not establish that any "identification" Clango Alpha or Clango Beta made is in fact "based on a non-exhaustive search identifying a neighbor."

a. Dr. Darrell points to the following statement in an email written by Mr. Schrempp: "Brute force won't yield the performance or accuracy needed for this." Darrell Report ¶ 129 (quoting AUDMAG01710721).

i. However, Mr. Wold testified at his deposition: "I would actually disagree with his statement. I would say, yes, that it is required for performance; but, actually, brute force tends to be the most accurate. So you would -- I would typically implement a brute force algorithm to make sure that I have the exact answer I can ever -- and the best answer I can come up with. And then to get performance -- I would not do a brute force search, but I would have the brute force search to compare." Wold Depo. at 38:4-39:11. A person of ordinary skill in the art would have understood that this is a common approach for developing and analyzing the accuracy of search algorithms.

ii. This is indeed the approach Mr. Wold followed. In an email concerning a February 2000 API (application programming interface) implementation, "[i]f you don't call MFDatabaseIndex, no index will be created and a brute force implementation of MFDatabaseLookup will be used for searching. If you do call MFDataBaseIndex, a new tree-based faster lookup will be used." AUDMAG01652077; *see also* Wold Depo. at 203:6-13. Similarly, in a December 1, 2000 email Mr. Schrempp stated that "[t]he engine runs in two modes: exhaustive search and index." AUDMAG00042894. Mr. Schrempp's December email also indicates that Audible Magic was still making changes to the search algorithm, several months after the alpha and beta tests. *See* AUDMAG00042895. The Darrell Report has therefore not established whether the July 5, 2000 Clango Alpha or the August 2000 Clango Beta used a brute force implementation, the (unspecified) tree-based implementation referenced in this email, or another search algorithm entirely. And in any event, whatever algorithm was used was not discernible to the public for either Clango Alpha or Clango Beta.

iii. In addition, it would not have been surprising to one of skill in the art that someone would opt to use a more accurate (exhaustive) search algorithm in early testing versions of an application. Alpha and beta tests are aimed at accessing a user's satisfaction with the application, and search accuracy would have been a very important feature for an application that aims to identify what song if playing on the user's computer. A skilled artisan would have also recognized that, given the small size of the reference databases used with both Clango Alpha and Clango Beta, an exhaustive search would have been a possible, if not probable, implementation. *See* AUDMAG00041833 ("For alpha we decided that 20,000 samples were all we could afford from a performance standpoint."); *see also* AUDMAG00042894-96 (Clango had about 25,000 references in December 2000, months after Clango Beta was released.). Finally, it is typical in alpha and beta testing to focus on soliciting feedback on user interface features as opposed to the search time or search algorithm, particularly where, as here, the search algorithm was not disclosed to the alpha and beta testers.

b. In addition, for Clango Alpha, Dr. Darrell states that "[f]or the July 2000 alpha version of the Clango program, Mr. Wold had begun utilizing kd-trees to structure and search the reference database of fingerprints." Darrell Report ¶ 187. The only evidence he cites in support of this notion is Mr. Wold's testimony; he cites no corroborating documents that the Clango Alpha used kd-trees to index and search the reference database. Moreover, the source code Dr. Darrell cites likewise does not establish that Clango Alpha used kd-trees. *See id.* ¶¶ 189-198. Critically, on their face, these files are dated July 26, 2000 at the earliest, several weeks after Clango Alpha was allegedly released. *See, e.g.*, AUDMAG01085773-83 (Wold Depo., Exhibit 10); *see also* AUDMAG01085789-94 (Wold Depo., Exhibit 11). Indeed, Mr. Wold stated that "there is an SCCS [source code control system] on Exhibits 10 and 11 that is July 26." Wold Depo. at 271:5-6. At his deposition, he was at best unsure whether Clango Alpha used kd-trees and also confirmed that it is impossible to tell from the files alone whether or not the code in those files was implemented in *any version* of Clango:

> Q But you can't say with certainty that these particular files, Exhibits 10 and 11, were in production code for any version of Clango, correct?
> . . .
> THE WITNESS: No. What I'm saying is – I mean, this doesn't prove it, this -- the existence of this document doesn't provide it. But I know that the k-d tree code was in the version of Clango that went out. And I -- I'm losing dates again. But I had forgotten if that was -- hold on. I can check here. Definitely by the August 9th release, it would have been.

Wold Depo. at 271:12-2.

c. Moreover, Dr. Darrell relies exclusively on dates from the SCCS (Source Code Control System) files in order to establish dates for these two code files. Setting aside that, as Mr. Wold testified, the existence of a code file does not demonstrate that that code file or any portion of the code contained in the file was included in any certain product, there are also problems with such exclusive reliance on *dates* from a SCCS to establish the file merely existed on that date. While these dates provide some information regarding the provenance of the files, they should not be relied upon as a complete confirmation of the dates, for multiple reasons.

i. A brief description of what an SCCS is is informative for this discussion. An SCCS is system that can be used to track changes in code projects over time. A code file has a corresponding SCCS file that tracks changes made to the file. Generally, a user "checks out" a file to make changes (which do not appear in the main, repository-held version of the code), and then "checks in" the file when ready, at which point the SCCS computes and stores the changes, or delta, since the last version, and creates a corresponding new version in the main repository. The SCCS files are stored as text files. There are therefore multiple reasons why SCCS files may not provide correct date information.

ii. First, since they are text files, they can be changed at any time by a user with privileges to modify the file. The date can be easily changed simply by

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE