DARREN EASTMAN
21446 Oneda Court
Los Gatos, CA 95033
darren@eastmantechnologies.com
Pro Se

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

OAKLAND

DARREN EASTMAN, Plaintiff, v. APPLE, INC., Defendant.	Case No. 4:18-CV-05929-JST **FOURTH AMENDED COMPLAINT** **JURY TRIAL DEMANDED** Judge: Hon. Jon S. Tigar Courtroom: 6 Action Filed: 8/13/2018

TABLE OF CONTENTS

LEGAL STANDARD

FACTUAL BACKGROUND OF INVENTIONS

CAUSES OF ACTION

A. PATENT NONJOINDER

TABLE OF CONTENTS (continued)

TABLE OF EXHIBITS

TABLE OF AUTHORITIES

<u>CASES</u> (continued) **Page**

<u>CASES</u> (continued) **Page**

<u>CASES</u> (continued) **Page**

Mueller Brass Company v. Reading Industries, Inc.
352 F. Supp. 1357, 1372 (E.D. Pa. 1972) …………………… 7

Pannu v. Iolab Corporation
155 F.3d 1344 (Fed. Cir. 1998) …………………… 6, 159, 165

PerSeptive Biosystems Inc. v. Pharmacia Biotech, Inc.
225 F.3d 1315, 1322-23 (Fed. Cir. 2000) …………………… 5

Popescu v. Apple Inc.
H040508 Cal.App.4th (2016) …………………… 169

Price v. Symsek
988 F.2d 1187, 1195 (Fed. Cir. 1993) …………………… 6, 167

R2 Investments LDC v. Phillips
401 F.3d 638, 642 (5th Cir. 2005) …………………… 158

Roberts v. Ball, Hunt, Hart, Brown & Baerwitz
57 Cal.App.3d 104 (2nd Dist. 1976) …………………… 16

SCA Hygiene Products Aktiebolag v. First Quality Baby Products, LLC
No. 15-927 580 U.S. (2017) …………………… 164

Sewall v. Walters
21 F.3d 411, 415 (Fed. Cir.1994) …………………… 4

Shukh v. Seagate Technology, LLC
803 F. 3d 659 (Fed. Cir. 2015) …………………… 2, 7, 169, 171

St. Joseph Solutions, LLC v. Microtek Medical, Inc.
No. 1:11-CV-388 (S.D. Ohio 2011) …………………… 3

Stark v. Advanced Magnetics Inc.
119 F.3d 1551, 1556 (Fed. Cir. 1997) …………………… 159

Tate Engineering, Inc. v. United States
477 F.2d 1336, 1342 (Fed. Cir. 1973) …………………… 159

Therasense, Inc. v. Becton, Dickinson & Company
649 F.3d 1276, 1292 (Fed. Cir. 2011) (en banc) …………………… 4

Trovan, Ltd. v. Sokymat SA, Irori
299 F.3d 1292, 1301 (Fed. Cir. 2002) …………………… 5, 164

Tucker v. Naito
188 USPQ 260, 263 (1975) …………………… 164

CASES (continued) **Page**

CONSTITUTIONAL PROVISIONS

STATUTES

STATUTES (continued) **Page**

RULES

PTO REFERENCES

CERTIFICATE OF SERVICE

Pursuant to Local Rule 5.3, I hereby certify that on April 29, 2020, this FOURTH AMENDED COMPLAINT herein was electronically filed with the Clerk of the Court using the Electronic Case Filing system; which automatically generates a “Notice of Electronic Filing” to all counsel of record registered in the case.

s/ Darren Eastman
DARREN EASTMAN
Pro Se

DARREN EASTMAN
21446 Oneda Court
Los Gatos, CA 95033
DARREN@EASTMANTECHNOLOGIES.COM

COMPLAINT FOR PATENT NONJOINDER

Plaintiff Darren Eastman files this complaint for patent nonjoinder, reputational damages and a demand for jury trial against Apple, Inc. (collectively, "Defendants" or "Apple") and alleges as follows:

THE PARTIES

1. Darren Eastman's an individual based at the address on the complaint.

2. Apple Inc. is a California corporation with its principal place of business at One Apple Park Way, Cupertino, CA 95014. Apple may be served through its agent for service process CT Corporation System at 818 West Seventh Street, Suite 930, Los Angeles, CA 90017.

JURISDICTION AND VENUE

3. This action arises under 35 U.S.C. § 256. This Court has original jurisdiction over this controversy pursuant to 28 U.S.C. § 1331 and § 1338.

4. Venue is proper in this Court pursuant to 28 U.S.C. § 1391(b) and (c) and/or 28 U.S.C. § 1400(b).

5. This Court has personal jurisdiction over Apple; who regularly and continuously do business in this District. Upon information and belief, Apple maintains an office within this District (Cupertino, California). Upon information and belief, Apple's office in Cupertino is a regular and established place of business. This Court has personal jurisdiction over Apple; because minimum contacts have been established with the forum, and, the exercise of jurisdiction would not offend traditional notions of fair play and substantial justice.

INTRODUCTION

6. This case concerns multiple patent misjoinder and nonjoinder.

7. After being filed as a FAC in Superior Court (solely to add needed forms on August 22, 2018 but without changing brief content) Apple moved the case to this Court on September 27, 2018. The Court severed and remanded wrongful conversion of property and termination causes of action to the Superior Court on November 14, 2018, citing 28 U.S.C. § 1441(c)(2) and granting leave to file a SAC; to unify evidence and attach relevant patent claims.

8. Leave was granted on April 10, 2019 to file a TAC—to state the correlations between plaintiff's conception and inventions not deemed previous art against the patent claims.

9. Plaintiff is the true and original, putative inventor of the "Find my iPhone" feature: which utilizes 13 utility patents filed misjoinder and nonjoinder by Apple. This innovative feature has since been extended to computers and tablets.

10. Plaintiff also declared a novel process for redemption of virtual tickets using a mobile device in his 2006 Intellectual Property Agreement (IPA), which matches a ‘14 patent Apple was granted for their “Passbook” application, along with both a ‘495 and ‘513 patents. Plaintiff is a co-inventor and was misjoinder 3 utility patents.

11. Plaintiff only discovered being nonjoinder *in re* the ticketing patents after his wrongful termination; while researching previous nonjoinder patents—showing a demonstrated and repeated pattern of discrimination by Apple; including, but not limited to its executives and legal group. While clearly not evident herein, when there’s no apparent disagreement, “as between inventors their word is normally taken as to who are the actual inventors." *Brader v. Schaeffer* 193 USPQ 627, 631 (1976). A § 256 dispute herein exists between Apple and plaintiff for both phone-finding and electronic ticketing patents.

12. Severe reputational damage occurred to plaintiff from both his nonjoinder on sixteen patents, and, subsequent wrongful termination.

13. Apple intentionally caused plaintiff to be unable to secure jobs before and after his wrongful termination at similar companies he was qualified; with strong interest in plaintiff existing at Google and Pixar. This began from Apple’s wrongful “anti-poaching” collusion behavior. *High-Tech Employee Antitrust Litigation* (N.D. Cal. 2015) but continued after his wrongful termination.

14. This § 256 case is nearly identical to *Shukh v. Seagate Technology, LLC* 803 F. 3d 659 (Fed. Cir. 2015), with the exception Shukh was given an award for his nonjoinder patents and told they’d be filed, whereas Apple intentionally concealed patent filings from plaintiff; while still employed by them. This is despite plaintiff reaching out to executives and patent counsel seeking patent protection; both before the feature was ever officially developed, and, after his initial *reduction to practice*.

LEGAL STANDARD

A. **Federal Rule of Civil Procedure 12(b)(6)**

15. 8(a)(2) requires that a complaint contain, “a short and plain statement of the claim showing that the pleader is entitled to relief.” While a complaint need not contain detailed factual allegations, facts pleaded by a plaintiff must be, “enough to raise a right to relief above the speculative level.” *Bell Atl. Corp. v. Twombly* 550 U.S. 544, 555 (2007). Surviving a Rule 12(b)(6) motion to dismiss requires sufficient factual matter that, when accepted as true, states a claim that is plausible on its face. *Ashcroft v. Iqbal* (2009) 556 U.S. 662, 678 (2009) “A claim has

facial plausibility when the plaintiff pleads factual content that allows the court to draw the reasonable inference that the defendant is liable for the misconduct alleged." *Id*. While this standard is not a probability requirement, "where a complaint pleads facts that are merely consistent with a defendant's liability, it stops short of the line between possibility and plausibility of entitlement to relief." *Id.* (internal citation omitted). In determining whether a plaintiff has met this plausibility standard, the Court must "accept all factual allegations in the complaint as true and construe the pleadings in the light most favorable" to the plaintiff. *Knievel v. ESPN* (2005) 393 F.3d 1068, 1071 (9th Cir. 2005).

16. In cases denying motions to dismiss §256 claims, plaintiffs identified specific theories the Court could reasonably assess for plausibility. *Gerawan Farming, Inc. v. Rehrig Pac. Co.* No. 1:11-CV-01273 LJO, 2012 WL 691758, at *3 (E.D. Cal. 2012) (finding that allegations plausibly suggested that plaintiff "conceived the distinct wall construction as part of his work on the Harvest Tote and this distinct construction was incorporated in the claims of the '293 Patent"); *St. Joseph Sols., LLC v. Microtek Med., Inc.* No. 1:11-CV-388, 2011 WL 5914010, at *12 (S.D. Ohio 2011) (identifying "connected sleeves" as an element of specific claim of patent that plaintiff allegedly demonstrated to patentee). The plaintiffs distinct work on remotely communicating with a lost device and redeeming virtual tickets is not unlike the Harvest Tote.

17. Herein, plaintiff demonstrates how his work to solve the longstanding problem of reliably retrieving a lost computing device was both novel, and, addressed the problem squarely enough that permutations such as whether an honest finder or thief may find a lost device before the true owner are also solved (or, methods to prevent the device from being erased or repurposed, methods to restrict usage of the device until unlocked by the true owner, methods to see where the device had travelled over time, and, methods to provide messaging for an honest finder to contact the true owner) as identified against the patent claims. Using a cloud server to initiate and manage a computing device when declared lost by the true owner is a further example of subproblems plaintiff also solved to achieve his successful invention. *In re* the Passbook patents, plaintiff declared a novel method to redeem virtual tickets by generating a barcode on the display surface of a mobile device. This declaration was made in his pre-employment Intellectual Property Agreement (IPA) a *decade* before Apple filed a patent with matching claims.

18. Both dependent and independent claims of the sixteen patents in-question are identified using the plaintiffs demonstrated evidence of *conception* and inventorship, however, "the presumption that an independent claim should not be construed as requiring a limitation

added by a dependent claim" herein exists. *Curtiss-Wright Flow Control Corp. v. Velan, Inc*. 438 F.3d 1374, 1380 (Fed Cir. 2006).

B. Correction of Ownership and §256

19. 35 U.S.C. § 256 authorizes District Courts to issue an order directing that a patent be corrected to properly reflect inventorship, when necessary. Inventors not properly listed on a patent may present important ramifications for the assignee in enforcement. If a patent doesn't properly list all inventors, the claims can be held invalid, the patent rendered unenforceable, or, litigation related to such enforcement dismissed—based on failure to properly join all inventors. The Federal Circuit has construed §256 to, "provide a cause of action to interested parties to have the inventorship of a patent changed to reflect the true inventors of the subject matter claimed in the patent." *Fina Oil & Chem. Co. v. Ewen* 123 F.3d 1466, 1471 (Fed. Cir. 1997). Further, "inventorship is a question of law that we review without deference." *Sewall v. Walters* 21 F.3d 411, 415 (Fed. Cir.1994). "We review the underlying findings of fact for clear error." *Hess v. Advanced Cardiovascular Sys., Inc*. 106 F.3d 976, 980 (Fed.Cir.1997).

20. Here, Apple committed intentional falsehoods by submitting sixteen misjoinder patent applications, which contain plaintiff's novel and unique invention disclosures. The Passbook patents are a misrepresentation; as other legitimate inventors contributed to plaintiff's original invention, but, he was nonjoinder. This resulted from Apple's gross negligence in not honoring its own pre-employment IPA.

21. Omissions occurred in all sixteen patents, as the putative inventor was nonjoinder; especially given Apple had long known the phone finding patents he'd solicited patent counsel and two executives help patenting before Apple's later filings were intentionally filed misjoinder in other employees' names. As such, Apple's actions here constitute inequitable conduct based on incorrect inventorship—these sixteen patents are likely unenforceable.

22. Patent applicants who intentionally falsify inventorship to the PTO risk invalidation of any issued patent based on inequitable conduct. "Inequitable conduct renders an entire patent (or even a patent family) unenforceable . . . ." *Therasense, Inc. v. Becton, Dickinson & Co.* 649 F.3d 1276, 1292 (Fed. Cir. 2011) (en banc). "To prevail on a claim of inequitable conduct, the accused infringer must prove that the patentee acted with the specific intent to deceive the PTO." *Id.* at 1290. Accused infringers may allege inequitable conduct as a defense in patent litigations. In light of the particularly strong written evidence plaintiff cites, one could therefore infringe on any of the sixteen patents herein; since Apple deceived the PTO.

23. For example, in 2000, the Federal Circuit found that the District of Massachusetts did not abuse its discretion in holding the asserted patents unenforceable for inequitable conduct, based on incorrect inventorship. *PerSeptive Biosys., Inc. v. Pharmacia Biotech, Inc.* 225 F.3d 1315, 1322-23 (Fed. Cir. 2000). The Federal Circuit found no clear error in the District Court's finding that there were, "at least five specific instances of intentional falsehoods, misrepresentations, and omissions" directed to the material issue of inventorship. *Id*. at 1322.

24. Such nonjoinder risks stem from the Constitution, which provides, "*The Congress shall have power . . . To promote the progress of science and useful arts by, securing for limited times to authors and inventors the exclusive right to their respective writings and discoveries*." *U.S. Const.* Art. I, § 8. Since the Constitution requires that patent rights are to be granted to "inventors," this means that those rights can only be granted to individuals and cannot be granted to business entities. Since corporations and business entities cannot take actions without people acting on their behalf, only people can conceive of and contribute to patentable inventions.

25. Reflecting the common law, 35 U.S.C. § 261 states that, "subject to the provisions of [the 1952 Patent Act], patents shall have the attributes of personal property." Plaintiffs interest in the patents in-question is supported by binding precedent. *Filmtec Corp. v. Allied-Signal, Inc.* 939 F.2d 1568 (Fed. Cir. 1991).

26. In § 256 determinations, the Court must begin, "with a construction of each asserted claim to determine the subject matter encompassed thereby." *Trovan Ltd. v. Sokymat SA, Irori* 299 F.3d 1292, 1301 (Fed. Cir. 2002). After defining the invention, the Court, "is then to compare the alleged contributions of each asserted co-inventor with the subject matter of the properly construed claim to then determine whether the correct were named." *Ethicon, Inc. v. US Surgical Corp*. 135 F.3d 1456, 1460-63 (Fed. Cir. 1998). For example, the Federal Circuit in 1998 affirmed that an electronics technician was a joint inventor with a medical doctor on a patent for a surgical instrument; based on the technician's conception of several features of it. *Ethicon*, 135 F.3d at 1462-64.

27. In 2012, the Federal Circuit affirmed that a scientist's development of a method of making the genus of claimed chemical compounds was, "enough of a contribution to conception to pass the threshold required for joint inventorship." *Falana v. Kent State Univ*. 669 F.3d 1349, 1359 (Fed. Cir. 2012). "Where the method requires more than the exercise of ordinary skill . . . the discovery of that method is as much a contribution to the compound as the discovery of the compound itself.'' *Id*. at 1358. Plaintiff demonstrates his *conception* and inventorship by a

significant enough threshold to demonstrate eligibility for joint inventorship of each patent.

28. In all inventorship disputes, conception must be proven by the rightful inventor. “Conception is the touchstone of inventorship, the completion of the mental part of invention.” *Burroughs Wellcome Co. v. Barr Laboratories, Inc.* 40 F.3d 1223, 1227-28 (Fed. Cir. 1994). Inventors listed on a patent aren’t entitled to remain on them if they cannot submit corroborating evidence. *University of Colo. Found. v. American Cyanamid Co.*, 105 F. Supp. 2d 1164 (2000). The Federal Circuit requires evidence to corroborate a purported inventor’s testimony, in order to avoid temptation to remember facts favorable to their case by the lure of protecting their patent, or, defeating another patent. *Mahurkar v. C.R. Bard, Inc.* 79 F.3d 1572 (Fed. Cir. 1996).

29. Further, named inventors cannot be joint inventors without, “some element of joint behavior, such as collaboration or working under common direction; one inventor seeking a relevant report and building upon it, or, hearing another’s suggestions at a meeting.” *Kimberly-Clark Corp. v. Procter & Gamble Distrib. Co.,* 973 F.2d 911, 917 (1992). Therefore, named inventors, “must contribute in some significant manner to the conception or reduction to practice of the invention.” *Pannu v. Iolab Corp.* 155 F.3d 1344 (Fed. Cir. 1998).

30. The Federal Circuit previously applied a “*rule of reason*” analysis in order to determine whether a putative inventor has sufficiently corroborated his claim of prior conception. *Price v. Symsek* 988 F.2d 1187, 1195 (Fed. Cir. 1993). A "*rule of reason*" analysis is applied to determine whether the inventor's prior conception testimony has been corroborated. *Coleman v. Dines* 754 F.2d 353 (Fed. Cir. 1985). See also *Holmwood v. Sugavanam* 948 F.2d 1236, 1238, 20 USPQ2d 1712, 1714 (Fed. Cir. 1991) (applied in reduction to practice determination) and *Berry v. Webb* 412 F.2d 261, 266, 162 USPQ 170, 174 (CCPA 1969) "There is no single formula that must be followed in proving corroboration.". An evaluation of all pertinent evidence must be made so that a sound determination of the credibility of the inventor's story may be reached. *Coleman*, 754 F.2d at 360, 224 USPQ at 862. Factors bearing on the inventor's credibility and on whether the inventor's testimony has been adequately corroborated are:

> (1) delay between the event and the trial, (2) interest of corroborating witnesses, (3) contradiction or impeachment, (4) corroboration, (5) the corroborating witnesses' familiarity with details of alleged prior structure, (6) improbability of prior use considering state of the art, (7) impact of the invention on the industry, and (8) relationship between witness and alleged prior user.

In re Reuter 670 F.2d at 1021 n. 9, 210 USPQ at 255. Notwithstanding this list of factors, case law clearly mandates some type of corroborating evidence to support an inventor's testimony. *Coleman*, 754 F.2d at 360, 224 USPQ at 862.

31. Joint inventorship has been described as "one of the muddiest concepts" of U.S. patent law. *Mueller Brass Co. v. Reading Indus., Inc.* 352 F. Supp. 1357, 1372 (E.D. Pa. 1972), however, the plurality of undisputed facts here may provide District Courts a clearer precedent than *Shukh v. Seagate Technology, LLC* 803 F. 3d 659 (Fed. Cir. 2015); when applying a "*rule of reason*" test. *Shukh* helped establish that a putative true inventor may seek a correction of ownership when reputational damage has clearly occurred. This case may clarify future § 256 cases; by reinforcing the necessary link between patent nonjoinder and reputational damage. A plurality of evidence supporting *conception* and *inventorship* should prevail; the "*rule of reason*" test shouldn't require such insurmountable burden an honest inventor cannot reasonably prevail. In other words, the corroboration requirement of the "*rule of reason*" test shouldn't be extended past reasonable bounds—this "formula" is poorly defined since *Berry* in 1969.

FACTUAL BACKGROUND OF INVENTIONS

A. Phone-finding Patents

32. Plaintiff was awarded a '631 patent for battery firmware technology he solely invented during the development of the MacBook Air computer at Apple. This allowed the practical sustainability of a product design where the laptop battery is permanently attached (not user serviceable) and thus increased plaintiff's credibility and value at Apple.

33. After plaintiff lost his original iPhone in Milwaukee, Wisconsin in 2008, he invented a novel method and apparatus for locating a lost smartphone or computing device; which solves the longstanding problem of an honest finder not being able to return the device—as well as the true owner reliably finding the device, and, proving it rightfully belongs to them.

34. After conducting a basic *reduction to practice*, plaintiff detailed his invention in a Radar application ticket to begin implementation (which is date/time stamped and unchangeable) before "shopping" the new feature internally to decision-makers at Apple.

35. The market advantage predicated Apple should allow users to retrieve their lost devices using its cloud services; which allow for solving the longstanding problem of reliably retrieving a lost smartphone—which may be found by either an honest finder, or, a thief.

36. Plaintiff knew prior art existed for finding (only) the static location of a lost smartphone, but, realized that even if he'd had a Blackberry instead of an original iPhone, such prior art still *would've been of little use in helping him retrieve his device*. Plaintiff knew that his

iPhone was somewhere in Miller Park during a Milwaukee Brewers game, but, he had no way to signal to the eventual honest finder (a stadium employee) that it was actually his phone and not another patrons, no way of determining if or when the phone may have been taken outside the stadium, no way to continue tracking the phones location over time as the battery depleted, no way to inform an honest finder that the phone was lost, and finally, no method to allow an honest finder to contact the plaintiff (or somebody he knew) to notify him of its whereabouts.

37. Lastly, no way to signal the device so that it could be found if it'd become "lost" but was still in the owner's proximity existed; such as under a sofa cushion, or in the original case depicted herein…falling under the plaintiff's stadium seat.

38. Plaintiff had several volunteers helping him search for his iPhone during the MLB baseball game. Since vanishing after plaintiff had held open a heavy door for an elderly patron, it was unknown if a thief had removed it from the nearby ground, or, if it'd been lost at an earlier point travelling to his seat.

39. Plaintiffs iPhone was recovered by a stadium employee in a nearby concourse. The employee took the device to the lost and found desk, whereupon plaintiff similarly was inquiring about it. Plaintiff realized he had no method of proving his iPhone was in-fact his and was lucky that *very few* iPhones were then in-use by the public. While plaintiff offered to unlock the phone to prove that it was his, this is often not possible, or, the device may not have a passcode.

40. Numerous other factors have contributed to this problem as smartphones have become more common; with many lost devices being found, but never retrieved from the resulting found property custodian (or police station) because of the standing problem of proving the device truly belongs to the owner claiming it.

41. Determining a device's physical location once statically (as in previous art) is not always very helpful as the only method of returning the device to its rightful owner.

42. Plaintiff realized that if he'd had a lost "discovery" mode on his iPhone, he could have actuated it with one of his wife's devices. This would have given the honest Miller Park employee who found it clear instruction as to how to contact the true owner. When arriving at the lost and found desk, plaintiff could have explained that his lost message and wife's phone number were on the display screen when the home button was pressed. This proves beyond doubt that the device belonged to the true owner—and not an opportunist thief.

43. Later at Milwaukee's Mitchell International Airport, the magnitude and regularity of lost devices became evident to plaintiff, as loudspeaker messages were played for several items

of value which travelers had mistakenly left at the security checkpoint, including mobile phones. This caused plaintiff to reason that police had a stricter definition of discerning whether an expensive smartphone belonged to one inquiring about it. The simple problem of having recovered devices of similar models or colors makes authenticating lost property in large public locations much harder. An additional burden exists of not mistakenly giving a lost smartphone to the wrong owner in good faith—since it's easy for multiple persons to be looking for lost smartphones, many of which look very similar to each other, and, all of which were lost in the same area. Even when honest finders are exclusively involved in finding and attempting to return lost electronic devices, many problems still existed. Again, observing that an iPhone was lost in an airport statically on a map does nothing to help the true owner either recover it, or, prove that it's rightfully their device once found and not another customer.

44. Plaintiff realized that if he lost his iPhone in the airport, even if the other barriers to identifying him as the owner didn't exist, he'd still have difficulty recovering it without the ability to send a custom notice to the phone that any user could see; without requiring a passcode, or, knowledge of how to specifically operate a smartphone. Eventually, we all must leave on a flight, so, it becomes much harder to contact the owner while they're in transit skyward, and now, without a phone to be reached at. Sending a smartphone messaging to call the airline or a family member becomes increasingly valuable. Even when arriving at a final destination, most people do not have another telephony device. Even if one has a secondary mobile phone, there's no method to learn about the lost device—without an honest finder being given the means to communicate with the true owner.

45. Plaintiff realized device security needed to be guaranteed for such functionality; so that only a registered user could activate or deactivate a lost "discovery" mode; else it could be compromised, manipulated or accidentally used without the true owner's knowledge, or, in some cases, used to wrongfully track an individual's movements. The solution to this problem was to have only one user account allowed to operate the feature, and, it needed to be used with the same credentials on a second device—which also authenticates with a cloud server.

46. Plaintiff realized using such a server allowed for an honest and secure way of independently verifying that the true owner of the smartphone had properly vetted credentials. This helps prevent a third party from intercepting (or otherwise assuming) the authority of the true owner without their knowledge.

47. Plaintiff further realized that using a cloud server also provided an easy method to

help manage the device check-ins, and, dynamic location information requests themselves from being used or accidentally being made available to a third party. A secure connection can thus be made in both directions; irrespective of whether the communication mediums a cellular network, or, a switched network providing wireless Internet access.

48. Plaintiff determined Apple already had such a cloud infrastructure to manage secure connections to its devices, even if it was a computer and not a smartphone. A reduction to practice could occur completely on Apple's demark, without requiring a third-party.

49. Plaintiff was granted privileged access in 2007 to Apple's cloud servers, as a result of his work on the "Back to my Mac" networking feature built into Mac computers; which allows one to connect to other Macs they owned, when connected using a uPnP or NAT-PMP compliant network connection. [1] Plaintiff had the ability to not only run queries to determine reachability statistics of a registered device, he could also "watch" to see if a device had registered for use—a vital part of his reduction to practice with a cloud-based, user account. Plaintiffs work on Back to my Mac saw him involved in the development and support of the feature, as well as its documentation. Moreover, he worked with a small team of network engineers who used Apple's cloud servers to manage the features usage and supportability when issues arose.

50. An example screenshot taken of the cloud server operating from the console on February 5, 2008 is presented by plaintiff in **Exhibit 15**.

51. As a result of performing expert, high-level engineering diagnosis and troubleshooting when supporting and validating unreachable computers and routers, or, bugs in the product itself, plaintiff knew how to manipulate network location data; which, in-turn would produce the approximate geographic location.

52. Rush Limbaugh famously complained on his nationally syndicated radio show, when Back to my Mac didn't allow him to see or connect to his Florida-based Macs. After the logs and other info necessary were given to plaintiff, he diagnosed the issue as a connectivity bug affecting computers with the same host name, which was then fixed in an update. Plaintiff used his exclusive access (which most engineers did not have) to the cloud severs to validate Mr.

[1] With Back to My Mac, you can connect to your other Macs securely over the Internet.
https://support.apple.com/en-us/HT204618
Ironically, plaintiff wrote the original version of this article, which now has a significant error. It states the minimum version of OS X required is 10.7.5, but, the recommended earliest version is actually 10.5.7. Otherwise, Mr. Limbaugh would never have been able to use the feature; which debuted in the initial 10.5.0 "Leopard" release. Apple issued a number of connectivity fixes in 10.5.7 resolving all known support issues plaintiff had identified. Further, plaintiff recommended 10.5.7 as the minimum supported version, which was done with that updates release.

Limbaugh's case, and, later verified that his computers successfully connected after supplying him a test fix—later incorporated in the public Mac update. Plaintiff also arranged to have a Knowledge Base article written by the .Mac support team to provide a temporary workaround, until such time as the software update was released. See **Exhibit 16**.

53. Mr. Limbaugh was thrilled, stating on air he was appreciative of Apple's troubleshooting response and instructed his IT assistant not to reveal plaintiffs name, or, anyone at Apple involved; for fear of them suffering reputational damage, as later reported in Fortune magazine. [2] Note this *Huffington Post* story explains that the plaintiff was assigned to investigate (mainly) his Back to my Mac issue with Mac OS X Leopard. [3] This occurred initially at the request of the Office of the CEO. Mr. Limbaugh had experienced a Leopard software bug, having previously well-understood how to use his computers.

54. Plaintiff couldn't have imagined his novel method to reliably retrieve a lost smartphone or computing device would end up much later causing him reputational damage from his own employer; who's CEO Mr. Limbaugh had appealed to for assistance, and, who'd previously mentored the plaintiff before recruiting him to join Apple.

55. In retrospect, Mr. Limbaugh seemed to predict and protect the plaintiff from unreasonableness by Apple for his diligence in helping him; a work habit ultimately causing his later untimely and wrongful termination. The multiple innovations plaintiff generated which're eligible for patent protection may have additionally caused his demise (through no fault of his own) from engineers and management who felt somehow upstaged; particularly his own manager.

56. Plaintiff completed a successful reduction to practice a few weeks before submitting a Radar ticket for development consideration, and, date/timestamping for Apple's patent counsel to use in prosecuting IP. Plaintiff attached his lab notebook entries (which are contained as exhibits) as well as sharing them with several employees.

57. Plaintiff had simulated making and receiving a request to initiate a lost "discovery" mode by manipulating and observing the cloud server for the corresponding user account. Plaintiff was able to verify that he could get lookup information on-demand every time he ran his test script. Log entries and database records showed that the privileged user account

[2] Rush Limbaugh gets special treatment from Apple. http://fortune.com/2008/03/12/rush-limbaugh-gets-special-treatment-from-apple/

[3] Rush Limbaugh Pleads to Steve Jobs on Air: Help Fix My Mac! https://www.huffpost.com/entry/rush-limbaugh-pleads-to-s_n_91866

logged in on both the iPhone and the cloud server were successful. Plaintiff further proved the communications were happening with encryption; which meant the queries between the “lost” iPhone and the cloud server weren’t susceptible to interception or third-party manipulation. Plaintiff thus declared a successful reduction to practice.

58. Plaintiff began exploring the problem of a lost iPhone still needing communication when the battery was depleting each day; even without an honest finder or thief intervention. Push communication for email is commonly used to update changes in the background, and, can also be used for some third-party applications which synchronize data. Plaintiff found that it was possible to *selectively* engage “Airport Mode” as a workaround to suppress network instructions or queries; which greatly extended the battery life, while allowing only a cellular connection.

59. In the same method wireless network drivers are put into a low power state mode to save energy and activated when needed in computers, plaintiff discovered the baseband connection to the cellular network could be managed in a similar way; by setting a preference that only one user could execute. This would allow iPhone to be sent the remote command instruction to begin conserving the battery energy when it reached a defined threshold. It also allowed this instruction to not be abused by another person if they were to use their phone briefly, as this instruction would only be activated remotely from the cloud server.

60. Plaintiff determined after his initial *reduction to practice* that the power savings threshold could be set to a lower value, like 50% of total charge; to prevent other applications’ instruction sets updating in the background, if the device was lost in one’s home, for instance.

61. Plaintiff deduced the usage case of customers losing a device and enabling lost “discovery” mode when it’d just fell under their sofa cushion, was likely quite high. While the phone may have a message on it to call your spouse’s phone if found, and, will appear as being at one’s residence on a map, it still doesn’t help one locate the device inside the already known area.

62. It occurred to plaintiff that playing a sound, or, even speaking something using a recorded message (in whatever language the device was setup) would accomplish this task using the same internal remote messaging already being used. Playing beeps and explosions from a sound effects soundtrack on plaintiffs iPhone (while it was placed under a sofa cushion) proved this reality, and, constituted another reduction to practice. Plaintiff further knew how to disable sleep on Mac computers when the display is closed and latched, which isn’t available as a customer setting; for obvious reasons. Despite the speakers now being located under a closed display bezel while in-use, plaintiff found that a Mac placed under a blanket (or large beanbag

chair) was still audible enough to be heard; even with just simple beeps, such as Morse code.

63. Determining a threshold for when to begin power savings when the device had been declared lost was another consideration plaintiff contemplated. If several email accounts were using push and/or applications were updating in the background, it would be premature to stop this if their phone was found in another room of their home or office 15 minutes later. One would certainly want to confirm the iPhone was lost someplace nearby; which causes the initial lost "discovery" mode to start, however, they'd take comfort knowing the device wasn't in danger of being stolen and continue using it a few minutes later, after retrieval. Every other previous time plaintiff had lost his personal mobile devices, root cause from falling out of a pocket, a belt device, or, a failure of the belt device. If the owner was at home or work, the device presumably will be found rather quickly. Even if it took an hour to find a lost phone in a large home or office, it may not be prudent to cause the phone to be effectively "turned off" and require manual update; for operations which normally would occur asynchronously without user direction. It also occurred to plaintiff that the distinctive noises used when incoming email and text messages arrive may function as a supplemental aide; when attempting to find a mobile phone that's fallen under the couch, or, is buried in a laundry basket, for example.

64. Realizing this feature would require the use of the cloud server infrastructure to implement his claims, plaintiff contacted VP Eddy Cue on January 27, 2009. Mr. Cue oversaw Apple's cloud services and replied that it was a good idea and now, "something we have on our list to consider."

65. Plaintiffs assertion about the necessity of Mr. Cue's cloud server team being necessary for implementation is a fine example of him already conducting a reduction to practice, otherwise, it'd be premature speculation to solicit input for an iOS feature from another, unrelated team. iOS could easily open a map or gather network information without the necessity of a cloud server. Plaintiff had deduced that a privileged cloud-based user account must be used in order to solve this longstanding problem; otherwise an existing AppleID credential could've been used. For convenience after implementation, AppleID's could by synchronized with a cloud-server account to inherit such a role, but, it wasn't possible at that time.

66. Plaintiff had discovered that a server connection could audit, parse and otherwise manage conditional responses on a lost device when logic was necessary; which a typical AppleID account cannot do, and, was widely adopted at the time for exclusively using Apple's services. An Apple cloud-based account was only used at this time to send and receive email. No

predetermined logic was available with either such accounts; which is necessary to accomplish the novel tasks needed to implement this feature. This will be discussed further in the claims.

67. Instead of exclusively providing push email service, plaintiff determined that cloud-based user account usage at Apple would need amending to allow the intelligent responses he needed to properly authenticate and provision a lost device "discovery" mode; which runs like an application on a remote server—managing the user-defined parameters of the features functionality, and, ensuring that several predetermined factors are enforced, as well as providing real-time updating of the devices current state and location. Prior art doesn't accomplish this.

68. If the lost device was moved ten miles, for example, previous art wouldn't be able to produce a movement path while it occurred, or, otherwise display a waypoint the device had previously travelled. Previous art only showed a device in the location it resided when the discovery request was made. It offered no autonomous or asynchronous ability to dynamically update the user when something changed with the device, after the owner declared it lost.

69. A thief could steal a smartphone at a train station and the user could see it was at the station, but, couldn't discern if the device was then suddenly taken on a train. The same location would show if the owner than searched a few minutes later, but, the network connection was unavailable or slow; especially if the device was powered-off. While the owner was still frantically checking the public areas of the train station, the thief is now many miles away.

70. A thief could thus disguise their "tracks" and present location of the phone by than changing the SIM card and turning the phone back on. The plaintiff's invention allows this to be combated by the cloud server account taking over the mobile devices management until it's been deactivated. Even if the battery was fully depleted and then recharged by a thief, the iPhone would remain in lost "discovery" mode until deactivated.

71. This has the additional side-benefit of discouraging theft, as the stolen device instead becomes a liability and cannot be used; thus, its expensive cash value is no longer realizable for a thief. As long as the application session for the lost phone continues running on the cloud server under the direction of the true owner, it cannot be defeated on the device itself.

72. Using a .Mac email account could not populate a devices location on a map as previous art does; which is clear to those unskilled in the art of networking or programming. Whereas Apple's struggled to understand this, a jury most decidedly won't. Why the plaintiff would recruit a busy cloud and music executive *in re* an iOS feature for a smartphone and then attempt patent protection for Apple is clear indication that the existing problems facing lost

computing devices had been overcome solely by plaintiff.

73. Plaintiff emailed VP Scott Forstall (of iOS) on February 18, 2009 and explained his new feature; along with his necessary, previous exchange with Mr. Cue. Mr. Forstall responded that, “it was a good suggestion.”

74. Before the June 15, 2010 release of the “Find my iPhone” feature, plaintiff emailed Mr. Forstall on March 8, 2009; asking if patent protection should be sought, but never received another response from him. Voicemails left for Mr. Forstall were not returned.

75. Plaintiff was then unsuccessful soliciting Apple’s patent counsel, whom he emailed on March 24, 2009. PC acknowledged plaintiffs’ message less than an hour later, but never responded further. See **ECF No. 19** for sealed **Exhibit 6**. Despite being the sole inventor of this novel method and apparatus to locate a mobile device, plaintiff was nonjoinder from thirteen patent applications; later filed in 2012 and 2013, and, subsequently granted by the PTO.

76. After plaintiff discovered he was wrongfully terminated while sick at home working on an issue at the CEOs request, his subsequent investigation with counsel found multiple patent misjoinder and nonjoinder.

77. Plaintiff’s counsel sent a demand letter to Apple, see **Exhibit 13**. After promising an investigation by an Apple Director which never occurred (see **Exhibit 14**) talks broke down and Apple didn’t respond further—even after CEO Tim Cook asked legal to follow-up.

78. Severe medical issues affecting plaintiff and his counsel, coupled with unemployment since Apple wrongfully terminated him and damaged his reputation, has caused his *pro se* status. Apple’s unclean hands continue gifting plaintiff, years after his departure.

79. It’s obvious to those unskilled in the art that plaintiff’s novel method and apparatus to find a lost smartphone is not previous art; else the commanding market advantage for such functionality would’ve been realized long ago by other competitors, or, third-party software developers. It explains best why the PTO even granted Apple the phone-finding patents.

80. Plaintiff developed several novel solutions for the longstanding problem of reliably retrieving a lost smartphone, especially if discovered by an honest finder. Experiments and work product plaintiff disclosed to others at Apple was stolen and used to implement plaintiffs *enabling* invention; misjoinder patents were then intentionally prosecuted by Apple for individuals with no role whatsoever in developing a solution now adored and used by millions.

81. Worse, Apple’s patent counsel was disclosed with this invention over two years earlier. The overwhelming dated, written evidence suggests that Apple not only had unclean

hands, but, intended to intentionally defraud plaintiff from the very beginning. In criminal matters, this is known as *mens rea*. Apple than ignored plaintiff's counsel for over a year before he filed litigation. Now that Apple's compelled, they argue plaintiffs novel work was previous art; demonstrating they cannot interpret basic block diagrams, descriptions or drawings. Apple thus argues it doesn't understand basic authentication, batteries, networking or programming; despite all being necessary for its products to operate. This ignorance proves Apple's malicious intent, else they would not sell hundreds of millions of devices per year. The question of legal ethics must also be raised, with Apple's patent counsel intentionally deceiving its client.

82. The Court could identify the plaintiffs unique work product from examining his lab notebook, yet the assignee (Apple) of the patents in-question has been unable. Even if Apple's prior art regurgitation under the best light is selective presentation of incomplete facts, it's still incorrect. *Roberts v. Ball, Hunt, Hart, Brown & Baerwitz*, 57 Cal.App.3d 104 (2nd Dist. 1976).

B. Passbook

83. Plaintiff previously worked for Cal State Fullerton; in an IT role oriented (but not limited) towards Mac users. He thus supported the Cultural Affairs department; who's tasked with the universities box office, which includes performing arts ticketing for hundreds of events per year. Plaintiff also supported Titan Athletics, including its baseball broadcasting operation.

84. Plaintiff regularly interfaced with a third-party who made expensive software for managing a box office operation, which ran on a Macintosh AppleShare IP Server. The software offerings in the ticketing area were unreliable, and, didn't solve the several problems institutions who cannot afford Ticketmaster had; which included issuing free tickets and not being able to sell tickets online. Redeeming tickets not sold directly from a box office can be impossible, with most usage cases herein not being able to sell tickets online. Patrons still had to call or visit the box office to purchase tickets, as "print at home" ticketing didn't exist yet.

85. Plaintiff identified the problem of ticket redemption as the hurdle in solving this longstanding problem, and, it represented a challenge existing outside the ticket issuing authority. This is why third-party software capable of selling event tickets was unable to solve this problem, making it a necessarily incomplete solution. At the very least, human labor was heavily required at multiple tiers to support paper tickets; being in use for over a century and still needing to be printed—even if they were sold with software at the box office or other point-of-sale.

86. Selling free or reduced-price tickets was impossible with Ticketmaster, as it costs the hosting entity to process each transaction; which isn't cost-effective for education and

nonprofits. While Ticketmaster can reliably sell tickets online, they couldn't be redeemed without a human to audit and interpret their authenticity. As a result, while Ticketmaster offered a convenient solution for patrons to choose and purchase tickets online, it still failed to solve the longstanding problems inherent with redemption; as either paper tickets were issued in the mail, or, were made available at the event box office, aka as Will Call. Any labor savings derived from online sales was thus replaced with delivery and redemption, making it inefficient for the venue.

87. One problem ticket redemption ushers face in education and nonprofits is being a dedicated function not suitable for internship credit. It's a paid or volunteer job, despite not being able to also accomplish other related tasks, like helping patrons find their seats. For free and reduced-price events, such cost can be prohibitive enough to limit numbers of performances which might otherwise occur.

88. The combined cost of providing reliable ticket sales and redemption is prohibitive without also paying Ticketmaster to offer online sales and physical delivery; which require barcode scanners and a standing contract, as well as additional training.

89. In addition to such issues, the ability to make changes to a venue seating plan based on performance was not easy and cost prohibitive. Box office software relied upon standard templates, which would upset the order of the corresponding database if overly manipulated. The result of trying to change the seating plan for one event dynamically could later threaten the later stability of the system; causing the software to then fail at the worst possible time—often when the box office was open. Flexibility was impossible to manage with reliability, while also creating an unruly amount of labor for the ticket administrator; to make one seating plan change.

90. In 2002, plaintiff began charting possible solutions for these varied problems, usually after intervening when the University ticketing operation suddenly failed. Plaintiff interviewed many persons associated with managing and producing ticketed performances, including professors who taught courses with live performance requirements. The number of events potentially offered for the community was hindered by the standing problem of no solution existing for education, nonprofit or small civic ticketing outside Ticketmaster; which still didn't additionally solve the longstanding problem of virtual ticket redemption.

91. Worse, even if entities could afford Ticketmaster, they still had the extra costs associated with physically redeeming paper tickets—as well as still needing box office staff to handle transactions before and during the event. Having one person responsible for the box office (as in such situations) means its impractical to expect them to work seven days a week. Lastly, the

need to handle cash meant existing students already working on the production in some capacity couldn’t be tasked to handle managing the box office for an hour or two. The public expects a box office to sell other tickets than just the current event occurring, which also introduces training and security access issues for the student volunteer who’s helping with makeup, etc.

92. Independent ticket management represents such a tangible point for education and nonprofits it often precludes them from selling tickets outside their own distribution means—the cost exceeds any profit. The innovations in the Passbook patent affords a method to escape paying such fees, irrespective of whether it’s exercised by merchants. This proves beyond doubt plaintiff’s disclosures in his IPA wasn’t prior art; else it’d been implemented much earlier. Digital redemption allowed the second prong of online ticket sales to thus succeed for the first time.

93. Creating a method and apparatus to solve these longstanding ticketing problems required plaintiff to explore different methods of solving them with app and web technology; which uses a locally owned and controlled server to than communicate and audit transactional detail when a patron arrives to redeem their ticket, change their ticket on-demand for different performance or time, or, requests a refund. The ability to dynamically change a seating plan also required using a database for experimentation and reduction to practice with a lasting solution.

94. The ability of redeeming tickets with an electronic device (such as a computer or PDA) or printed at-home avoided the problem of needing dedicated, paid staff to take tickets; allowing event ushers to instead focus on helping patrons find their seats and answering questions about the performance. Event security could even listen for a beep from University equipment in entrance lanes, which would indicate a valid ticket had been presented to the authorization reader at the entrance. So many computers and PDAs used then (and even now) in academia meant it was reasonable for bringing along to a campus event. Most faculty, staff and students are regularly instructed to not leave their electronic devices in their vehicles for theft reasons, so, such devices are regularly silenced or otherwise powered down in backpacks, or, under seats.

95. Using mobile devices to redeem tickets which were also purchased online wasn’t just a novel convenience itself; it meant one person could reliably manage the operation of taking and validating patrons’ tickets, instead of 5-6. For events which utilized event security, one guard at the entrance could alternatively handle such duties themselves, freeing up valuable financial and time resources for nonprofits. This also helps combat the problem of having two ticketed performances occurring with overlap in two different venues on-campus; as it only requires two persons, instead of ten. Scaling the size of the performance or venues upwards for sporting events

or local touring events causes an even higher level of efficiency in needed resources. Using automated ticket redemption means one person can manage both automated entry lanes, and also, a single line for exception cases. Such efficiency means less ushers; especially for events which aren't sold-out. Since ticket sales are known before the event, it allows academia and nonprofits to not overwork their usher staff, some of which may be volunteers, or, may comprise the bulk of their paid, hourly event staff. Overall, digital ticket redemption means local communities can enjoy more performing arts than could otherwise be staged.

96. Some of the raw methods/tools plaintiff used to prototype his innovation appear in the two sentences of approximate space plaintiff was given to declare his previous art in his Intellectual Property Agreement; which Apple HR required him to complete during orientation on his first day of employment—explaining he otherwise couldn't be paid or start his position, as expected. Plaintiff asked for more paper and more time, both of which were declined by Apple's HR. Plaintiff assumed he could amend his IPA later, since the central premise of developing a complete ticketing solution was listed. The plaintiff's recruiter had no guidance or ability to affect an IPA amendment, either. Given plaintiff had relocated and was in temporary housing without his possessions, losing his new position was of particular concern.

97. In the same sense one cannot object to concrete and heavy machinery being used to erect a new building, its similarly obtrusive to expect a software creation not to be created using databases and programming. The tools plaintiff used for creating such an important invention cannot be used against him; to deny his patent eligibility. A patent for a new kind of concrete, for example, would still need to be mixed and transported with existing construction equipment while being developed. It's fundamentally impossible to use an application on any device which handles ticket redemption without a database. For any responsible scale, it's necessary for two databases to exist—one for the local device the patron uses to buy and redeem the ticket, and yet another to manage the overall server for the venue using the invention.

98. Steve Jobs mentored plaintiff previous to his start date at Apple, being particularly impressed with his acute problem-solving ability and attention to detail. In 2006, plaintiff disclosed the ticketing problems and his solution; as it represented a tangible profit opportunity for Apple. One solution plaintiff explored with Mr. Jobs was simply adding ticketing services to a dedicated section of the iTunes application, or, creating a new application dedicated to ticketing. This would allow education and nonprofits to sell tickets for free, while charging a modest fee for all others. The fact commissions weren't paid by everybody would be offset by selling more

Macs, as iPhone didn't exist yet, however, Mr. Jobs and plaintiff did discuss ways of redeeming tickets sold by Apple with a Palm Treo 650; which is the smartphone Mr. Jobs was using at that time. Plaintiff was using a Blackberry. Discussion loomed over whether it was reliable to use third-party devices; as opposed to offering digital redemption only on Apple devices, which could be strictly controlled.

99. This prompted the thought experiment of attending a concert at nearby Shoreline Amphitheater; redeeming tickets using a Palm Treo 650, a PowerBook, and, a printed barcode on paper. The use of paper was deemed not environmentally sound and inconvenient compared to a device you carry most of the time anyway, one of which Apple currently made. Taking a notebook computer to a commercial concert or sporting event is impractical and might conflict with local security restrictions. Mr. Jobs and plaintiff agreed that the Palm Treo 650 was the most logical redemption method, overall.

100. Since iPhone was under development for several years before its announcement without most employee's knowledge at Apple, it's unclear if Mr. Jobs was already considering the later suitability of iPhone for redeeming tickets, or, if his discussions with plaintiff helped convince him that an Apple phone project was worth the risk. In the same manner Mr. Jobs flushed out the problems of ticket redemption with plaintiff; he later did this with other employees *in re* iPhone's multitouch interface…by using a finger for all operations. This is why Mr. Jobs indicated that this feature should be tabled until smaller Apple devices were available.

101. Plaintiff was concerned about HRs shortcomings with his IPA and mentioned this at a meeting a few months later with Mr. Jobs, who indicated that nobody would challenge his authority in this area—he'd ensure plaintiff was included in patents. Mr. Jobs than whimsically stated that it didn't matter, "because [plaintiff] was always going to work for me and that neither of us are going anywhere." He then mentioned the monopoly Ticketmaster had could further be in jeopardy, and, that he was proud of plaintiff for his diligence in solving the problem for education and nonprofit; while still making it profitable now for Apple.

102. One example plaintiff had mentioned in this area with Mr. Jobs was his idea of adding a ticket sales field to participating artist pages on iTunes. This would allow tickets to be purchased directly from iTunes for concerts, while offering an opportunity to sell customers digital music from that artist at the same time. Concerts seemed like an ideal trial for Apple to enter the online ticketing marketplace; as the element of risk was manageable with a box-office to intervene if a technical problem occurred. The impetus also existed, Mr. Jobs reasoned, for bands

who weren't selling their music on iTunes to begin doing so; as a result of the vastly more favorable ticketing offering Apple could offer them that competitors couldn't. The corresponding result was Apple standing to profit from their iTunes music sales; in return for providing free or deeply discounted ticket sales, which could be redeemed digitally for convenience. Mr. Jobs theorized if Paul McCartney was touring, it was a great opportunity to also offer Beatles albums on iTunes. Plaintiff agreed and stated that the benefit afforded to lesser-known artists was significant and still represented a tangible profit center for selling their music on iTunes. Trying to find a cost-effective way to sell tickets for lesser-known artists would help them discover the equality they'd enjoy; selling their music alongside major recording labels in the same place.

103. It was clear Mr. Jobs saw the potential for Apple to offer a more compelling solution than Ticketmaster; since they could do no better than using paper as a redemption model, and, would lose revenue simultaneously to larger accounts who simply wanted to pay less fees, while offering patrons a more elegant redemption model than Ticketmaster. In the last mention of this topic, plaintiff voiced his concern to Mr. Jobs that Ticketmaster would somehow steal (or later develop through parallel innovation) his idea. Plaintiff never expected his own company would finally start development and then leave him off the patent; despite having originally made an IPA declaration before his Apple employment.

104. Unlike the phone-finding patents, the Passbook patent involves additional claims not undertaken by plaintiff, however, his initial discoveries, trial and error, discussions with Mr. Jobs, IPA and his notes represent *enabling* technology—without which, the rest of the invention couldn't exist, or, have purpose. *The significant amount of time which plaintiff spent working on a solution for this longstanding problem before he even worked at Apple, and, nearly a decade before Apple pursued it carries considerable weight*.

105. Apple filed for the Passbook patent nonjoinder of plaintiff on June 9, 2013. Apple never disclosed its development or patent disclosures to plaintiff, despite being the originator of the novel idea and having an IPA on-file that legal could examine anytime.

106. Mr. Jobs death on October 5, 2011 may explain Apple's failure to join all inventors, however, *it's clear Apple doesn't consult employee IPAs before new patent filings, else this wouldn't have occurred*, and, plaintiff would've been contacted by PC for his disclosures—as Apple's obligated to perform; given the Constitutions strict guidance patents are only for individual inventors to be named and not entities.

107. *In re* the three Passbook patents, Steve Jobs was also nonjoinder. Mr. Jobs helped

plaintiff flush out the eventual claims stemming from his innovation and had suggested that mobile devices should be later used exclusively for redemption. This patent relies on a mobile phone to operate the ticket books. As principal inventor, plaintiff feels Mr. Jobs should also be added as a co-inventor; despite his being deceased.

108. The careless and offensive method Apple uses to decide inventors doesn't account for who actually participated in the claims, who previously declared IP before employment, or, even during their Apple employment. Given the co-founder of Apple is also nonjoinder on the three Passbook patents proves such negligence. Plaintiff is hopeful the Court may order Mr. Jobs also be named as a co-inventor of said patent in a corresponding certificate of correction.

CAUSES OF ACTION

A. Patent Nonjoinder Claims

COUNT 1 Patent 8,666,367

Remotely locating and commanding a mobile device

109. *The '367 patent includes the following claims plaintiff invented, specifically 1, 4, 5, 9, 12, 13, 14, 15, 18, 19, 21, 22, 23, 24, 25 and 28 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

110. **Claim 1**. A computer-implemented method performed by a mobile device, the method comprising: accessing, by the mobile device, a notification service on a server separate from the mobile device, the notification service hosting a plurality of command collection topics, where a distinct mobile device is subscribed to each command collection topic; accessing, by the mobile device, a command collection topic hosted on the notification service and subscribed to by the mobile device; polling, by the mobile device, the command collection topic subscribed to by the mobile device to determine that one or more new remote command messages have been received by the command collection topic subscribed to by the mobile device; retrieving, by the mobile device, in response to the determining that one or more new remote command messages have been received by the command collection topic, at least one of the one or more new remote command messages included in the command collection topic subscribed to by the mobile device, wherein the one or more new remote command messages identify commands to be executed by the mobile device; determining, by the mobile device, whether the command identified by the retrieved remote command message can be executed by the mobile device; publishing, by the mobile device, a result message associated with the command to a result topic hosted on the

notification service; and selectively executing, by the mobile device, the command based on a result of the determining.

111. The principal embodiment and feature of both this claim and patent necessarily concern the cloud server; particularly as depicted by plaintiff in his design notes. The cloud server in plaintiff's embodiment matches the plurality of services the notification service (or server) provide herein. Plaintiffs first mention of the server appears in **Exhibit 8**, where he discloses, "location data sent to Apple" as the third process step; followed by a discussion of communicating predetermined power state changes actuated from the server, along with, "display[ing] lost message on phone using device privilege mode, if user wishes to do so." This implies the usage of the server hosting a plurality of command collection topics, such as placing the lost device into lost "discovery" mode, displaying a message on the lost device, playing a sound on the device, or, finding its location on a geographical map. The lost device is subscribed to receiving notifications from the server as a result of the true owner declaring it lost and logging into the cloud server, whereby placing said device into lost "discovery" mode; so, functionality of the lost device is reduced to limited operations as may be issued using remote command messages. The interpretation and receiving of multiple remote commands are simply part of the overall embodiment of the invention; whereas providing the lost devices resistance is adequate, and, a communication method's available; remote command interactions may occur asynchronously—provided lost "discovery" modes still enabled by the true owner. Additionally, plaintiff disclosed that an entire subprocess of the overall embodiment is named "Message" because the notifications which may be received or sent by the mobile device from the cloud server are command collection topics. A notification and message are often considered the same in programming. For example, plaintiff expected to use the methods (using the C programming language) UNMutableNotificationContent, UNTimeIntervalNotificationTrigger, UNLocationNotificationTrigger, and finally, UNNotificationRequest to manage the remote command messages and their interactions between the lost device and cloud server. [4] This would allow a remote message to be sent and displayed on the lost device, it would allow for a message to be sent with new geographical coordinates when the location of the device has changed, and, finally, control both the actuation and cessation of lost "discovery" mode; in addition to handling

[4] Local and Remote Notification Programming Guide https://developer.apple.com/library/archive/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/SchedulingandHandlingLocalNotifications.html#//apple_ref/doc/uid/TP40008194-CH5-SW1

other remote command instructions on-demand from the true owner using the cloud server. The usage case of playing a sound as a remote command on the lost device (as discussed in the factual background for finding a device lost under a sofa, for instance) instead, the UNNotificationSound method would be used, and, appear in an example non-mutable array as "content.sound = [UNNotificationSound defaultSound];" to so cause the lost device to emulate sound as the result of a remote command message being sent from the cloud server. Plaintiff expected to use a non-mutable array for playing sound on the lost device to allow the true owner the option to also speak through the device remotely; as opposed to just playing a default sound.

112. **Exhibit 9** demonstrates a remote command message showing both the current and two former locations of a lost iPhone on a geographical map overlay; with the four buttons used to issue other remote-control messages (such as locating a selected device from a registered list) also being depicted in the lower example.

113. **Exhibit 10** focuses on inter-network connectivity between lost devices and the cloud server, which is herein managing notifications. The user record mapping in the lower example demonstrates how a plurality of devices controlled by a true owner can be subscribed (or unsubscribed) from receiving remote command message when lost "discovery" modes enabled; as well as demonstrating the connection scheme used to manage such connections and events.

114. **Exhibit 11** includes a narrative for handling devices while they're thought to have been stolen. Plaintiff explains multiple instances where remote command messages are being sent to the mobile device from the cloud server. First is the assertion that, "we could lock the device and invalidate the true passcode while privileged mode is in-use" juxtaposed with the observation that a remote command message to reverse the former command is necessary; "we must allow the device to be unlocked due to accidental enabling [of the feature] or the phone being found." Even disabling lost "discovery" mode requires sending a remote command message to invalidate the initial message; as otherwise it'd be impossible to remove the message for an honest finder after the device was found, for instance. The resulting discussion *in re* law enforcement captive modes represent the need for sending additional, custom remote command messages from the cloud server to the lost device; which may have a lost person along with its geographical proximity.

115. **Exhibit 12** shows the cloud server communicating with different data sources to present the location data of a lost device on a geographic map overlay. This also represents a remote command message to find the device being executed; with the results being shown in either a standalone application, or, a web browser. Secondly, another remote command is

depicted in a notification message displayed on the lost device, for an honest finder to use in its return to the true owner; which, has necessarily been issued by the cloud server after lost "discovery" mode had initially been enabled.

116. **Claim 4**. The computer-implemented method of claim 1, wherein the command comprises a locate command.

117. The locate command (as described in the claim) is discussed in operative detail in **Exhibit 8**, while the user interface example for locating a device is shown in **Exhibit 9**; where a "Find Devices" button than locates the devices current geographical position. This position is then charted on a user interface element with map overlay, as illustrated *supra* in the same exhibit. Apple's **Figure 8** shows the receive locate command **805** denoting the impression of the "Find Device" button in plaintiffs **Exhibit 9**. After determining the location **810** and **815**, the resulting geographic coordinates are published as a result message **820**; which is received by the cloud server in plaintiffs **Exhibit 8** and **Exhibit 10**, allowing it to generate the position(s) of the lost device on a map overlay in **Exhibit 9**.

118. **Claim 5**. The computer-implemented method of claim 4, further comprising: determining one or more geographic coordinates indicating a location of the mobile device; and including the one or more geographic coordinates in the result message.

119. The plaintiff illustrates such an example in **Exhibit 9**, indicating the location of the mobile device in a result message; listed with one or more geographic coordinates. The user interface element plaintiff depicts is very similar to Apple's example in **Figure 9** and **Figure 10**; where "Jake's" iPod is shown on map **900** and **1005**, using **915** and **1025** to denote the position of the music player. Plaintiffs example in **Exhibit 9** shows an iPhone user interface element; which indicates where on the map (precisely as in **915** and **1025**) the lost device has been located—not just in one location as Apple cribs here, but also in two previous locations in nearby cities. It's plainly obvious the lost iPhone has been most recently located (and charted geographically) in Los Gatos; with previous locations charted in Cupertino and Saratoga.

120. Additionally, plaintiff discloses the method for "presenting data of device location" in **Exhibit 12**, where a diagram depicts that the location data for the lost device may be presented in a web browser (as in Apple's **Figure 9** and **Figure 10** examples) or via an application on either a computer or mobile device; in addition to the possibility of a custom user interface for iPhone itself containing map charting. In the plaintiff's embodiment, a lost iPhone is located through a network, with the corresponding position recorded and reported by the cloud

server; which, in turn may populate any of the various application or web browser control node examples with geographic map charting, along with a result message.

121. Plaintiff mentions in **Exhibit 8** under the "Message" section, "display phone location after translating the GPS location for web display. Show device in map on web app or page" as well as the linked subprocess stating, "chart lost path since last activation by user (if running again) chart previous "check-in" spots for map." The final linked subprocess states, "display device movement with charting of each check-in."

122. **Claim 9**. The computer-implemented method of claim 1, wherein the command collection topic subscribed to by the mobile device includes a plurality of command nodes, where each command node corresponds to a distinct remote command type and the at least one remote command message is retrieved from one of the pluralities of command nodes.

123. Plaintiff describes a plurality of command nodes in **Exhibit 9**; where each command node corresponds to a distinct remote command type. In plaintiffs' example, four buttons are displayed, with only the "Find Devices" button visible—one of the three iPhones in the device list (to the right) hasn't been placed into lost "discovery" mode. Once a device has been located, the three other buttons activate. Plaintiff had planned for these buttons to play a sound to locate the device when lost in a user's proximity, remotely erase (or wipe) the device, and finally, to display a message on the display screen of the lost device for an honest finder. Plaintiff depicts this example remote command type in **Exhibit 12**, whereas the "Example Lock Screen When Lost" depicts a message indicating that the users iPhone is lost. The message also provides a number to ring the true owner; not unlike Apple's example with Jake's iPod.

124. **Claim 12**. The non-transitory computer-readable medium of claim 10, further operable to cause data processing apparatus to perform operations comprising: supplying authentication credentials associated with the mobile device to the notification service.

125. Plaintiff depicts the authentication credentials associated with the mobile device to the notification service under the "Example Process UI" in **Exhibit 9**; whereas the true owner of a lost device is authenticating using a privileged user account against the cloud server, which is also in-use on the lost device. The fields labeled for username and password are plainly evident, however, the "User Records Mapping" in **Exhibit 10** reveals additional authentication details; showing how user records and SSL are used between a lost mobile device and a known computer using the same privileged account credentials. A depiction of the cloud server and it's connected network topology with lost devices is provided *supra*, in the same exhibit.

126. **Claim 13**. The non-transitory computer-readable medium of claim 10, wherein the command comprises a locate command.

127. As discussed *supra*, the locate command denoted in this claim is represented by the plaintiff in his embodiment as the "Find Devices" command node button in **Exhibit 9**.

128. **Claim 14**. The non-transitory computer-readable medium of claim 13, further operable to cause data processing apparatus to perform operations comprising: determining one or more geographic coordinates indicating a location of the mobile device; and inserting the one or more geographic coordinates into the result message.

129. Herein this claim now reinforces the computer (or server) side of the claims for application ambiguity. The server presents and records the location data transmitted from the lost mobile device on a map overlay. As discussed *supra* in claim 5, plaintiff illustrates an example indicating the location of the mobile device in a result message; listed with one or more geographic coordinates in **Exhibit 9**. The user interface element plaintiff depicts is very similar to Apple's later example in **Figure 9** and **Figure 10**, where "Jake's" iPod is shown on map **900** and **1005**; using **915** and **1025** to denote the position of the music player. Plaintiffs example in **Exhibit 9** shows an iPhone user interface element; which indicates where on the map (precisely as in **915** and **1025**) the lost device has been located—not just in one location as Apple shows, but also in two previous locations in nearby cities. It's plainly obvious the lost iPhone has been most recently located (and charted geographically) in Los Gatos, with previous locations charted in Cupertino and Saratoga. Additionally, plaintiff discloses the method for "presenting data of device location" in **Exhibit 12**, where a diagram depicts that the location data for the lost device may be presented in a web browser (as in Apple's **Figure 9** and **Figure 10** examples) or via an application on either a computer or mobile device; in addition to the possibility of a custom user interface for iPhone itself, which contains map charting. In the plaintiff's embodiment, a lost iPhone is located through a network, with the corresponding position recorded and reported by the cloud server; which, in turn may populate any of the various application or web browser examples containing geographic map charting, along with a result message. Plaintiff mentions in **Exhibit 8** under the "Message" section, "display phone location after translating the GPS location for web display. Show device in map on web app or page" as well as the linked subprocess stating, "chart lost path since last activation by user (if running again) chart previous "check-in" spots for map." The final linked subprocess states, "display device movement with charting of each check-in."

130. **Claim 15**. The non-transitory computer-readable medium of claim 10, further operable to cause data processing apparatus to perform operations comprising: establishing a connection to the notification service over a wireless data connection.

131. The plaintiff illustrates establishing a connection to the notification service over a wireless data connection in **Exhibit 10**, whereas the "Connection Path Network" diagram shows both a cellular and wireless network connection potentially communicating with the recovery user media access control, or server. An iPhone is labeled as "iPhone Wi-Fi" and connected to the Internet cloud, which is connected to the notification server.

132. **Claim 18**. The non-transitory computer-readable medium of claim 1, wherein the command collection topic subscribed to by the mobile device includes a plurality of command nodes, where each command node corresponds to a distinct remote command type and the at least one remote command message is retrieved from one of the pluralities of command nodes.

133. As previously described *supra* in claim 9, plaintiffs **Exhibit 9** shows how each command node corresponds to a distinct remote command type. In this example embodiment, the plaintiff has one remote command message ready to be transmitted, which will locate the mobile device selected in the device list; further placing it in lost "discovery" mode. In **Figure 3** of the application, login begins the process at **305**; presenting a list of linked mobile devices in **310**, a user then selects a mobile device from managed devices in **315**, available commands for the selected device occur in **320**, and, finally, the true owner can select a remote command to be executed in **325**. The plaintiff discloses the same, identical process in his earlier embodiment. **Exhibit 9** and **10** show logins, while **Exhibit 9** also shows a mobile device selected in the device list, with the "Find Devices" button representing a remote command; just as in **320** and **325** in **Figure 3**.

134. **Claim 19**. A mobile device comprising processor electronics; a storage medium storing instructions executable by the processor electronics to cause the processor electronics to: establish a connection to a notification service on a server separate from the mobile device, the notification service hosting a plurality of command collection topics, where a distinct mobile device is subscribed to each command collection topic; access a command collection topic hosted on the notification service and subscribed to by the mobile device; poll the command collection topic subscribed to by the mobile device to determine that one or more new remote command messages have been received by the command collection topic subscribed to by the mobile device; retrieve, in response to the determining that one or more new remote command messages

have been received by the command collection topic, at least one of the one or more new remote command messages included in the command collection topic subscribed to by the mobile device, wherein the one or more new remote command messages identify commands to be executed by the mobile device; execute a command identified by the retrieved remote command message; identify in the remote command message a result topic hosted on the notification service; and publish a result message associated with the command to the identified result topic hosted on the notification service.

135. Plaintiff's illustrated *supra* how his exhibits demonstrate how a mobile device establishes a connection to a notification service on a cloud server; which hosts a plurality of command collection topics—subsequently than subscribed to future remote message commands when a lost device has detected lost "discovery" mode activated by the true owner from a cloud server. Plaintiff also demonstrates how a remote command message has been successfully executed on a mobile device. The first example's in the "Example Process UI" in **Exhibit 9**; depicting Darren's iPhone being selected from a registered device list, followed by having lost "discovery" mode enabled by the authenticated cloud server user pushing the "Find Device" button. The "Example UI" shows a second remote command execution since subscription; with said lost iPhone being presented on a geographic map overlay with its current position, alongside two previous locations since it detected appreciable movement since being declared lost. The lost "discovery" mode allows such remote command messages to be sent, which in this example, is requesting the geographic location coordinates of Darren's iPhone. **Exhibit 12** shows a tertiary example of remote command execution on a mobile device after its been necessarily subscribed. The user interface element clearly depicts a message for an honest finder, which includes the means to contact the true owner. This message has been sent from the server in **Figure 11**, whereas (nearly identical) honest finder remote command message **1110** is displayed on the display surface **1105**. The five process steps listed in **Figure 12** mirror exactly plaintiff's embodiment; which has created examples such as the honest finder message, where it retrieves the command in **1210** from the cloud server and ultimately executes it on the mobile device at **1225**, with the corresponding result of the command messages execution **1220** being sent to the cloud server.

136. **Claim 21**. The mobile device of claim 19, wherein the command comprises a locate command.

137. Herein this claim now reinforces the mobile device side of the claims for

ambiguity. As discussed *supra* in claims 4 and 13, the locate command in this claim is shown by the plaintiff in his embodiment as the "Find Devices" command node button in **Exhibit 9**.

138. **Claim 22**. The mobile device of claim 19, wherein the instructions further cause the processor electronics to: retrieve one or more geographic coordinates from a location processor included in the mobile device; and generate a result message including the one or more retrieved geographic coordinates.

139. Herein this claim now reinforces the mobile device side of the claims for ambiguity. The mobile device may use a built-in GPS circuit, or, it may use network location approximation; depending on whether GPS circuitry is built-into the device. As discussed *supra* in claim 5, plaintiff illustrates an example indicating the location of the mobile device in a result message, listed with one or more geographic coordinates in **Exhibit 9**. The user interface element plaintiff depicts is very similar to Apple's later example in **Figure 9** and **Figure 10**, where "Jake's" iPod is shown on map **900** and **1005**, using **915** and **1025** to denote the position of the music player. Plaintiffs example in **Exhibit 9** shows an iPhone user interface element; which indicates where on the map (precisely as in **915** and **1025**) the lost device has been located—not just in one location as Apple shows, but also in two previous locations in nearby cities. It's plainly obvious the lost iPhone has been most recently located (and charted geographically) in Los Gatos, with previous locations charted in Cupertino and Saratoga. Additionally, plaintiff discloses the method for "presenting data of device location" in **Exhibit 12**, where a diagram depicts that the location data for the lost device may be presented in a web browser (as in Apple's **Figure 9** and **Figure 10** examples) or via an application on either a computer or mobile device; in addition to the possibility of a custom user interface for iPhone itself; which contains map charting. In the plaintiff's embodiment, a lost iPhone is located through a network, with the corresponding position recorded and reported by the cloud server; which, in turn may populate any of the various application or web browser examples containing geographic map charting, along with a result message. Plaintiff mentions in **Exhibit 8** under the "Message" section, "display phone location after translating the GPS location for web display. Show device in map on web app or page" as well as the linked subprocess stating, "chart lost path since last activation by user (if running again) chart previous "check-in" spots for map." The final linked subprocess states, "display device movement with charting of each check-in."

140. **Claim 23**. The mobile device of claim 19, wherein the instructions further cause the processor electronics to: establish a connection to the notification service over a wireless data

connection.

141. Herein this claim again reinforces the mobile device side of the claims for ambiguity. As discussed *supra* in claim 15, plaintiff establishes a connection to the notification service over a wireless data connection in **Exhibit 10**, whereas the "Connection Path Network" diagram shows both a cellular and wireless network connection potentially communicating with the recovery user media access control, or server. An iPhone is labeled as "iPhone Wi-Fi" and is connected to the Internet cloud, and, to the notification server.

142. **Claim 24**. The mobile device of claim 19, wherein the instructions further cause the processor electronics to: present, in response to executing the command, a message on a display of the mobile device.

143. This has been previously interrogated, especially in claims involving the specific example herein of instructions causing a message being drawn on the display of the mobile device by a processor in **Exhibit 12**; such as claim 1, claim 5, claim 9, claim 14, claim 19 and claim 22.

144. **Claim 25**. The mobile device of claim 19, wherein the instructions further cause the processor electronics to: output, in response to executing the command, an alert comprising one or more sounds to a speaker included in the mobile device.

145. In addition to the discussion *supra* concerning how plaintiff intended to play a sound on a lost mobile device with a remote command instruction from a cloud server in claim 1, a button is reserved for this purpose in the "Example Process UI" in **Exhibit 9**. This is explained at **62** and the planning for the remote command instruction to play a sound at **123**.

146. **Claim 28**. The mobile device of claim 19, wherein the command collection topic subscribed to by the mobile device includes a plurality of command nodes, where each command node corresponds to a distinct remote command type and the at least one remote command message is retrieved from one of the pluralities of command nodes.

147. Herein this claim again reinforces the mobile device side of the claims for ambiguity. As previously described *supra* in claim 9 and claim 18, plaintiffs **Exhibit 9** shows how each command node corresponds to a distinct remote command type. In this example embodiment, the plaintiff has one remote command message ready to be transmitted, which will locate the mobile device selected in the device list; further placing it in lost "discovery" mode. The plaintiff has the same, identical process in his previous IP depicted herein. **Exhibit 9** and **10** show logins, while **Exhibit 9** also shows a mobile device selected in the device list; with the "Find Devices" button representing a remote command, just as in **320** and **325** in **Figure 3**.

COUNT 2   Patent 8,881,310

System and method for remotely initiating lost mode on a computing device

148. *The '310 patent includes the following claims plaintiff invented, specifically 1, 2, 3, 5, 6, 7, 8, 11, 12, 14, 15, 16, 17 and 18 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

149. **Claim 1.** A computer implemented method, comprising: receiving, by a lost computing device, an authorized command to initiate lost mode on the lost computing device, wherein the authorized command includes contact information associated with a requesting user; initiating, by the lost computing device, lost mode on the lost computing device, wherein initiating lost mode comprises: locking the lost computing device; suppressing select functionality of the lost computing device; displaying the contact information on the lost computing device; transmitting first location data identifying an initial geographic location of the lost computing device, wherein the first location data includes a first time the lost computing device was at the initial geographic location; upon a determination that the lost computing device has traveled beyond a geographic distance from the initial geographic location, transmitting second location data identifying an updated geographic location of the lost computing device, wherein the second location data includes a second time the lost computing device was at the updated geographic location; and upon an amount of time elapsing after transmission of the second location data, transmitting third location data.

150. Plaintiff invented a method to initiate a lost “discovery” mode on a lost device, which locks the device from regular use; until such time as the true owner deactivates lost mode.

151. Plaintiff outlines this in **Exhibit 8**; in the second line of the flowchart, showing the order in which, the novel processes occur when the feature’s activated by the true owner of the phone or device. The first step highlights the true owner making the declaration the device is lost, with the second step being the user activating lost “discovery” mode.

152. Activating lost “discovery” mode’s accomplished by logging in to a cloud server with a privileged user account, that’s also enabled on the lost phone. The user may choose a button (or other user interface element) to send the instruction to both the cloud server, and, phone which activates lost “discovery” mode. Throughout the phone finding patents, Apple uses the term lost mode to denote the same thing as plaintiffs lost “discovery” mode.

153. The privileged user account is listed in the device’s contacts, which allows it to

differentiate it from others. This stops any condition from occurring where an improper or unintended user could use their cloud-based server account to initiate lost “discovery” mode, which may jeopardize the true owners use of the device, or, ability to find it if it’s actually lost.

154. The concept of using a unique user account on the device and cloud-server is shown in the “Example Process UI” in **Exhibit 9**. The interface has a field for the privileged user’s username and password to be entered, for login to the cloud server.

155. The concept of using a button (or other user interface element) to send a remote instruction to a device to initiate lost “discovery” mode after being declared lost by the true owner is also depicted in **Exhibit 9**. A radio button depicting the text “Find Devices” appears next to a list of potential devices the currently logged-in cloud server user may initiate lost “discovery” mode upon. It’s possible a user may have multiple other devices or smartphones and/or may be the privileged user for other users’ smartphones; such as minor children or employer owned devices used primarily in the workplace. The user in the example embodiment shown in plaintiffs **Exhibit 9** has three iPhones for which they’re the privileged user for.

156. Displaying the contact information of the true owner on the lost computing device is the sixth process step in plaintiffs **Exhibit 8** flowchart. An example lock-screen of the device when it’s been locked and placed into lost “discovery” modes further depicted in **Exhibit 12**.

157. The example interface element shows that the name used for registering the iPhone has been used in the “lost phone” message text; so that an honest finder would know at least the first or last name of the true owner. A telephone number that the true owner can be reached at is also depicted underneath for the honest finder, and, can be programmed dynamically by the user from the cloud-server, or, as a predefined contact number when the feature is initially enabled on the device. This allows the true owner to use a different contact number if they’re traveling, or, not able to answer the previously defined contact telephone number for the lost device, for example. Nonetheless, an instruction is sent to the lost device with this vital information; along with a visual reminder that the true owner could still use their passcode (if they enabled one) to unlock the device and disable lost “discovery” mode, without needing to only do so using the cloud-server. For many cases where a lost device is simply misplaced in the true owner’s proximity (such as falling out of a pocket and under a sofa cushion) this allows for the convenience of not needing to return to a secondary device to instruct the cloud server to end lost “discovery” mode.

158. The transmitting first location data identifying an initial geographic location of the

lost computing device is the third step listed in plaintiff's operation flowchart in **Exhibit 8**. Therein, plaintiff stated, "location data sent to Apple" and it's known from the beforementioned factual narrative that the location data was managed by the cloud-based server; both for the periodic (or on-demand) location tracking of the lost device, and, displaying such waypoints on a user interface on a map.

159. Plaintiff depicts the fifth process step in this exhibit as, "chart lost path since last activation by user (if running again, chart previous "check-in" spots for map" followed immediately by a sub-process step which states, "display device movement with charting of each check-in." This is important, because the final element of the first claim states, "wherein the first location data includes a first time the lost computing device was at the initial geographic location; upon a determination that the lost computing device has traveled beyond a geographic distance from the initial geographic location, transmitting second location data identifying an updated geographic location of the lost computing device, wherein the second location data includes a second time the lost computing device was at the updated geographic location; and upon an amount of time elapsing after transmission of the second location data, transmitting third location data." Here, the patent refers to the ability of charting progress waypoints over time, or, when the devices GPS or location-based network data indicate it's been moved a tangible distance from the previous waypoint. Herein plaintiff's doing the same thing; with the earlier example of a phone becoming lost at a train station and then moving away as a thief leaves on a departing train. The true owner has the ability to determine with relative certainty that the lost device has moved since it's last check-in—with the cloud-based server managing the location data being sent from the device at either user-defined, or, programmatic variable intervals.

160. The example interface in **Exhibit 9** depicts a visual interpretation of such a scenario. It shows that the lost iPhone is currently in Los Gatos, but, had reported a location change twice; once, along the border of Saratoga, with another reported location change on the border of Cupertino. Without knowing the particulars of how this iPhone was lost, an impartial observer who observed its true owner logging into the cloud-based server and pushing the "Find Device" button on their computer may reasonably discern the iPhone was lost around the Apple campus in Cupertino, and further, was likely inside either an honest finder or thieves vehicle travelling towards Los Gatos. The obvious determination is that the iPhone has been recovered from where it was lost and now is in-transit. This incremental updating of the location proximity of the lost device signals a firm departure from Apple's incorrect previous art assertion; which

only shows a static point, and also, doesn't help determine the location waypoints of the lost device during the time when it's not been dynamically located.

161. Often, an honest finder may intercept a lost device and begin looking for the owner in the proximate location it's discovered (which would show it was still near the origination point of the device becoming lost) or, a thief might attempt instead to move as far away as possible to avoid detection and thus increase the chances greatly of never being caught. In such an example, the thief has boarded a train and could even be in another county or state during the same duration an honest finder would instead be frantically trying to find the true owner in the train station; the clear difference of which would be visually depicted with the map overlay interface generated by the cloud-based server.

162. **Claim 2**. The method of claim 1, wherein the status data includes location data identifying the location of the computing device and is associated with a time indicating when the status data was gathered from the computing device.

163. As discussed *supra*, when the plaintiff's embodiment has been actuated (by lost "discovery" mode being enabled by the true owner authenticating to the cloud sever) the location and timestamp information Apple herein denotes as status data is recorded when the device appreciably changes location by a measurable threshold. As covered in the previous example *in re* the flowchart in **Exhibit 8**, having the time associated with a geographic waypoint is valuable. The cloud server must use time as a reference point for both authentication and identifying when a device has appreciably moved when declared lost, as must all computers. The timestamp in the corresponding cloud server log helped plaintiff's reduction to practice, for example, as he could see that the device he was requesting status information from had successfully registered its location. Moreover, the plaintiff's demonstration of this in-practice is contained in **Exhibit 9**; where the lost iPhone's shown having been in three nearby cities since it was declared lost and the "discovery" mode had been enabled via the cloud server. Lastly, **Exhibit 8** mentions under the "Progress" section of the flowchart that, "display device movement with charting of each check-in" as well as the preceding, "chart lost path since last activation by user; if running again, chart previous "check-in" spots for map." The status data is gathered from the computing device and registered with the cloud server; which can necessarily reconcile the associated collection time, especially when displayed on a map with location waypoints depicted.

164. **Claim 3**. The method of claim 1, wherein the status data is transmitted upon the remaining battery life associated with the computing device reaching a predetermined milestone.

165. The plaintiff discusses battery considerations as a subject matter expert in the narrative summary, however, plaintiff's notes in **Exhibit 8** discuss transmitting the location status data of the device based on measured power states; which themselves represent a predetermined resistance milestone. Plaintiff states, "location data is sent to Apple" before a conditional statement in the flowchart stating, "try until battery deplete" followed by "continue indefinite if power adaptor connected." Herein plaintiff demonstrates appropriate demonstration of a resistance milestone by plainly identifying a predetermined power state change.

166. **Claim 5.** The computer implemented method of claim 1, wherein initiating lost mode further comprises: presenting a user interface element on the lost computing device that is configured to enable the lost computing device to contact the requesting user based on the contact information associated with the requesting user.

167. Here again we see language describing the display of a "lost, please contact" message on the display of the device that's now been placed in lost "discovery" mode. This is depicted exactly in **Exhibit 12**; wherein an example message states that the name of the users iPhone has been lost, and, to call a pre-populated telephone number for the privileged contact.

168. Moreover, plaintiff circled for importance in the same diagram a note stating, "user record allows storage of device names and contact numbers." This user record is for the privileged user of the device, which has a contact info record containing a telephone number like any electronic vCard does. Not one difference exists between plaintiff's original implementation in his notes, and, the patent claims and diagrams submitted by Apple to the PTO.

169. The reason for this perfect match between plaintiff (and later) Apple's method of displaying a contact on the device display when in lost "discovery" mode's because no other possibility exists by which to accomplish such a task reliably; one doesn't know when they may misplace or have their device stolen—hence it's novel because otherwise this longstanding problem would've been solved with prior art. Previous art also contains no ability to pre-embed a contact for later display when the devices declared lost by its true owner.

170. **Claim 6**. A system, comprising: one or more processors; and memory containing instructions that, when executed by the one or more processors, causes the one or more processors to perform operations comprising: authenticating a requesting user operating a requesting computing device to initiate a lost mode on a computing device, where the authenticating is performed over a communications network coupled to the requesting computing device and the computing device; sending a first command over the communications network to the computing

device to initiate the lost mode on the computing device, where the lost mode includes locking the computing device or suppressing select functionality of the computing device; receiving, over the communications network, status data from the computing device, wherein the status data indicates at least a remaining battery life associated with the computing device; presenting the status data of the computing device on the requesting computing device, wherein the status data includes an estimated amount of remaining time until the computing device runs out of battery life; and sending a second command to the computing device to send status data less frequently based on the status data indicating the remaining battery life of the computing device.

171. Authentication for a requesting user operating a requesting computing device to initiate a lost mode on a computing device, where the authenticating is performed over a communications network coupled to the requesting computing device and the computing device is shown in **Exhibit 9**, where the user authenticates with the cloud server; which allows them to press the “Find Device” button after confirming the unique identifier of the lost device. This is depicted using device name in the illustration for simplicity. The communication network coupled between the requesting device, lost device and cloud server is depicted in **Exhibit 10**; showing the further interoperability between cellular and switched networks, followed by the user record mapping that’s used.

172. Sending a first command over the communications network to the computing device to initiate the lost mode on the computing device, where the lost mode includes locking the computing device or suppressing select functionality of the computing device, is discussed broadly in **Exhibit 8**; followed by showing an “Example Lock Screen When Lost” user interface in **Exhibit 12**. Plaintiff discloses in **Exhibit 11** that, “we could lock the device and invalidate the true passcode while privileged mode is in-use.” Actuating such functionality from the cloud server by the true owner is clearly depicted in the example interface.

173. Receiving, over the communications network, status data from the computing device, wherein the status data indicates at least a remaining battery life associated with the computing device; presenting the status data of the computing device on the requesting computing device, wherein the status data includes an estimated amount of remaining time until the computing device runs out of battery life; and sending a second command to the computing device to send status data less frequently based on the status data indicating the remaining battery life of the computing device was discussed *supra*, *in re* the battery / adaptor power state change and predetermined resistance threshold in **Exhibit 8**.

174. **Claim 7.** The computer implemented method of claim 1, further comprising: terminating lost mode upon receiving a correct password, wherein terminating lost comprises: unlocking the lost computing device; restoring suppressed functionality of the lost computing device; and removing the displayed contact information associated with the requesting user.

175. Here, the method for terminating lost mode upon receiving a correct password is no different than the interface in plaintiffs **Exhibit 12**, where the bottom of the display showing contact information associated with the requesting user features an unlock function. This function comprises the word “unlock” and depicts the space for the devices 4-character password, or passcode as it’s referred to with iOS devices. While iOS devices now use a 6-character passcode, until very recently, they utilized a 4-character numerical password.

176. The lost “discovery” mode plaintiff invented also may be terminated by using the cloud-server account that was used to initially find the device.

177. More importantly, plaintiff declared under “Handling Device When Stolen” in **Exhibit 11** an entry for the fourth operational note that, “we must allow the device to be unlocked due to accidental enabling, or, the phone being found.” Both usage cases where a true owner might recover their device using plaintiff’s invention are specifically disclosed. The true owner could terminate lost “discovery” mode either by entering the devices password on the device lock screen itself, or, programmatically by sending a cancel signal using the privileged cloud-based server account.

178. Plaintiff felt it was important to allow either method to terminate lost “discovery” mode; because the true owners secondary device used to initiate the devices discovery may be in a completely different physical proximity than the phone is, or, may not belong to them, but was borrowed for the express purpose of locating and retrieving their lost device by signing into the cloud-based server and initiating lost “discovery” mode. In these situations, it wouldn’t otherwise be possible for the true owner to use their recovered device without also having access to either the previous (or a new) secondary device, which could be used to then terminate lost “discovery” mode. For example, if a true owner initiated lost “discovery” mode for their iPhone from a friend’s house or nearby business establishment and then retrieved the device from an honest finder, they’d be unable to use the device for a potentially uncertain amount of time.

179. The biggest impact arises not just from temporary convenience, as in the usage case of a device owner who has no access to a secondary computing device for an extended period of time. Some users have work environments like this, but still need and/or are permitted

to have a computing or telephony device. The delivery driver or farmer who may initiate lost "discovery" mode at the start of their day may then find their smartphone and have no way to use it for the entire day. Plaintiff had concerns about the practical usability and safety concerns on such scenarios arising from the use of an invention to find a device that's become lost causing a potential greater impact to the user than if they had never used it. Take for example, the delivery driver whose iPhone has fallen out of their pocket and becomes lost under the seat of their vehicle. The driver will find their iPhone during their shift and then be unable to use it the entire day; whereas if they hadn't used the computer at the distribution center to find it with lost "discovery" mode before leaving, they could simply unlock it normally.

180. Finally, the usage case exists wherein a lost device has no passcode enabled, meaning either an honest finder or thief could obtain access to its content and memory instantly. While most users of electronic devices have a password to lock their device, they may not and thus a special usage case exists; wherein such users need additional protection to keep their data safe, as well as securing its safe return. The device becomes locked in such cases and thus must be unlocked by terminating lost "discovery" mode using a cloud-based server account. This is the only exception case where the password cannot be entered on the device itself to terminate lost "discovery" mode when found by the true owner. Plaintiff carefully planned for both usage cases.

181. Further, plaintiff also disclosed the ability to invalidate even the correct password for a device if desired by the true owner as an additional remote command option. The first sentence of **Exhibit 11** reads, "we could lock the device and invalidate the true passcode while privileged mode is in-use." Whether the implementation of the invention overall opts to forcibly enforce this, let the user decide with a preference, or, not require this added restriction altogether is immaterial to the solution of the overall problem. Plaintiff had considered this as an extra layer of security; for intelligence agencies and tech employees, for example. The scenario this prevents is an honest finder providing or selling the device to a thief or rogue actor with malicious intent, who could then subject it to a vulnerability exploit compromising the passcode and allowing full access to the device, since lost "discovery" mode would effectively now also be disabled.

The other usage case is for the paranoid user who yet uses a simple password for their device, such as the "1, 2, 3, 4, 5" password used in the film *Spaceballs* to secure an entire planets oxygen supply in its atmosphere. By requiring only an unlock via the cloud server, a true owner can thus guarantee their device is genuinely in their possession. Given the password "1, 2, 3, 4, 5, 6" is used for consumer devices by 23.2 million people and "1, 2, 3, 4, 5, 6, 7, 8, 9" is used by 7.7

million in just the UK alone, this is significant. [5] Apple iOS devices changed from 4-digit to 6-digit passcodes in 2015 with iOS9. Johns Hopkins Information Security Institute cryptographer Matthew Green stated cracking a four-digit pass code can be done in 6.5 minutes (the longest is 13 minutes). A six-digit pass code is better, averaging of 11 hours, with a maximum of 22 hours; this is using an iPhone decryption device which defeats Apple's imposed delay between unsuccessful login attempts. [6]

182. **Claim 8.** A lost computing device, comprising: a processor; and a memory containing instructions that, when executed, cause the processor to: receive an authorized command that lost mode be initiated on the lost computing device, wherein the authorized command includes contact information associated with a requesting user; initiate lost mode on the lost computing device, wherein initiating lost mode comprises: locking the lost computing device; suppressing select functionality of the lost computing device; displaying the contact information on the lost computing device; transmitting first location data identifying an initial geographic location of the lost computing device, wherein the first location data includes a time the lost computing device was at the initial geographic location; upon a determination that the lost computing device has traveled beyond a geographic distance from the initial geographic location, transmitting second location data identifying an updated geographic location of the lost computing device, wherein the second location data includes a second time the lost computing device was at the updated geographic location; and upon an amount of time elapsing after transmission of the second location data, transmitting third location data.

183. Here, the basic premise of the overall invention is being repeated, but with respect to the functions the lost device executes to operate. The cloud-server interaction still must occur, but here Apple describes solely the interactions which the smartphone makes with its discrete processor unit. The bulk of communication is simply response-centric from the cloud server; however, instructions are sent which only the lost device can perform on its own, even after being set in lost "discovery" mode with its initial instruction. Locking the phone if it doesn't already have a password enabled, displaying the contact info on the devices display which also indicates the device's lost, sending approximate location data when the device (using an accelerometer or GPS, for instance) has moved appreciably, or, when asked on-demand from the user via the cloud

[5] Is 123456 your password? Then you need to change it!
https://www.bbc.co.uk/newsround/48002968

[6] iPhone Security: Your 6-digit passcode is no longer safe
https://www.newsweek.com/iphone-security-your-six-digit-passcode-no-longer-safe-891401

server, and ending lost “discovery” mode when either the successful password is entered, or, an instruction is sent from the cloud server by the true owner are primary examples.

184. For the purposes of this claim, it’s mostly for added ambiguity of the methods declared in the application during the examination period, and, for later infringement prosecution. In this sense, it should be considered moot overall by the Court as duplication of the original claim; being cognizant plaintiff still illustrates with his lab notebook exhibits that such interaction from the lost device is expected and necessary for his invention to succeed. Like a human, if the lost device is either incapable or unwilling to communicate, they remain so difficult to find that the original problem of reliably locating and retrieving a lost smartphone remains unsolved.

185. This is why (among other reasons) plaintiff has subprocess routines or steps listed after the principal function or claim has been activated in **Exhibit 8**. The first example shows an event loop depicting a lost device sending location data at measured intervals if the battery threshold has reached a predetermined “low” amount of resistance, to send without restriction if the level is sufficient, or, the devices plugged in to an adaptor, irrespective of resistance. All these conditional functions occur on the device, obviously, and also mimic the related condition of the device’s battery being fully depleted, or, the rare edge case of the processor or memory experiencing hardware failure when the user declares that it’s been lost. Since a user could have a fully depleted battery on their device (to the extent it would not activate) and then lose it, this distinction is important. The true owner could then attempt to initiate lost “discovery” mode from an instruction sent by the cloud server; however, it’d similarly fail in the same manner, as the processor isn’t available to interpret and execute instructions for regular use, let alone receiving responses from a remote server. Similarly, if the cellular or network apparatus on the lost device couldn’t connect to anything, such as in a heavily concreted building or jungle island, the processor, battery and other hardware may be working fine and could interpret instructions from the cloud server, however, the lack of cellular or network connectivity may render the lost device unreachable; no different than if it’s power supply failed, or, the processor had stopped working.

186. **Claim 11**. A method comprising: receiving, by a computing device, a first command over a communications network to initiate a lost mode on the computing device; locking the computing device or suppressing select functionality of the computing device; determining a remaining battery life of the computing device; and sending, over the communications network, status data indicating at least a remaining battery life of the computing device, wherein the status data includes an estimated amount of remaining time until the

computing device runs out of battery life.

187. As previously interrogated *supra*, the plaintiff's method for initiating lost "discovery" mode over a communication network is detailed in **Exhibits 8-10**. Locking the lost device to suppress its functionality is detailed in **Exhibit 11**; with an example user interface showing the locked device in **Exhibit 12**. The battery resistance threshold discussion *in re* the lost device is mentioned in **Exhibit 8**.

188. **Claim 12.** The lost computing device of claim 8, wherein the instructions further cause the processor to: present a user interface element configured to enable the lost computing device to contact the requesting user based on the contact information associated with the requesting user.

189. Herein is a repeat of the previous. The cloud-server may send instructions, but ultimately the lost device must use its own processor and memory to determine whether the privileged user matches the user account which has initiated the lost "discovery" mode request from the cloud server. A user could give a device to another person, for example, and not logout or otherwise remove themselves as the privileged contact for the device. While the true owner may be different than the person who's been given access to use the device, only the true owner as registered on the device itself can successfully make lost "discovery" mode requests.

190. In this sense, the device acts as a security audit layer against not only cloud server users, but, even somebody with trusted internal access to the cloud server itself, as plaintiff had. This is why plaintiff could perform a reduction to practice only with his own devices. Even a person with special access to the server that allowed for seeing every device which had ever contacted it could not then use plaintiff's invention (as correctly implemented) to than initiate lost "discovery" mode as a prank on a friends iPhone, for instance.

191. This is why **Exhibit 10** is helpful and obvious for proving plaintiff's previous invention of this claim; it shows a relational block diagram entitled "User Record Mapping" that features a lost iPhone on the left, with another computer logged-in on the right as the same contact as the requesting privileged user. Both boxes which contain the word "user record" are joined, along with the boxes for SSL, which indicate to those skilled in the art that an encrypted tunnel has been established with matching certificates for the same user; both on the lost device, and, making the authenticated request from a computer connected to the cloud server. The diagram finishes by illustrating a "login" box for both entities, which is known by those skilled in the art as having validated a successful challenge response between both a requesting party and a

security authority, such as when a ticket has been successfully exchanged using Kerberos.

192. **Claim 14.** The lost computing device of claim 8, wherein the instructions further cause the processor to: terminate lost mode upon receiving a correct password, wherein terminating lost mode comprises: unlocking the lost computing device; restoring suppressed functionality of the lost computing device; and removing the displayed contact information associated with the requesting user.

193. This claim's largely been properly interrogated *in re* plaintiffs prior inventorship in preceding statements.

194. **Claim 15.** A non-transitory computer-readable medium containing instructions that, when executed by a lost computing device, cause the lost computing device to: receive an authorized command that lost mode be initiated on the lost computing device, wherein the authorized command includes contact information associated with a requesting user; and initiate lost mode on the lost computing device, wherein initiating lost mode comprises: locking the lost computing device; suppressing select functionality of the lost computing device; displaying the contact information on the lost computing device; and transmitting first location data identifying an initial geographic location of the lost computing device, wherein the first location data includes a first time the lost computing device was at the initial geographic location; upon a determination that the lost computing device has traveled beyond a geographic distance from the initial geographic location, transmitting second location data identifying an updated geographic location of the lost computing device, wherein the second location data includes a second time the lost computing device was at the updated geographic location; and upon an amount of time elapsing after transmission of the second location data, transmitting third location data.

195. Herein this claim describes largely the cloud server interaction, depicting them as authorized commands. The preceding arguments are similarly germane here, as the cloud server has been established as being necessary and proper for a secure implementation of the invention; otherwise impropriety might overcome the solution for this longstanding problem.

196. It should be noted that the chart entitled "Presenting Data of Device Location" in **Exhibit 12** depicts the cloud server at the top of the device hierarchy, with lost devices connecting to it via a cellular or switched network connection, but, the instructions from the true owner are sent from an application or web browser on another device. Since communication from the cloud server is necessary to send instructions to the lost device, this illustration is important. The absolute necessity of a cloud server to accomplish the goal of the invention is evident in the

diagram. The lines drawn from each devices web browser or application to control the methods of the patent sending and interpreting instructions from the lost device on the left are joined together with the cloud server; with a single line than extending to the lost device through one of the communication mediums transport layers.

197. An important distinction here is that Apple also opted to implement plaintiff's invention exactly as he depicts in the beforementioned **Exhibit 12** block diagram. Apple allowed both a web connection to be used to login to the cloud server and actuate the feature, as well as developing a standalone application for iOS devices, such as iPhones and iPads. This allows a standalone application to instead handle the security provisioning of login to the cloud server, which in-turn makes a request to the lost device—instead of having to login to a web page using a mobile web browser application. The overall implementation plan appears to be an exact photocopy of **Exhibit 12**; however, it is solely the plaintiff's novel method and apparatus.

198. **Claim 16**. A computing device comprising: one or more processors; and memory containing instructions that, when executed by the one or more processors, cause the one or more processors to perform operations comprising: receiving, by the computing device, a first command over a communications network to initiate a lost mode on the computing device; locking the computing device or suppressing select functionality of the computing device; determining a remaining battery life of the computing device; sending, over the communications network, status data indicating at least a remaining battery life of the computing device, wherein the status data indicating a remaining battery life associated with the computing device includes an estimated amount of remaining time until the computing device runs out of battery life; receiving, by the computing device, a second command to send status data less frequently based on the status data indicating the remaining battery life of the computing device.

199. Herein this claim relates specifically to the necessary instructions being executed on a computer or cloud server and not the lost device; for added application ambiguity. Each sub method in this claim has previously been interrogated against the plaintiff's evidence.

200. **Claim 17**. The computing device of claim 16, wherein the status data includes location data identifying the location of the computing device and is associated with a time indicating when the status data was gathered from the computing device.

201. Again, this claim's for application ambiguity and has already been previously interrogated in claims 2, 7 and 12, *supra*.

202. **Claim 18**. The computing device of claim 16, wherein the status data is

transmitted upon the remaining battery life associated with the computing device reaching a predetermined milestone.

203. Once again, this claim's for application ambiguity and has already been previously interrogated in claims 1, 3, 6, 8 and 11, *supra*.

COUNT 3 Patent 9,104,896

Remotely initiating lost mode on a computing device

204. *The '896 patent includes the following claims plaintiff invented, specifically, 1-8, 11, 12, 16 and 17 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

205. **Claim 1.** A method, comprising: authenticating a requesting user operating a requesting computing device to initiate a lost mode on a computing device, where the authenticating is performed over a communications network coupled to the requesting computing device and the computing device; sending a first command over the communications network to the computing device to initiate the lost mode on the computing device, where the lost mode includes locking the computing device or suppressing select functionality of the computing device; receiving, over the communications network, status data from the computing device, wherein the status data indicates at least a remaining battery life associated with the computing device; presenting the status data of the computing device on the requesting computing device, wherein the status data includes an estimated amount of remaining time until the computing device runs out of battery life; and sending a second command to the computing device to send status data less frequently based on the status data indicating the remaining battery life of the computing device.

206. Authenticating a requesting user operating a requesting computing device to initiate a lost mode on a computing device has been previously discussed. **Exhibit 10** highlights the plaintiff's authentication method which Apple declares, depicting the user record mapping required to securely initiate a request instruction to enter lost "discovery" mode.

207. Whereas "authenticating is performed over a communications network coupled to the requesting computing device and the computing device" is concerned, **Exhibit 10** depicts a diagram entitled "Connection Path (Network)" which shows a cellular network and switched Internet network sharing connectivity with an iPhone connected using only a cellular connection, an iPhone using only wireless Internet, a cloud server, and finally, the true owners computer used to initiate lost "discovery" mode. The illustration's clear that irrespective of communication

method, that both cellular and switched networks can be used both for authentication, and, for the invention’s overall operation. **Exhibit 10** contains the transmission mediums and authentication methods together because they’re essential for operation and thus inextricably linked.

208. Finally, we turn to intelligently managing the battery to expel location updates and the ability to light the display; to reveal contact data for an honest finder. “…status data from the computing device, wherein the status data indicates at least a remaining battery life associated with the computing device; presenting the status data of the computing device on the requesting computing device, wherein the status data includes an estimated amount of remaining time until the computing device runs out of battery life; and sending a second command to the computing device to send status data less frequently based on the status data indicating the remaining battery life of the computing device.” **Exhibit 8** contains the overall process steps, with their respective requirements. In the sub-process for “Location Data Sent to Apple” contains clear language stating, “try until battery deplete” and “continue indefinite if power adaptor connected.” These two conditional logic methods are collectively labeled “Power States” on the left.

209. Given plaintiff was previously awarded a ‘631 patent for battery technology he invented at Apple, it’s abundantly clear that he understood this overall process framework to not be encompassing of the details of all the corresponding processes. Plaintiff intended to further refine his battery usage power savings algorithm; once he could receive information about the firmware Apple uses in iOS to monitor the batteries health and current charge state. Plaintiff had planned a selective power state change already identified in the **Exhibit 8** block diagram, which shows he understood that putting the lost device in a lower power state was necessary for proper success. Plaintiff established the need to manage the battery intelligently when the device had been declared lost, which a plain read of **Exhibit 8** reveals to one not skilled in the art. Otherwise, plaintiff wouldn’t have specifically mentioned “power states” or otherwise included an event loop for power management.

210. Such power state changes prevent the scenario of a battery with a low amount of resistance being able to overdraw from the processor; when responding to CPU directives sent from third party applications. A scheduled task or push email synchronization session could, in-itself be enough to cause the processor to enter a higher performance state, which than extends resistance at a far steeper rate than simple linear depletion. It also helps protect the chances of the true owner recovering the device if the battery has been fully depleted and shuts down; whenever an honest finder or thief alike connected a power charging source, the instruction which keeps the

device locked, displays the contact info and negotiates location data relay will continue to remain active. This keeps a thief from simply waiting for the battery to fully deplete, and then, restore the device to normal operation with a new privileged user when it does become again operational. It also can greatly extend the time the lost device can report its location data to the cloud server.

211. Plaintiff worked on the initiatives necessary to reduce the PPW (performance per watt) for Mac computers to radically extend overall battery life. Further, plaintiff was the subject matter expert for power management for the support of Mac computers worldwide. His experience solving difficult problems with battery logic and power management is well known to many inside Apple. It explains why his first and only Apple patent (see **ECF No. 34**) is related to detecting system errors with battery health. It's preposterous to suggest plaintiff wasn't aware of the need for intelligent power management, after himself working on all of Apple's power savings initiatives in some capacity. When a case existed where batteries appeared to operate outside normal operation, they were sent to plaintiff for forensic analysis; to determine if an exception case existed in software which could be unduly exercising the battery. Sudden depletion events, while rare in the aggregate are part of using advanced polymer battery chemistries, such as lithium-ion; used exclusively in all Apple products since the Intel processor transition. Even earlier PowerPC-based architectures were potential targets for sudden depletion, which would still render a lost device quickly unreachable for the true owner. Ultimately, one not skilled in the art can easily discern that plaintiff was one of the best possible engineers to be brainstorming any support technology using batteries.

212. **Claim 2.** The method of claim 1, wherein the status data includes location data identifying the location of the computing device and is associated with a time indicating when the status data was gathered from the computing device.

213. Herein this claim has been previously interrogated–*in re* location data transmission, making it necessarily apply. The new addition here's the timestamp; reflected with the transmission of location data on the lost device to the cloud server. Showing waypoints established and transmitted by the lost device (either dynamically, or, in a synchronization of recent locations updated in batches periodically when a network connection so allows) can be easily depicted using a map; as opposed to just a timestamp in the event log for that user on the cloud server. The example UI in **Exhibit 9** depicts a visual interpretation of such a scenario. The lost iPhone reported its location in Cupertino, where it was likely lost; followed by briefly passing through Saratoga where the device reported location data, followed by its final location being

reported in Los Gatos. While not implemented, a timestamp could also be affixed to each location on a map with a tooltip; further minutia plaintiff considered when solving the problem of reconciling a lost device that was moving.

214. Thus, a relevant point—why plaintiff illustrated three lost device locations on the map example, as retracing the movement of the lost device during a thief's chain of custody is undeniable when applying reasonable doubt to a suspect in a prosecution. Having a screenshot or log file with a location and timestamp is significant forensic evidence, helping a prosecutor or Court decide when a true owner's device was in a given location. It's also valuable for a situation where the true owner has been abducted or kidnapped, but, their device is in possession of the suspect, or alternatively, if they had their smartphone, but, didn't know their own location.

215. Plaintiff illustrates this clear understanding of waypoints over time being valuable to the invention, and, how its inextricably linked to a timestamp in **Exhibit 11**. Under the section entitled "Handling Device While Stolen" the plaintiff states in #5, "Optional upload to law enforcement database? This would allow cops to track and find lost device without user interaction." Under subsection E, plaintiff states, "forensic data from server could be valuable in court or missing persons cases." It's known that the first 24 hours a missing person's missing can often be most critical for identifying a suspect. Since all location data here is joined with a timestamp, it makes this summation necessarily relevant for establishing inventorship, as plaintiff was clearly considering the ramifications that time identification can have when something is declared lost. Plaintiff noting the forensic data was stored on the server further validates its identical timestamp usage for this embodiment.

216. Google has used plaintiff's invention to even capture a device movement when they haven't been declared lost in a database server called *Sensorvault*, which allows law enforcement to present a warrant to request location-based data based on time. Large-scale location monitoring is useful to Google because it allows the company to target advertisements based on where consumers regularly travel, as well as to assess the ads' effectiveness. It is useful to law enforcement because, "*it can allow investigators to view the movement of all devices within a specific area over a specific period of time to track down suspects or witnesses in cases that otherwise might go cold*." [7] The relationship between timestamps and plaintiff's lab notebook

[7] Google can see where you've been. So can law enforcement. https://www.washingtonpost.com/opinions/google-can-see-where-youve-been-so-can-law-enforcement/2019/04/15/90542fa6-5fbe-11e9-bfad-36a7eb36cb60_story.html?utm_term=.201ebfbe8d72

entries discussing law enforcement captive mode in **Exhibit 11** couldn't be more well-defined factually; particularly as related to the claims herein.

217. The practice was first used by federal agents in 2016, [8] according to Google employees, and first publicly reported last year in North Carolina. [9] It has since spread to local departments across the country, including in California, Florida [10] , Minnesota [11] and Washington. This year, one Google employee said, the company received as many as 180 requests in one week. Google declined to confirm precise numbers. The new orders, sometimes called "geofence" warrants, specify an area and a time period, and Google gathers information from *Sensorvault* about the devices that were there. It labels them with anonymous ID numbers, and detectives look at locations and movement patterns to see if any appear relevant to the crime. Once they narrow the field to a few devices they think belong to suspects or witnesses, Google reveals the users' names and other information. Investigators who spoke with *The New York Times* said they had not sent geofence warrants to companies other than Google, *and, Apple said it did not have the ability to perform those searches*. Google would not provide details on Sensorvault, but Aaron Edens, an intelligence analyst with the sheriff's office in San Mateo County, who has examined data from hundreds of phones, said most Android devices and some iPhones he had seen already had this data available from Google. [12]

218. Since plaintiff's idea was novel, if Apple had properly joined him as an inventor to this patent, his disclosures certainly would have revealed this important use of timestamp-based location-data for law enforcement purposes. This idea has a discrete method and purpose. As such, Google would clearly be infringing upon plaintiff's novel idea and work product as an Apple employee, directly related to the problem of lost or stolen device recovery; this would have instead put Apple in the role of plaintiff, instead of as a defendant in this action.

219. The owner of a trade secret has the rights to possess the idea and its physical embodiments, to limit its disclosure to others, and, to contract for the terms of its use by others.

[8] In the Manner of Search of Information Regarding Accounts Associated with Certain Location and Date Information, Maintained on Computer Servers Controlled by Google, Inc. (2). Case No. 1-19-MJ188 https://int.nyt.com/data/documenthelper/758-austinaffidavit2/d448fe5dbad9f5720cd3/optimized/full.pdf#page=1

[9] To find suspects, police quietly turn to Google. Were you near the Raleigh fire? Detectives may already know. https://www.wral.com/Raleigh-police-search-google-location-history/17377435/

[10] Clerk & Comptroller Receipt of Florida Search Warrant, Log #126, Case No. MI-01-0130 https://int.nyt.com/data/documenthelper/764-fdlelocationsearch/d448fe5dbad9f5720cd3/optimized/full.pdf#page=1

[11] How did the police know you were near a crime scene? Google told them. https://www.nytimes.com/interactive/2019/04/13/us/google-location-tracking-police.html

[12] Tracking Phones, Google Is a Dragnet for the Police https://www.nytimes.com/interactive/2019/04/13/us/google-location-tracking-police.html

Lariscey v. U.S. 949 F.2d 1137, 1141 (Fed. Cir. 1991). An originator may enforce these rights through several legal theories, including trade secret misappropriation, breach of contract, breach of trust or confidence, and, quasi-contract. Since plaintiff has documented evidence showing his method before Google adopted such a process and was an Apple employee at the time, Apple could've been the assignee and thus owner of the additional trade secret. A defendant who uses another's trade secret is liable; even if he modifies or improves upon the trade secret, as long as the substance of the defendant's use is derived from the originator's secret, as in *Forest Laboratories, Inc. v. Pillsbury Company* 452 F.2d. 621, 625 (7th Cir. 1971).

220. Accordingly, Apple not only could be receiving patent licensing revenue from Google if they hadn't committed intentional nonjoinder of plaintiff, but, they could've implemented a similar system for law enforcement themselves, which may have saved countless lives given worldwide OS device popularity and sales; particularly among the younger population, who's more prone to abduction and kidnapping.

221. Plaintiffs friend was murdered on August 5, 2000. Both his computer and telephone were stolen, later making the suspect eligible for the California death penalty; since a robbery was committed during the murder. If plaintiff had developed his idea much earlier in time, the true suspect could have been caught before he fled to Texas. It would've prevented a mutual friend from being wrongfully incarcerated for nearly a year for his murder; before DNA evidence finally corroborated his story and led to the discovery of both the suspect and victims' possessions in Texas. This event shaped plaintiffs' further refinement of his novel idea, which is why he included specific mention of special law enforcement use.

222. Plaintiff was impressed to read in the same April 13, 2019 *New York Times* article beforementioned, a man was arrested for murder based on location data with time stamps from Google, which showed his phone nine months earlier at the spot of a murder. After a week in jail, the suspect was exonerated and released, as police determined a man who sometimes used his car was the murder suspect. It took nearly a year for plaintiffs' friend to be released from jail on suspicion of murdering a mutual friend. If plaintiff's invention had instead occurred while he was in college, it would've shown his friend was not present at the murder scene the evening it took place, and further, had indeed driven there to meet the victim the following day, when he discovered the body.

223. Even if the PTO had declined to patent this additional methods process innovation, Apple still would've had plaintiffs' disclosures. A decent possibility exists that Apple would have

developed such a method for law enforcement before Google, leading to less tragic losses of life from abduction, kidnapping and murder. While Google used parallel innovation after Apple filed thirteen patents for related methods (plaintiff invented originally at Apple) as a basis for *Sensorvault*, the burden lay on Apple for intentionally filing thirteen applications it already knew (in writing) was invented by another employee.

224. The public this has a choice when deciding whether to purchase an Apple or Google device for their loved ones. If they purchase an Apple device, there's no chance law enforcement may lawfully learn the location and timestamps from their device since being reported lost. If they purchase a Google device, an excellent chance exists that the device can be located and either the loved one, suspect or both will be recovered before more foul play may occur. Given this choice is predicated solely by Apple's ignorance and intentional malice in ensuring plaintiff was nonjoinder of the phone-finding patents as direct causation, it presents a very conscious choice for the Court and customers about reasonableness and responsibility. It wasn't enough for Apple to punish its employee and plaintiff for having a good idea, it was necessary to thus punish the public, as consumers of Apple's products.

225. **Claim 3.** The method of claim 1, wherein the status data is transmitted upon the remaining battery life associated with the computing device reaching a predetermined milestone.

226. Herein this claim has been previously interrogated–*in re* location data transmission upon remaining battery life as in plaintiffs **Exhibit 8**, making it necessarily apply.

227. **Claim 4.** The method of claim 3, wherein the milestone is the remaining battery life reaching a predetermined percentage of total battery life.

228. Herein this claim has been previously interrogated, making it necessarily apply. Plaintiffs beforementioned narrative discussing the relation between sudden depletion at **211** and dynamic processor directives causing an uncertain total discharge time at **210** bear weight here.

229. **Claim 5.** The method of claim 3, wherein the milestone is the remaining battery life reaching a predetermined amount of remaining time left until the computing device runs out of battery life.

230. Herein this claim has been previously interrogated–*in re* the remaining battery life reaching a predetermined amount of remaining time left until the computing device runs out of battery life, making it necessarily apply. Beforementioned battery life considerations of significant detail needn't be repeated again here.

231. **Claim 6.** A system, comprising: one or more processors; and memory containing

instructions that, when executed by the one or more processors, causes the one or more processors to perform operations comprising: authenticating a requesting user operating a requesting computing device to initiate a lost mode on a computing device, where the authenticating is performed over a communications network coupled to the requesting computing device and the computing device; sending a first command over the communications network to the computing device to initiate the lost mode on the computing device, where the lost mode includes locking the computing device or suppressing select functionality of the computing device; receiving, over the communications network, status data from the computing device, wherein the status data indicates at least a remaining battery life associated with the computing device; presenting the status data of the computing device on the requesting computing device, wherein the status data includes an estimated amount of remaining time until the computing device runs out of battery life; and sending a second command to the computing device to send status data less frequently based on the status data indicating the remaining battery life of the computing device.

232. Herein this claim deals with the same methods and evidence beforementioned, but, concerns the processor of one device communicating with the processor in the cloud server.

233. **Claim 7.** The system of claim 6, wherein the status data includes location data identifying the location of the computing device and is associated with a time indicating when the status data was gathered from the computing device.

234. Herein this claim has been previously interrogated–*in re* associating a timestamp with the identified location of a lost device. The previous discussion regarding both plaintiffs' multiple waypoints of a found device on a map established in **Exhibit 9**, and, the potential law enforcement usage plaintiff highlighted (which necessarily requires associating a timestamp with the identified location of the missing device) in **Exhibit 11** is substantial towards proving the plaintiffs original inventorship. It's impossible to overcome the *conception* established between law enforcement usage, and, the element of location data associated with a timestamp.

235. **Claim 8.** The system of claim 6, wherein the status data is transmitted upon the remaining battery life associated with the computing device reaching a predetermined milestone.

236. Herein this claim has been previously interrogated–*in re* remaining battery life reaching a predetermined milestone. Such a milestone is calculated when the device has been put into lost "discovery" mode; if the battery life remaining is unsatisfactory for maintaining the best life versus reduced performance. This is generally accomplished by temporarily disabling or throttling the application layer of the device to use only the amount of memory necessary to retain

a pointer (or reference point) as to what functions and operations were running when the low-power state was entered; so they can be recovered when resistance has returned to a nominal threshold. The goal of reducing Apple's portable computers PPW impact was a substantial part of plaintiff's power management support duties. As such, plaintiff had intimate knowledge of *how*, *when* and *why* to engage power state changes, as supported by Intel's processor directives; with the same being true given plaintiff's expert knowledge of batteries and intelligent power management with earlier PowerPC architecture, by Freescale Semiconductor.

237. **111**, **165**, **173**, **185**, **210** and **236** (*supra*) discuss methods of planning an intelligent resistance curve the plaintiff had planned, which go into far greater detail than the minimal description afforded by Apple in this '896 application.

238. The patent plaintiff holds that Apple filed in 2008 entitled "Detection of System Battery Errors" was attached as a deposition in **ECF No. 34**. In a related example, a battery error may be detected before the operating system is loaded onto the computing device. In another example, the error may be detected when the computing device is waking from a reduced power mode. Herein plaintiff illustrates a novel and superior ability to manage and query the battery life of a computing device; far exceeding the basic logic needed in this patent claim to establish a reasonable threshold for beginning power savings, and then, enforcing it until resistance has reached a certain pre-determined value.

239. In detailed description 0021 of the application, it states: "When computing device **200** is turned on, cycled (turned off and on), or reset (e.g., reset signal sent to processor **202**), the computing device initiates a pre-boot sequence. The pre-boot sequence is a hardware function that prepares computing device **200** such that an operating system or other software applications may be loaded onto the computing device. Basic Input/Output System (BIOS) instructions **224** may define the functions of the pre-boot sequence and are initiated (or loaded) before the operating system is loaded. In general, BIOS instructions **224** are the firmware code executed by computing device **200** during the pre-boot sequence. Examples of BIOS instructions **224** include Extensible Firmware Interface, Open Firmware, and Linux BIOS. A pre-boot sequence may include operations such as initializing processor **202**, main memory **206**, and various input/output devices. Additionally, the pre-boot sequence may also include a self-test. This self-test may include verifying processor **202**, verifying main memory **206**, and identifying errors with system battery **228**. As explained in more detail below, computing device **200** may also be configured to conduct a self-test when the computing device detects a change in current supplied by system

battery **228**."

240. It's evident to those both skilled and unskilled in the art that plaintiff already had developed a novel method to query and test batteries in computing devices; including when already in a low-power state, or even, in situations where no useable operating system was present on the device. Even if the mass storage failed on a computer, plaintiff can still apply such advanced troubleshooting algorithms to its battery.

241. Those unconvinced beyond doubt as to the applicability of the battery claims in the phone finding patents (collectively) must take notice of detailed description 0023:

"Examples of parameters that may be tested include an expansion parameter, a battery memory parameter, an overcharged parameter, an expected life parameter, a cell imbalance parameter, a connection parameter, and other parameters. It should be appreciated that the expansion parameter defines a swelling of the system battery, for example, from temperature variations. The battery memory parameter defines a loss of maximum energy capacity caused by the repeated recharging of the system battery. The overcharged parameter defines whether the system battery is charged over the maximum energy capacity. *The expected life parameter defines a measure of an estimated length of functionality of the system battery*. The cell imbalance parameter defines a measure of the voltage balance of the system battery."

242. While not disclosed by Apple, plaintiff had clearly intended to utilize a simple expected life parameter; to define a measure of estimated length of functionality for the lost device's battery. This would then reliably return a threshold determination; whether to implement power savings of the lost device, or, do nothing and let the current performance profile remain. When plaintiff alludes in **Exhibit 8** to, "continue indefinite if power adaptor connected" it's a clear indication that the typical power profile associated with normal device use would remain; as no situation would reasonably then ensue causing an appreciable enough resistance loss to necessitate a power state change, to reduce overall consumption and conserve resistance.

243. If Apple had rightfully asked for plaintiff's disclosures and not intentionally left him nonjoinder, it's clear to one unskilled in the art that the claims in the phone finding patents (particularly this one) *in re* battery life would have contained significantly more accurate and helpful information. A plain read indicates neither the misjoinder inventors, nor, application author have any tangible understanding of how battery technology works. Very loose, poorly defined statements like those in this claim are used. This is because plaintiff wasn't present to explain how his invention was to both be properly implemented, and, described in the patent

application disclosures.

244. The misjoinder inventors fail in the application to state *how* or *why* conditions affecting the remaining battery life play a role in measuring the available resistance and calculating an appropriate power state. No descriptions made to differentiate what causes a battery to have resistance woes, or, what measures have been made to ensure the initial calculation isn't devoid of any number of issues potentially affecting a battery assembly itself. A battery that's reaching the threshold of failure may certainly not show any indication if a simple voltage tests performed when lost "discovery" mode's been enabled; as plaintiff assumes from the poor documentary evidence in the patents claims. This causes an imperfect estimation that plaintiffs previous battery innovations solve, meaning that if a lost device was suffering from cell imbalance or swelling, it'd be necessarily prudent to begin a reduced power state immediately, irrespective of the resistance reported by the battery firmware.

245. Apple also doesn't explain the process for communicating with a battery using firmware, which must be down with iOS devices and Mac computers; because the operating system cannot interpret data from batteries without the firmware which manages the cells themselves. Again, we see the pitfalls of Apple's misjoinder and nonjoinder; by intentionally not joining plaintiff from the phone finding patents, the applications contain unsure statements about batteries, as the work product they claim relies on plaintiff's information they neither possess nor understand. When one works from a facsimile and wrongfully pronounces it their own, it's impossible to properly describe and elaborate on that which wasn't wholly provided. The battery narratives are so poor, they suggest even a detailed explanation from plaintiff (such as this) would still be confusing and difficult for them to describe in a patent application.

246. The added notion the misjoinder inventors simply worked from a photocopy of plaintiff's lab notebook entries (contained herein as exhibits) begins to seem plausible, especially in-light of the battery discussion in the claims. Plaintiff showed his notebook to several people, attached the entries to a Radar bug, and, they could've been copied without his knowledge; while he wasn't at his cubicle. The building plaintiff was based at the time (De Anza 3) had the highest theft rate of any Apple corporate building. It was then assumed by the same iOS software engineers in the group who later claimed the phone finding patents, so, the possibility for additional badge access being available to those who knew the misjoinder inventors is high, as plaintiffs' team was among the last to move. Even after the resulting move to a different building, plaintiff had personal and work items removed from his cubicle he still cannot reconcile. It's

instructive to note plaintiff only had a cubicle in an open floor plan and did not have a proper lab to secure materials of importance or high confidentiality.

247. In detailed description 0025, it describes Figure 4. The power management ability and understanding of plaintiff further removes doubt *in re* the battery threshold claim. “During a pre-boot sequence or when computing device **200** detects a change in current from system battery **228**, power management circuit **220** tests one or more parameters of the system battery. Power management circuit **220** is an integrated circuit that is configured to manage the power of computing device **200**. For example, power management circuit **220** may control backlighting, hard disk spin down, power modes, charging system battery **228**, and other power management operations. A System Management Controller, a System Management Unit, and a Power Management Unit are examples of power management circuit **220**. The misjoinder inventors don’t disclose problems with managing the power of a battery in a lost computing device, describe what the power management circuit is responsible for and can limit to conserve power, or finally, what elements of iOS communicate with battery firmware.

248. The messaging necessary to, “set remaining battery life associated with the computing device reaching a predetermined milestone” is also curiously absent. In detailed description 0027, it states: “The message **404** is a value that describes the parameter. For example, message **404** may be a hexadecimal key that describes a particular parameter. In another example, message **404** may be a binary flag that describes a particular parameter. Additionally, in some examples, the message may be constructed to describe multiple parameters. Power management circuit **220** transmits message **404** to processor **202**. In an example embodiment, processor **202** receives message **404** from power management circuit **220** and stores the message in register **402**, which is a memory available on the processor.” A message must be sent to the battery firmware from the PMU the misjoinder inventors don’t understand; so that the remaining battery life associated with the device may reach a predetermined milestone. This claim says that the status message is sent, but doesn’t indicate what is sending the message, what the parameters of the message are, how the message has been calculated, or finally, where the message is being sent. The unfortunate errors of omission from the original inventors nonjoinder continue.

249. **Claim 11.** A method comprising: receiving, by a computing device, a first command over a communications network to initiate a lost mode on the computing device; locking the computing device or suppressing select functionality of the computing device; determining a remaining battery life of the computing device; and sending, over the

communications network, status data indicating at least a remaining battery life of the computing device, wherein the status data includes an estimated amount of remaining time until the computing device runs out of battery life; and receiving, by the computing device, a second command over the communications network to send status data less frequently based on the status data indicating the remaining battery life of the computing device.

250. Plural instances here exist which have already been interrogated, detailing the plaintiff's eligibility for inventorship *in re* this claim. Herein this claim simply reinforces that the lost device is capable of receiving instructions to perform the tasks already discussed.

251. **Claim 12.** The method of claim 11, wherein the status data includes location data identifying the location of the computing device and is associated with a time indicating when the status data was gathered from the computing device.

252. Plural instances here exist which have already been interrogated, detailing the plaintiff's eligibility for inventorship *in re* this claim. Herein this claim simply reinforces that the cloud server is capable of receiving location data identifying the location of the computing device, which is associated with a timestamp when gathered from the lost device.

253. **Claim 16.** A computing device comprising: one or more processors; and memory containing instructions that, when executed by the one or more processors, cause the one or more processors to perform operations comprising: receiving, by the computing device, a first command over a communications network to initiate a lost mode on the computing device; locking the computing device or suppressing select functionality of the computing device; determining a remaining battery life of the computing device; sending, over the communications network, status data indicating at least a remaining battery life of the computing device, wherein the status data indicating a remaining battery life associated with the computing device includes an estimated amount of remaining time until the computing device runs out of battery life; receiving, by the computing device, a second command to send status data less frequently based on the status data indicating the remaining battery life of the computing device.

254. Plural instances here exist which have already been interrogated, detailing the plaintiff's eligibility for inventorship *in re* this claim. Herein this claim simply reinforces that lost devices which have multiple processors also can claim the same methods previously disclosed, as well as indicating both the cloud server and lost device have processors interpreting instructions from each other to accomplish the necessary transactions.

255. **Claim 17.** The computing device of claim 16, wherein the status data includes

location data identifying the location of the computing device and is associated with a time indicating when the status data was gathered from the computing device.

256. Plural instances here exist which have already been interrogated, detailing the plaintiff's eligibility for inventorship *in re* this claim. Herein this claim reinforces (with further duplication) that location data associated with a timestamp is interpreted by lost devices with multiple processors, and, the cloud server.

257. It's worth note plaintiff mentions language in his operational flowchart in **Exhibit 8** exploring this exact concept. Under the "Progress" section, plaintiff states, "chart lost path since last activation by user (if running again, chart previous "check-in" spots for map" followed by the next process step, which states, "display device movement with charting of each check-in." Clearly, a timestamp's a necessary element and obvious indicia of plaintiff's notes here, as no processor can determine or differentiate a path since the last request without using the measure of time. A timeline establishing where the lost device has reported its location data is established here by plaintiff; necessarily relying upon timestamps, as disclosed in the claim.

258. A lost device may not otherwise "know" or be able to reconcile the time which has passed since it was moved if it wasn't connected temporarily to a communication network. Lastly, units of time defined in portions of seconds is the only method a processor of any kind may interact, interpret or further audit any given event, sequence or task; irrespective of the programing logic or method used to control the instructions. The conversion of time values into assembler or hexadecimal allows for efficient processor execution of instructions, while allowing the resulting value to be displayed in a human-readable format (i.e. 1, 2, 3, 4, etc.) next by an application layer on a computing device or webpage, such as the "Find my iPhone" feature.

259. The example map embodiment in **Exhibit 8** (showing a lost iPhone transmitting its location data from three discrete locations) could only be thus possible using timestamps for the location and status data, consistent with this claim. The imperfect understanding in the application *in re* calculating the time when location data is processed suggests an unsure understanding by the implementors—necessary for creating such a feature in embedded devices as the claim and invention here represents. Time is the only means of both programmatically defining and organizing events within occurrence.

260. The clock cycle allows a familiar map result for a found object to be more valuable; as a valid time when the device was either in-transit, not in-transit, or, transmitting location and status data is an important distinction; which can occasionally transcend convenience

and prosecution investigations, by helping ensure the physical wellbeing of the owner prevails. Without knowing when a device was present in a geographic location, the value of such data remains of little use. Thus, the stated acknowledgments in the plaintiff's notes as related to claims involving time demonstrate overall *conception* and inventorship.

COUNT 4 Patent 9,706,032

Device locator disable authentication

261. *The '32 patent includes the following claims plaintiff invented, specifically 1, 4, 6, 7, 8 and 10 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

262. **Claim 1**. A method comprising: entering, by a mobile device, an activation operating mode, wherein the mobile device is configured to enable one or more functions in the activation operating mode, and wherein on the mobile device, user-erasable content and settings including one or more user-configured security settings have been erased, the one or more user-configured security settings including a setting for user authentication that specifies that network user credentials stored on a server shall be used for authentication after the user-erasable content and settings have been erased; while in the activation operating mode, transmitting a request for user account information to the server, the request being associated with a hardware identifier of the mobile device, the hardware identifier uniquely identifying the mobile device to the server, wherein: the user account information was stored on the server before the one or more user-configured security settings were erased on the mobile device, and the user account information includes user credentials that are identifiable by the hardware identifier; presenting, on the mobile device, a user interface for configuring the mobile device, the user interface including a challenge for authenticating a user of the mobile device based on the user account information received from the server; and in response to receiving an input through the user interface responding to the challenge, activating the mobile device.

263. Plaintiffs embodiment while in the activation operating mode described in claim 1 is part of lost "discovery" mode; whereas transmitting a request for user account information to the server, the request being associated with a hardware identifier of the mobile device, the hardware identifier uniquely identifying the mobile device to the server, wherein: the user account information was stored on the server before the one or more user-configured security settings were erased on the mobile device, and the user account information includes user credentials that are identifiable by the hardware identifier; presenting, on the mobile device, a

user interface for configuring the mobile device, the user interface including a challenge for authenticating a user of the mobile device based on the user account information received from the server; and in response to receiving an input through the user interface responding to the challenge, activating the mobile device. **Exhibit 8** describes the activation of lost "discovery" mode, which permits the remote execution of special tasks on the device which's been declared lost by the true owner. The "Example Process UI" in **Exhibit 9** depicts the hardware identifier uniquely identifying the mobile device to the cloud server, using user credentials which are identifiable by the hardware identifier; presenting, on the mobile device, a user interface for configuring the mobile device, the user interface including a challenge for authenticating a user of the mobile device based on the user account information received from the server, as depicted in **Exhibit 12's** example lock screen.

264. It's instructive to note that the challenge for authenticating the true owner of the lost device is represented in the unlock illustration; whereas a 4-digit passcode may be used to successfully unlock the device, but, predicated on the cloud server's authority. It's helpful to note plaintiff also discloses the ability for the true owner to proactively lock the device in such a manner that the valid passcode will no longer be accepted; until such time as an instruction is sent from the cloud server by the true owner to allow the correct passcode to once again be activated. **Exhibit 11** details handling the missing device during the case of being stolen. The first consideration plaintiff lists states, "we could lock the device and invalidate the true passcode while privileged mode is in-use." This distinction's important, as the overall purpose of this patent concerns locking the device to prevent a thief from necessarily erasing or repurposing it; as well as potentially disabling the lost "discovery" mode that's been enabled via the cloud server.

265. **Claim 4.** The method of claim 1, wherein the user account information stored on the server is different from user credentials that are local to the mobile device, and activating the mobile device comprises enabling one or more functions that are different from security features.

266. Herein the differentiation between accounts which may be actively used on a lost device conflicting with the privileged account of the true owner used to authenticate with the cloud server, and, send an instruction to enable lost "discovery" mode's discussed. Differentiation between accounts which may be used to provision a mobile device when activated with the carrier is further described, but poorly.

267. The "Example Process UI" in **Exhibit 9** shows a field for the true owner of the lost device to enter the account username and password credentials for the privileged account also

present on the device. While the device list shows the lost iPhone highlighted, this user selection ensures the cloud server accepts the correct credentials on-demand for the correct device that's chosen. Otherwise, the username and password credential fields would not appear in plaintiffs example interface embodiment; as the user could then just select a desired device in a list and press the "Find Device" button. This distinction is very important given this claim.

268. Moreover, plaintiff presents a "User Records Mapping" block diagram in **Exhibit 10** especially relevant to this claim. While explained in beforementioned greater detail at **57**, **125**, **153**, **155**, **189** and later, at **269**, **272**, **273**, **281**, **287**, **288**, **290**, **313**, **315** and **334**, the diagram demonstrates how the user record on both the lost device and a secondary device (being used to find the other) both must share the same privileged user account.

269. Security features of an account are different, but, in the case of maintaining a secure connection between the requesting device, cloud server and lost device, they must each login using the same account, and, the accounts must support an accepted and compliant security protocol. This is why plaintiff shows an SSL connection above the login entry for both client and server sides of the user record mapping. Using a secure account is a choice and Apple's display figures also don't demonstrate secure account connections like plaintiffs. This means that while the claims are necessarily in agreement for *conception* and inventorship, that as implemented, a third-party could intercept and monitor the lost device finding session. While **Figure 2** shows an authentication module, it does not demonstrate that it's secure as plaintiff additionally does. While the process is identical, the distinction is important, because a similar account's necessary to ensure the corresponding certificate is valid for encryption. The requirement of having the same privileged user account also ensures all transactions are encrypted and not subject to intrusion from a weak endpoint. Otherwise, a more complicated method of validating certificates becomes necessary for the same guarantee. This important distinction is not in the application.

270. Given Apple has before allowed the superuser account to be enabled without a password [13] (the worst possible security vulnerability possible) it's no surprise the misjoinder inventors similarly don't understand basic security; assuming it's a black box (like batteries) that's magically self-aware and devoid of the need for disclosure, planning or understanding. The Apple photocopiers failed to include a secure implementation in this application; they could see

[13] MacOS bug lets you log in as admin with no password required. Here's how to protect yourself until Apple patches bafflingly bad bug. https://arstechnica.com/information-technology/2017/11/macos-bug-lets-you-log-in-as-admin-with-no-password-required/

the SSL boxes in plaintiffs' notes, but, omitted it from lack of understanding the importance.

271. **Claim 6.** The method of claim 4, wherein the hardware identifier includes a hash generated based on one or both of a media access control (MAC) address and an international mobile equipment identity (IMEI) of the mobile device.

272. This claim involves storing a record on the cloud server to differentiate which devices correspond to the privileged user accounts associated with them, albeit not well disclosed or revealed. Plaintiff demonstrates his inventorship to this claim easily, as the diagram in **Exhibit 10** displays both the MAC address and IMEI as discreet fields, connected to the cloud server. Since bubbles are used above these fields to indicate transmission over a cellular or switched network to the cloud server and lost devices, it cannot be construed as a network transport illustration; else the bubbles would've been unnecessary and not also included.

273. Further, plaintiff uses the block diagram further below to indicate that both a lost iPhone and Mac have their MAC and IMEI unique identifying data registered and known, which is actuated by the cloud server above; denoted as "Recovery User Media Access Control" and not to be confused with a Mac computer located to the left. The "Recovery User" has a hash of either the unique MAC or IMEI addresses associated with privileged user accounts, which is stored on the cloud server for impartial, secure validation.

274. **Claim 7**. The method of claim 4, comprising limiting functionality of the mobile device until the mobile device is activated.

275. This method's depicted in plaintiffs **Exhibit 12**, where a lost mobile device has been locked by the true owner; using a cloud server to reconcile the unique hardware identifier and account credentials of the privileged user. The exhibit stresses "Example Lock Screen When Lost" above the user interface depiction, which shows visibly that the device is lost and locked. As discussed *supra*, plaintiff mentions, "we could lock the device and invalidate the true passcode while privileged mode is in-use." It's clear to one unskilled in the art that plaintiff's description of locking the device when declared lost by the true owner matches exactly with limiting the functionality of a mobile device, as espoused in this claim. Lastly, plaintiff contributed further than Apple in terms of additional methods of limiting functionality, suggesting in **Exhibit 11** a possible, "law enforcement captive mode to emulate the privileged user, [which] would only be for murder or kidnapping, but possible." This embodiment goes further, as it allows a law enforcement authority to use a warrant to cause Apple to activate the lost "discovery" mode of the missing persons mobile device; which, by design allows for limiting functionality. Law

enforcement could thus also limit functionality of the lost device while searching for its true owner under a public safety exception.

276. **Claim 8**. A non-transitory computer-readable medium comprising code that, when executed by a processor, causes a device to perform operations including: entering, by a mobile device, an activation operating mode, wherein the mobile device is configured to enable one or more functions in the activation operating mode, and wherein on the mobile device, user-erasable content and settings including one or more user-configured security settings have been erased, the one or more user-configured security settings including a setting for user authentication that specifies that network user credentials stored on a server shall be used for authentication after the user-erasable content and settings have been erased; while in the activation operating mode, transmitting a request for user account information to the server, the request being associated with a hardware identifier of the mobile device, the hardware identifier uniquely identifying the mobile device to the server, wherein: the user account information was stored on the server before the one or more user-configured security settings were erased on the mobile device, and the user account information includes user credentials that are identifiable by the hardware identifier; presenting, on the mobile device, a user interface for configuring the mobile device, the user interface including a challenge for authenticating a user of the mobile device based on the user account information received from the server; and in response to receiving an input through the user interface responding to the challenge, activating the mobile device.

277. Plaintiff has demonstrable evidence to support inventorship of this claim; based on the need to transmit a request for user account information to the server, the request being associated with a hardware identifier of the mobile device, the hardware identifier uniquely identifying the mobile device to the server, and, wherein: the user account information was stored on the server before the one or more user-configured security settings were erased on the mobile device, and the user account information includes user credentials that are identifiable by the hardware identifier; presenting, on the mobile device, a user interface for configuring the mobile device, the user interface including a challenge for authenticating a user of the mobile device based on the user account information received from the server; and in response to receiving an input through the user interface responding to the challenge, activating the mobile device.

278. The transmission of information to the server, with the request being associated with a unique hardware identifier of the device is depicted in plaintiffs **Exhibit 9**, where the “Example Process UI” shows a list of devices that a user may request the cloud server execute a

lost “discovery” mode instruction, which is demonstrated by showing devices with different names. **Exhibit 10** shows a “User Record Mapping” crucial for the cloud server to authenticate the true owner of the lost mobile device, as it contains also the associated device hardware identifiers. While the cloud server keeps a record of such unique hardware identifiers, it’s then presented in user interfaces using the name the user has assigned to the device, for practicality sake. Using the 15-digit IMEI or 12-digit MAC address of the device (as depicted in **Exhibit 10**) is not easily discernable to the owner. 990000862471854 (or 00-14-22-01-23-45 for a computer) is much harder for a user to reconcile than the human readable “Darren’s iPhone” shown in **Exhibit 9**. Note the example locked user interface element in **Exhibit 12** could accept the true owner’s valid passcode to unlock the device; if the cloud server had not suspended authentication completely, however, it may reinstate passcode usage—if the true owner made this request.

279. **Claim 10.** A system comprising: a processor; and a non-transitory computer-readable medium comprising code that, when executed by the processor, causes the processor to perform operations including: entering, by a mobile device, an activation operating mode, wherein the mobile device is configured to enable one or more functions in the activation operating mode, and wherein on the mobile device, user-erasable content and settings including one or more user-configured security settings have been erased, the one or more user-configured security settings including a setting for user authentication that specifies that network user credentials stored on a server shall be used for authentication after the user-erasable content and settings have been erased; while in the activation operating mode, transmitting a request for user account information to the server, the request being associated with a hardware identifier of the mobile device, the hardware identifier uniquely identifying the mobile device to the server, wherein: the user account information was stored on the server before the one or more user-configured security settings were erased on the mobile device, and *the user account information includes user credentials that are identifiable by the hardware identifier; presenting on the mobile device, a user interface for configuring the mobile device, the user interface including a challenge for authenticating a user of the mobile device based on the user account information received from the server; and in response to receiving an input through the user interface responding to the challenge*, activating the mobile device.

280. Herein is conclusion of previous discussion and interrogation *in re* unique hardware identifiers and user accounts being used to authenticate with a cloud server authority. The cloud server is protecting the true owner’s device by only recognizing as an authority the

previous privileged users account credentials, and, unique hardware identifier. This helps prevent a thief from bypassing or changing security settings on the device, since the instruction to enable lost "discovery" mode has been received and executed. While this seems obvious to one unskilled in the art, the cloud server must be established as an authority over the lost devices when enabled.

281. Plaintiff describes this claim in **Exhibit 8**, where the overall method and process workflow mention the usage of "device privilege" mode; or, the scheme wherein the universal hardware identifier is matched to the privileged user account associated with the device on the server, creating a mode capable of locking the device. Such modes may disable the ability of a thief to erase the device, or, otherwise disable the lost "discovery" mode; so, the device may be repurposed or sold without the true owner's consent.

282. Beforementioned discussion concerning relevance of user account authentication contained at **125**, **163**, **171**, **206**, **269**, **278** and later at **302** and **340** support plaintiffs inventorship for this claim; as well as use of unique hardware identifiers at **280** and later at **313** and **315**. The user interface in **Exhibit 9**, **Exhibit 10** and **Exhibit 12** further support plaintiffs claim.

COUNT 5 Patent 9,763,098

Bypassing security authentication scheme on a lost device to return the device to the owner

283. *The '98 patent includes the following claims plaintiff invented, specifically 1, 2, 4, 9, 10, 13 and 14 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

284. **Claim 1**. A non-transitory machine readable medium storing a program for execution by at least one processing unit, the program for bypassing device security protections to communicate with a privileged contact of a secure device, the program comprising sets of instructions for: displaying, while the device is in a locked mode in which a plurality of services are unavailable on the device, a selectable user interface (UI) item on the device for enabling a person to operate the device to communicate with a privileged contact while the device is in the locked mode; determining whether a secured router that provides restricted access for lost devices is available; upon the selection of the UI item, displaying a list of available Wi-Fi networks to select a Wi-Fi network while the device is in the locked mode, wherein a Wi-Fi network for the secured router is displayed with an indication that indicates that the secured router provides restricted access for lost devices, wherein selection of the Wi-Fi network for the secured router allows a connection to the Wi-Fi network without a password when the device is in the locked mode; and initiating a communication, through a Wi-Fi network of the list of available Wi-Fi

networks while the device is in the locked mode, with the privileged contact from a list of privileged contacts stored on the device.

285. **Exhibit 12** depicts visual evidence of plaintiffs inventorship in this claim. In the "Example Lock Screen When Lost" user interface element, the text for the locked device screen (when lost "discovery" modes been enabled) says, "(User's iPhone is LOST! Please call (123-456-7890)" with a corresponding unlock field; which accepts the device passcode. Pressing a telephone number using iPhone has always caused it to be automatically dialed; even if the number is presented by another Apple or third-party application, such as Contacts, Notes, etc. Any honest finder who views the plaintiffs example lock screen user interface element would thus know it was possible to press the telephone number on the display where it's drawn; even if they were unskilled in the art, and, even if the device was operating in a limited functional capacity by design—thus such messaging appearing when lost "discovery" modes active. It helps to reinforce the otherwise evident; that dialing the telephone number listed in this user interface element is the only function an honest finder may perform.

286. **Claim 2.** The non-transitory machine-readable medium of claim 1, wherein the program further comprises a set of instructions for receiving a selection of the privileged contact from the list of privileged contacts before initiating the communication.

287. Receiving the message *in re* a selection of the privileged contact from the list of privileged contacts before initiating the communication is depicted in the "Example Process UI" contained in **Exhibit 9**. A list of three devices configured with a corresponding privileged user account appears; with username and password credential fields contained above, which must be manually entered in the plaintiffs implementation—to ensure a thief who accesses a device already logged-in as the true owner couldn't then locate or disable their other devices, and also, as a means to ensure the privileged user account and unique hardware identifier hasn't changed since last established on the cloud server. Finally, a "Find Device" radio button is located in the same embodiment, which transmits the set of instructions for receiving a selection of the privileged contact from the list of privileged contacts before initiating the communication. Since the cloud server must audit also the validity of both the privileged user account password and potential changed parameters as just discussed, it's necessary for the instructions to be interpreted by the cloud server first; before initiating attempts to communicate with the lost device. Since such operations occur extremely quickly, the instructions sent first to the cloud server (and even latency while contacting and locating the device) may be imperceptible from the user perspective;

as soon as the "Find Device" buttons depressed, it may appear to them that an instantaneous connection has been made with the lost device.

288. The illustration between user records for the true owners privileged user, and, their other devices matching (before a secure login occurs) is shown in **Exhibit 10**. The login box displayed at the bottom of each device in the diagram is not accidental, as it displays conditional authority being necessary before a request to put a device in lost "discovery" mode may occur.

289. **Claim 4**. The non-transitory machine-readable medium of claim 3, wherein the program further comprises sets of instructions for: receiving a selection of another privileged contact from the list of privileged contacts; and displaying a different list of selectable communication mechanisms for selection by the person.

290. Plaintiff disclosed two embodiments in **Exhibit 9** showing a third-party application being used to actuate and manage lost "discovery" mode. The first example depicts a lost iPhone that's been located; with the previous location data for three past movements since being declared lost listed on a map. The second example shows the login screen; wherein a true owner may select a device to declare lost and enter their privileged user account credentials into the cloud server. **Figure 12** of this patent depicts a very similar user interface as the plaintiffs showing the lost iPhone on a map. **Figure 13** depicts the location with a timestamp identifier between a previous location since being located, which mirrors plaintiffs second embodiment. Additionally, plaintiff mentions in **Exhibit 12** that, "user record allows storage of device names and contact numbers" whereby contact numbers can be associated as contacts, or, privileged contacts—for purposes of this claim.

291. Plaintiff has a chart entitled "Presenting Data of Device Location" in **Exhibit 12**, which plainly states that two of the potential implementation options are third-party applications. The first says "App on Devices" and the second says "App on Computers" with appropriate lines connecting them to the cloud server for proper operation. This obviously applies both to overall device-finding applications, and, using a telephony application to dial the privileged contact which appears on the lock screen. For example, the true owner of an iPhone may have a third-party VOIP application set as default for making outgoing calls; as opposed to the telephone application Apple provides by default for making and receiving calls.

292. **Claim 9.** A device comprising: a set of processing units; and a memory storing a program for execution by at least one of the processing units, the program for providing a graphical user interface (GUI) for bypassing device security protections to communicate with a

contact of the device, the GUI comprising: a display area for displaying UI objects; and a selectable UI object for enabling a person to operate the device to communicate with a privileged contact while the device is in a locked mode in which a plurality of services are unavailable on the device and in which the device must be unlocked before making the plurality of services available, wherein a selection of the UI object causes a display of a list of privileged contacts while the device is in the locked mode, wherein the list of privileged contacts comprises a set of automatically generated privileged contacts when no contacts were previously designated as privileged contacts, and wherein a selection of a displayed privileged contact causes an initiation of a communication with the selected privileged contact while the device is in the locked mode using a communication mechanism assigned to the selected privileged contact.

293. Plaintiff plainly describes a graphical user interface (GUI) for bypassing device security protections to communicate with a contact of the device in **Exhibit 8**; where he mentions under “Display Lost Message on Phone Using “Device Privilege” Mode, If User Wishes To Do So” that, in addition to a “Default Lost Message” which could appear on the lost devices GUI, a “Custom User Defined Message” or even additionally “Custom UI to Differentiate From Provider Text, Etc.” could also be used—to help make it easier for an honest finder to see that the device owner was trying to alert them of a method of contact. This is as opposed to being dismissed as a notification or text message which may have been received since the device was lost, and, might discourage an honest finder from bothering to read the true owners lock message.

294. A display area for displaying UI objects is disclosed by plaintiff in **Exhibit 12**, wherein it states, “Example Lock Screen When Lost” and depicts a user interface object displaying a message for an honest finder of a lost computing device to call them at a provided number, as well as offering the option to unlock the device with a password.

295. A selectable UI object for enabling a person to operate the device to communicate with a privileged contact while the device is in a locked mode in which a plurality of services are unavailable on the device and in which the device must be unlocked before making the plurality of services available, wherein a selection of the UI object causes a display of a list of privileged contacts while the device is in the locked mode, wherein the list of privileged contacts comprises a set of automatically generated privileged contacts when no contacts were previously designated as privileged contacts, and wherein a selection of a displayed privileged contact causes an initiation of a communication with the selected privileged contact while the device is in the locked mode using a communication mechanism assigned to the selected privileged contact is

again disclosed by plaintiff in **Exhibit 12**, wherein it states, “Example Lock Screen When Lost” and depicts a UI object displaying a message for an honest finder of a lost computing device to call them at a provided number, as well as offering the option to unlock the device with a password. In the example text, plaintiff states that, “[The Device Owner’s Name] iPhone is LOST!” on the first line. The second line of text states, “Please call (123-456-7890)” with the bottom portion of the UI allowing the device to be unlocked with the password of the true owner. Moreover, plaintiff has circled text for emphasis next to his UI example, which states, “User record allows storage of device names and contact numbers.” This is related to the true owner having a designated contact for use as a communication proxy when the devices declared lost.

296. This eliminates the need for the true owner to need to call or send text messages to the device; which an honest finder might accidentally interpret as messages intended for the true owner and disregard them, particularly if they hear periodic audible noises from the device. In some cases, that might prompt an honest finder to shut down the lost device, so that the response noises stop disturbing them.

297. It’s necessary to call to particular attention that plaintiffs example interface makes a conclusive point of illustrating that the device has been locked and cannot be further used without entering a 4-digit passcode, as was standard for iPhone then. The messaging to call a contact of the honest finder (above the unlock input mechanism) is the only other item appearing on the device, and, in the example. Its clear Apple devices do not (even now) allow operation of the device when the lock screen’s present. Since no lock screen indicated that the device was lost before plaintiff’s example interface existed, it’s clear to one unskilled in the art that the device has been locked as a result of being declared lost by the true owner.

298. Recognizing that other countries have longer telephone numbers, and, that a user could additionally have a name using more characters than average, plaintiff circled text for emphasis next to his interface example which states, “Need room for LOC, Longer #” and refers to localization. When localization occurs to an interface element, it’s given optional text versions based on languages supported by the device. The natural elements of languages coupled with the subjective attributes of name length and local telephone rules mean significant variance may exist outside English examples. This helps reconcile programmatic interface issues, such as how to handle truncation for a name or telephone number that’s too long to fit on the required line. Such distinctions are critical here; truncating the final portion of a name is immaterial in comparison to instead omitting digits from the telephone number. Such an issue might cause the invention to

still fail when found by an honest finder—representing a genuine tragedy.

299. **Claim 10.** The device of claim 9, wherein the list of privileged contacts are enabled through the device's stored contact list.

300. Herein is ambiguity for the application, that’s been previously interrogated. The plaintiffs privileged contact being enabled from the device’s stored contact list is quite evident from **Exhibit 8** and **Exhibit 12**.

301. **Claim 13.** A non-transitory computer readable medium storing a program which when executed by at least one processing unit presents a graphical user interface (GUI) for bypassing device security protections to communicate with a contact of the device, the GUI comprising: a display area for displaying UI objects; and a selectable UI object for enabling a person to operate the device to communicate with a privileged contact while the device is in a locked mode in which a plurality of services are unavailable on the device and in which the device must be unlocked before making the plurality of services available, wherein a selection of the UI object causes a display of a list of privileged contacts while the device is in the locked mode, wherein the list of privileged contacts comprises a set of automatically generated privileged contacts when no contacts were previously designated as privileged contacts, and wherein a selection of a displayed privileged contact causes an initiation of a communication with the selected privileged contact while the device is in the locked mode using a communication mechanism assigned to the selected privileged contact.

302. **Exhibit 12** shows a pre-populated telephone number on a touchscreen phone device. Given touching a telephone number in contact lists or other applications causes that number to be dialed by the baseband connection, there’s no reason to otherwise suggest that tapping the example telephone number plaintiff depicts in **Exhibit 12** would not dial that telephone number; provided the device was capable of making telephone calls. Thus, plaintiff’s righteous contention that **Exhibit 12** shows a selection of a displayed privileged contact causing an initiation of a communication with the selected privileged contact, while the device is in locked mode; using a communication mechanism assigned to the selected privileged contact. It’s known from previous establishing, inspection of the narratives, and, visual UI element in **Exhibit 12** that the device has been declared lost, and, placed into a state whereas a plurality of normal services isn’t available until successful authentication occurs by the true owner.

303. **Claim 14.** The non-transitory computer readable medium of claim 13, wherein the list of privileged contacts are enabled through the device's stored contact list.

304. As discussed at **168**, **290**, **295**, and later at **347**, plaintiff circled text for emphasis next to his UI example in **Exhibit 12**, which states, “User record allows storage of device names and contact numbers.” Herein, the privileged contacts represent contact numbers on the device. Directly above this text is a box which says, “Custom UI on Phone” showing its being controlled by an instruction from the cloud server, which triggers the device presenting the contacts info.

COUNT 6 Patent 9,979,776

Remotely locating and commanding a mobile device

305. *The ‘776 patent includes the following claims plaintiff invented, specifically 1, 2, 3, 4, 5, 6, 8, 10, 11, 13, 16 and 18 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

306. **Claim 1**. A computer-implemented method of remotely commanding a mobile device, the method comprising: by a computing device: receiving input uniquely identifying a mobile device, wherein the mobile device is remotely located from the computing device; displaying a plurality of remote commands available to be performed by the mobile device, wherein the displaying includes identifying at least one of the plurality of remote commands as enabled for execution by the mobile device and at least one other of the plurality of remote commands as disabled for execution by the mobile device; receiving input selecting a remote command from the plurality of displayed remote commands; generating a remote command message instructing the mobile device to execute the selected remote command; and transmitting the remote command message to a server to communicate the remote command message to the mobile device.

307. The method to remotely command a mobile device is explained generally in **Exhibit 8**, whereas the features operative process is explained. The true owners’ devices are uniquely identified and managed using the unique IMEI or MAC address, which is retained by the cloud server; as illustrated in **Exhibit 9** and also in **Exhibit 10**. The manner and purposes for using these two nomenclatures for a unique hardware identifier was previously explained in the ‘32 patent, particularly at **223** and **228**. A plurality of remote commands is demonstrated in **Exhibit 9**, whereas there are four total commands depicted; with one currently being enabled. This is because the true owner has authenticated with the cloud server (which also handles notifications) already, which has populated a device list showing three iPhones uniquely registered to them. Once the “Find Devices” button is then pressed, the other remote command

buttons become active and usable by the true owner; as the notifications for remote command messaging has now been established between the cloud server and lost mobile device.

308. **Claim 2**. The computer-implemented method of claim 1, wherein communicating the remote command message to the mobile device comprises the remote command message being retrieved from the server by the mobile device.

309. Communicating a remote command message to the mobile device produces a resulting response, but only if the device is reachable via a communication network. The enabling instruction thus allowing other remote command messages to be transmitted to the lost device from the server is the "Find Devices" button depicted in **Exhibit 9**. This allows the other remote commands to be displayed and executed by the true owner, which are transmitted from the cloud server to the lost device when the user presses one of the buttons in the example user interface.

310. **Claim 3**. The computer-implemented method of claim 2, wherein the remote command message is retrieved from the server by the mobile device in response to a notification from the server indicating that the remote command message is available for retrieval from the server.

311. Herein the remote command message that's part of lost "discovery" mode is sending instructions to the mobile device that's been declared lost by the true owner; but, only after they've first authenticated with a cloud server, which issues the remote command messages to lock the device, and, display contact information—providing a means for an honest finder to otherwise contact the true owner. This is explained in the overall feature workflow in **Exhibit 8** but demonstrated visually in the "Example Process UI" in **Exhibit 9**; whereas no remote command messages are yet available for lost device retrieval from the cloud server. Once the "Find Device" button has been pressed by the true owner, remote command messages can then be made available for retrieval from the cloud server.

312. **Claim 4**. The computer-implemented method of claim 2, wherein the transmitting the remote command message to a server comprises transmitting the remote command message to a notification server for publication in a command node included in a command collection topic uniquely subscribed to by the mobile device, where the command collection topic is one of a plurality of command collection topics hosted on the notification server and the command node is one of a plurality of command nodes included in the command collection topic, wherein the notification server notifies the mobile device that the remote command message is available for retrieval in the command node, and wherein the notification server transmits the remote command

message to the mobile device only in response to the mobile device accessing the command node.

313. The "User Record Mapping" in **Exhibit 10** demonstrates how the cloud server in plaintiff's original embodiment also constitutes the services of the notification server here; for publication of such things as the device list, unique hardware identifier and a corresponding privileged user account. The server maintains these data classifications as discussed previously to enforce the exclusive control policy over the mobile device; when it's been declared lost and the true owner uses the cloud server to enable lost "discovery" mode; as explained in **Exhibit 8** and depicted operating in **Exhibit 10**. This results in a locked device, which can have remote events executed, such as displaying a message for an honest finder in **Exhibit 9**. No other user may access the command node in plaintiff's embodiment; only the privileged user account that's provisioned on the server—to process remote events for devices with the unique hardware identifiers having already been registered by the true owner.

314. **Claim 5**. The computer-implemented method of claim 1, further comprising: displaying a selectable list of mobile devices associated with a remote management account, the selectable list including information uniquely identifying each mobile device.

315. As discussed *supra*, a selectable list of mobile devices associated with unique hardware identifiers, which are registered to the privileged user account is denoted here by Apple as a remote management account. The plaintiffs example user interface element in **Exhibit 9** depicts this so precisely (using three iPhones) it's a direct copy. **Figure 1** depicts the Internet in **105**, with a cloud server connected at **115**, with connected computers at **110** and **130** and iPhones at **125** and **120**; all the devices are connected via the network, to reinforce that the same user is authenticated to each device using their privileged account. Plaintiff has depicted the same chart in his **Exhibit 10**, denoting a cellular and switched network clouds merged; with two iPhones, one computer and the server. The only problem's that one not skilled in the art can easily discern that Apple's chart appears to be an identical copy, but, was submitted to the PTO seven years after plaintiff had already disclosed it to many Apple employees. Many unskilled in the art have already made the determination to plaintiff that it's obvious Apple copied plaintiffs work; in the same manner as a guilty child taking an examination who didn't study and instead copies the work of the child seated next to them, calling it their own. A jury shall reach the same conclusion; Apple's continued intentional dishonesty cannot overcome a basic test of the evidence. Unfortunately for Apple and its misjoinder inventors, the patent theft games finally concluded after several years; at great personal and professional cost to one of its esteemed former engineers

and plaintiff.

316. **Claim 6**. The computer-implemented method of claim 5, further comprising: indicating, for at least one mobile device included in the selectable list of mobile devices, whether at least one mobile device is online.

317. Plaintiff shows in **Exhibit 9** an "Example Process UI" that clearly depicts three iPhones by unique device name which're online; as the "Find Devices" button has become active and pluralized, with the inventor drawing a line to connect said button with the device list, while circling his iPhone in the list of devices that're online. Those unskilled in the art can clearly discern that three mobile phones are online and available for lost "discovery" mode to be enabled. One additional basis highlighting this is the fact the other three buttons in the interface have been "grayed out" and thus cannot be pushed until a device has been first selected in the device list and then the "Find Device" button has been pushed. In yet another example, since the "Find Device" button has pluralized its text with three online mobile phones depicted, an impartial observer not skilled in the art cab determine that all three online devices could be similarly locked with lost "discovery" mode; otherwise the button would not be pluralized.

318. **Claim 8**. The computer-implemented method of claim 1, wherein the remote command comprises a locate command, the computer-implemented method further comprising: receiving a result message including geographic coordinates corresponding to a location of the mobile device.

319. **Exhibit 9** depicts a remote command comprising a locate command, whereas the obvious "Find Device" button then executes a remote locate command; which is transmitted from the cloud server to the mobile device declared lost by the true owner. A resulting result message is transmitted from the lost device to the cloud server, which includes the approximate geographic coordinates. This location data is then presented on a map overlay, as depicted in the "Example UI" also contained in **Exhibit 9**. Plaintiffs embodiment shows his lost iPhone in Los Gatos; while also showing two previous locations in Saratoga and Cupertino—the coordinates of which were transmitted to the cloud server from the iPhone after lost "discovery" mode had been enabled by pressing the "Find Device" button.

320. **Claim 10**. A non-transitory computer-readable medium, storing instructions executable to cause one or more data processing apparatus to: display a list of one or more mobile devices associated with a remote management account; receive input selecting a mobile device included in the list of one or more mobile devices; display a list of two or more remote commands

available to be performed by the selected mobile device, wherein the displaying the list of two or more remote commands includes identifying at least one of the two or more remote commands as enabled for execution by the selected mobile device and at least one other of the two or more remote commands as disabled for execution by the mobile device; receive input selecting a remote command from the list of two or more remote commands; generate a remote command message identifying the selected remote command; and transmit the remote command message to a server to communicate the remote command message to the mobile device.

321. Plaintiff demonstrates a list of one or more mobile devices associated with a remote management account in **Exhibit 9**, whereas the "Example Process UI" shows three iPhones—in a list and connected using a cloud server login. Four buttons are presented in this UI for executing remote command instructions; with only one of them being available because the selected device hasn't yet been declare lost by the true owner. Once they push the "Find Device" button that's exposed with such text, the three buttons *supra* than become active and text for their functions is than presented for the user in the same manner as the "Find Device" button is, however, that button would now change states to allow the user to declare the phone found and end lost "discovery" mode. Button impressions to execute remote command instructions thus are transmitted from the server to the mobile device that's had lost "discovery" mode enabled. Herein Apple discloses the ability to display or execute two (or more) remote command instructions, whereas in plaintiff's original embodiment Apple later copied, he shows three discreet remote command instructions in the user interface, which could potentially be actuated; in-addition to the fourth button for beginning and ending lost "discovery" mode itself. Plaintiff has four distinct buttons for displaying and executing available remote command messages; whereas Apple's **Figure 15** display features remote command instructions grouped into congruent boxes for each command, exactly as plaintiff had already depicted.

322. Since this claim's reinforcing the cloud server component interacting with the lost mobile device when presenting and executing remote commands, it's instructive to compare how identical the topology is between connected mobile device and computers using the same account to communicate with a server in plaintiffs **Exhibit 10** and **Exhibit 12** against Apple's **Figure 1**; as well as **Exhibit 9** depicting the same remote command structure and topology as **Figure 15**. The only difference is the significantly better quality of Apple's diagrams; copying the plaintiffs handwritten notes into formal topologies with device images took far more time than plaintiff had to write down his ideas and plans while actually inventing them. A warm photocopier always

produces excellent copies; Apple had grown accustomed to copying plaintiffs work misjoinder across several patent applications. This explains why each patent draws from the same theme of images; plaintiff didn't create any more notes that Apple could utilize.

323. **Claim 11**. The non-transitory computer-readable medium of claim 10, wherein the instructions are further executable to cause the one or more data processing apparatus to: display, for one or more mobile devices included in the presented list, an indication that the mobile device is online.

324. As discussed *supra* for claim 6, **Exhibit 9** shows an "Example Process UI" that clearly depicts three iPhones by unique device name which're online; as the "Find Devices" button has become active and pluralized, with the inventor drawing a line to connect said button with the device list, while circling his iPhone in the list of devices that're online.

325. **Claim 13**. The non-transitory computer-readable medium of claim 12, wherein the item of information comprises a message to be displayed on the mobile device.

326. Apple's **Figure 12** has a user interface element which says, "if found, please call Jake at 866.555.1234" at **1210**, which looks very similar to plaintiffs "Example Lock Screen When Lost" in **Exhibit 12**. Its decidedly fitting and ironic that both **Exhibit 12,** and, **Figure 12** share the same number herein—the latter is a copy of the former. The message to be displayed on the mobile device is so predominant Apple provided a screenshot of it for the scope note; on the very first page of the application. Apple here loses in its attempt for claim ambiguity, after its warm photocopier got the best of its application drafter on the very first page with **Figure 12**.

327. **Claim 16**. A computing system comprising: an input interface; a display; a network connection configured to interface with a communication network; and processor electronics configured to: present, on the display, a user interface listing one or more mobile devices associated with a remote management account; receive, via the input interface, a selection corresponding to one of the one or more mobile devices; present, in the user interface, two or more remote commands, wherein presenting the two or more remote commands includes identifying at least one of the two or more remote commands as enabled for execution by the selected mobile device and at least one other of the two or more remote commands as disabled for execution by the selected mobile device; receive, via the input interface, a selection corresponding to one of the two or more remote commands; and transmit, via the network connection, a remote command message corresponding to the selected remote command to a server to communicate the remote command message to the mobile device.

328. The embodiment described herein exists in plaintiffs exhibits, as previously discussed. The input interface is represented in **Exhibit 9**, using the “Example Process UI” to show how the feature’s enabled, disabled and operated during use. The network connection configured to interface with a communication network is clearly established in **Exhibit 10**, whereas plaintiff shows mobile devices using both cellular and switched networks to reach the cloud server; which, in-turn receives and transmits both location data and remote command instructions. The user interface listing one or more mobile devices associated with a remote management account is depicted with three iPhones in **Exhibit 9**, as previously discussed. The selection, enabling and disabling of remote command instructions is depicted in the same example user interface, and, utilize the same communication network topology discussed *supra*.

329. **Claim 18**. The computing system of claim 16, wherein the processor electronics are further configured to: present, in the user interface, one or more disabled remote commands corresponding to the selected mobile device; receive, through the input interface, a selection corresponding to one of the one or more disabled remote commands; and transmit a message to the server to be communicated to the mobile device, the message enabling the disabled remote command for execution by the selected mobile device.

330. Previous discussion *in re* disabled remote commands *supra* has demonstrated and explained how the example user interface depicted in **Exhibit 9** is showing three commands currently disabled; with the one command being enabled in the embodiment being the “Find Devices” button. This button executes and stops lost “discovery” mode, which than activates the three remote command buttons depicted. This distinction is important for two reasons. First, the buttons are shown in a deactivated state with the operative path to enable them, as demonstrated by the “Find Device” button being enabled. Secondly, there’s no reason plaintiff otherwise would have specifically denoted three obvious remote command buttons above an already active button; the undisputable fact they exist shows clearly their intended purpose. Lastly, plaintiff wisely wrote the text “button” directly to the left of the topmost deactivated button; removing any confusion those unskilled in the art could even remotely espouse *in re* their purpose.

<u>COUNT 7 Patent 8,660,530</u>

Remotely receiving and communicating commands to a mobile device for execution by the mobile device

331. *The ‘530 patent includes the following claims plaintiff invented, specifically 1, 2, 5, 6, 9, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21 and 22 as listed below*. Evidence is supported by

Exhibit 1, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

332. **Claim 1**. A computer-implemented method, of remotely commanding a mobile device, the method comprising: receiving input from a user uniquely identifying a mobile device, wherein the mobile device is remotely located from the user; presenting to a user one or more remote commands available to be performed by the mobile device, wherein the presenting includes identifying at least one of the one or more remote commands as enabled for execution by the mobile device; receiving user input selecting a remote command from the one or more presented remote commands; generating a remote command message instructing the mobile device to execute the selected remote command; and transmitting the remote command message to a notification server for publication in a command node included in a command collection topic uniquely subscribed to by the mobile device, where the command collection topic is one of a plurality of command collection topics hosted on the notification server and the command node is one of a plurality of command nodes included in the command collection topic, wherein the notification server notifies the mobile device that the remote command message is available for retrieval in the command node, and wherein the notification server transmits the remote command message to the mobile device only in response to the mobile device accessing the command node.

333. The computer-implemented method herein refers to the cloud server; which manages the users connected devices when locating a lost mobile device is actuated. Even if a standalone application is used to locate, track and send remote control messages to a lost device, the cloud server still necessarily handles all the requested functions and transactions.

334. Remotely commanding a mobile device involves using plaintiff's novel method to uniquely identify the true owners lost mobile device; which is not located in their proximity and currently residing in an unknown location. As previously discussed in the '32 patent in **Count 4**, The "Example Process UI" in plaintiffs **Exhibit 9** depicts the hardware identifier uniquely identifying the mobile device to the server, using user credentials that are identifiable by the hardware identifier; presenting, on the mobile device, a user interface for configuring the mobile device, the user interface including a challenge for authenticating a user of the mobile device based on the user account information received from the server. This involves storing a record on the cloud server to differentiate which devices correspond to the privileged user accounts associated with them, using a portion of the unique IMEI (if it's a mobile telephony device) or MAC address for computing devices. Plaintiff demonstrates his inventorship overwhelmingly; as

the diagram in **Exhibit 10** displays both the MAC address and IMEI as discreet fields, connected to the cloud server.

335. Identifying at least one of the one or more remote commands as enabled for execution by the mobile device is clearly depicted in plaintiffs **Exhibit 9**. The “Example Process UI” displays list of three privileged devices; which have authenticated with a cloud server also registered with each device using the same user account. A “Find Device” radio button is located in this embodiment, which transmits a locate remote management command to the mobile device. Three other remote command message buttons are connected to the “Find Device” button, but, are in a deactivated state; as they cannot be executed by the user until the device has first been found using the remote locate command message available for user impression in the diagram.

336. Selecting a remote command from the one or more presented remote commands and generating a remote command message instructing the mobile device to execute the selected remote command is depicted in plaintiffs **Exhibit 9**. As mentioned *supra*, four distinct remote command buttons are depicted in the example web interface. Once the highlighted lost iPhone belonging to the plaintiff (in the diagram) has been found, the resulting three buttons become available for the user to execute with a mouse cursor impression. Once the user has thus pressed the “Find Device” button, it transmits the remote command message to a notification server (which is the same as the cloud server throughout plaintiffs diagrams and narrative) for publication in a command node included in a command collection topic uniquely subscribed to by the lost mobile device; where the command collection topic is one of a plurality of command collection topics hosted on the notification server and the command node is one of a plurality of command nodes included in the command collection topic, wherein the notification server notifies the mobile device that the remote command message is available for retrieval in the command node, and, wherein the notification server transmits the remote command message to the mobile device only in response to the mobile device accessing the command node. The command collection topic disclosed herein contains four remote command messages in the plaintiff’s embodiment. As plaintiff explains in **Exhibit 8** (under the distinct “Message” section of the workflow) the remote message for signaling to an honest finder a means of communicating with the true owner is accomplished using “device privilege” mode; which constitutes the lost mobile device receiving and executing remote command messages from the cloud server. Example remote command messages are further disclosed by the plaintiff in this section of the workflow in **Exhibit 8**.

337. **Claim 2**. The computer-implemented method of claim 1, further comprising: presenting to the user a selectable list of mobile devices associated with a remote management account, the selectable list including information uniquely identifying each mobile device.

338. As detailed in the “Example Process UI” in **Exhibit 9**, the example user interface displays a list of three distinct iPhones; each containing a unique device name displayed in the device list. Darren, Nicole and Junior each have a nickname being depicted for their unique mobile telephony devices. The remote management account is clearly established in the same example user interface, as a username and password field for the cloud server are featured, *supra*.

339. **Claim 5**. The computer-implemented method of claim 1, wherein the remote command comprises a locate command.

340. Herein described is the cloud server issuing a remote command instruction to locate a device that’s been chosen by an authenticated user. The “Example Process UI” in plaintiffs **Exhibit 9** depicts a “Find Device” button, which sends a remote command instruction from the cloud server to the device the owner has declared lost, and, chosen in the device list. The locate command executes on the mobile device; using either the GPS circuit, or, network location data to determine the approximate geographical coordinates, which are then transmitted to the cloud server. When Apple states, “obtain device identification information” in **Figure 2** at **215**, this refers to also needing to determine if the lost mobile device has a GPS circuit, or, must use network location data—this includes most non-telephony devices. While identifying the device from the unique hardware identifier is the only stated purpose at **58** for **215** in the application, this is actually done in the step for providing access information at **210**; as authentication than returns the populated device list, as depicted in plaintiffs **Exhibit 9**.

341. Herein is another improper understanding of both how the feature works and basic networking; exposed from copying the plaintiffs notes and desperately trying to make the disclosure appear as their own creation. Apple pretends that magic simply determines whether the lost device has a GPS, or, must rely on triangulation data. This is both embarrassing and frustrating, as the application actually discusses on page 15 that a device may (or may not) have a GPS circuit; exactly one paragraph before explaining how the “locate” command works at **5**. The only discussion *in re* obtaining device information discusses the unique hardware identifier. This is an abysmal failure; the device listing on the cloud server doesn’t know how to magically “wave a wand” and determine whether to expect near-precise GPS coordinates in real-time, or, only approximate locations which may sometimes not be accurate at all! Such a difference must be

detected from the hardware device IO plane of the mobile device. Detecting the IMEI or MAC address does nothing to differentiate whether it has a GPS circuit. Many millions of smartphones have an IMEI, but do not contain GPS, and, can never have such a circuit added later. This distinction becomes critical for the proper operation of the locate command, as the results which are then presented to the user on a map overlay may have a significant delta between location accuracy. Any person who's ever had a GPS device tell them to perform a turn off a bridge, or drive through an ocean, for example, understands that the information given to even a precise instrument for measuring distance is subject to correct curation by the human interpreting the results. As such, location triangulation can sometimes be very precise, but like incorrect map logic in GPS devices, they can sometimes be quite wrong. If an access point the lost device can reach to communicate to the cloud server was moved back from one's residence before those results are ever realized by the data provider, it may cause the device to appear located in Brooklyn, New York when it's really in Las Vegas, Nevada. The ramifications in cases of abduction or kidnapping can be dire, however, a more common scenario might see a device location in an adjacent city, whereas the user is expecting it to be much closer. The detection of what network hardware is present is thus crucial, as presenting network location data from the nearby connected data should never be used over a GPS circuit. Apple's disclosure fails to account for this; assuming the cloud server "magically" knows what to ask the mobile device processor to use. Since any telephony device which has GPS nearly always has a wireless network card, this could potentially be a life-threatening omission, which magic cannot remedy. Given there's 786 patents cited, and, it took 5 years for the application to be approved, it's clear the PTO had similar reservations about the applications disclosures. Significantly more complicated pharmaceuticals often achieve patents much faster with less examination required; suggesting to those skilled in the art that if the plaintiff had been properly joined, the application wouldn't suffer from confusing defect and would've been approved much sooner—especially given Apple paid for application priority.

342. **Claim 6**. The computer-implemented method of claim 5, further comprising: receiving a result message including geographic coordinates corresponding to a location of the mobile device.

343. Herein we turn to the cloud server receiving the location data associated with the lost mobile device executing the locate remote command instruction depicted as the "Find Device" button in plaintiffs **Exhibit 9**. As discussed previously, the connection path depicted in

Exhibit 10 demonstrate how the results of a remote command may be returned to the server.

344. **Claim 9**. A computer program product, encoded on a non-transitory computer-readable medium, operable to cause data processing apparatus to perform operations comprising: presenting to a user a list of one or more mobile devices associated with a remote management account; receiving user input selecting a mobile device included in the list of one or more mobile devices; presenting to the user a list of one or more remote commands available to be performed by the selected mobile device, wherein presenting the list includes identifying at least one of the one or more remote commands as enabled for execution by the selected mobile device; receiving user input selecting a remote command from the list of one or more remote commands; generating a remote command message identifying the selected remote command; and transmitting the remote command message to a notification server for publication in a command node included in a command collection topic uniquely subscribed to by the selected mobile device, where the command collection topic is one of a plurality of command collection topics hosted on the notification server and the command node is one of a plurality of command nodes included in the command collection topic, wherein the notification server notifies the mobile device that the remote command message is available for retrieval in the command node, and wherein, only in response to the mobile device accessing the command node, the notification server transmits the remote command message to the mobile device.

345. Presenting to a user a list of one or more mobile devices associated with a remote management account is depicted in plaintiffs **Exhibit 9**, as discussed *supra*. A user selecting a device from a list of two (or more) is depicted with three iPhones in this same example. Note that the iPhone bearing the plaintiffs first name is circled at the top of the device list; to denote it's been chosen using a mouse cursor. The list of one or more remote commands available to be performed by the selected mobile device is also depicted in this example, with four (total) remote command buttons visible; the remote command button for issuing the locate command is shown as active. The transmission of the remote command message to a notification server for publication in a command node (included in a command collection topic uniquely subscribed to by the selected mobile device) where the command collection topic is one of a plurality of command collection topics hosted on the notification server and the command node is one of a plurality of command nodes included in the command collection topic, wherein the notification server notifies the mobile device that the remote command message is available for retrieval in the command node, and wherein, only in response to the mobile device accessing the command

node, the notification server transmits the remote command message to the mobile device, is depicted in plaintiffs general workflow narrative under “Messages” in **Exhibit 8**; as well as the copious example provided in **Exhibit 12**. The various application or web browser clients are organized in a diagram connected necessarily to the cloud server; which is effectively the notification server herein. In presenting the data of a device location, this diagram illustrates how command messages are managed by the cloud server; with the lost device only executing such remote command messages when lost “discovery” mode has been enabled by the true owner. Otherwise, there’s no purpose for the user’s plurality of mobile devices to remain subscribed to such remote events only needed if the device has been declared lost. Depicting the “Example Lock Screen When Locked” in this same exhibit both demonstrates and proves a successful execution of a remote command message from the mobile device’s processor; after necessarily being subscribed for the duration of the lost “discovery” mode, invoked by the cloud server. Moreover, the command node as referenced here is the cloud server; using its application or web browser interface from **Exhibit 12** to perform user impressions of the remote command message buttons depicted in **Exhibit 9**.

346. **Claim 11**. The computer program product of claim 9, further operable to cause data processing apparatus to perform operations comprising: prompting the user to provide an item of information in response to the selected remote command.

347. Herein this claim concerns the true owner providing information in response to selecting the remote command which displays a message for an honest finder on the screen of the lost mobile device; while it’s in lost “discovery” mode. The practical embodiment used in the Apple feature, shown in **Figure 11** of the application, and finally, in plaintiffs **Exhibit 12** are one and the same. The “Example Lock Screen When Lost” depicted in **Exhibit 12** states that [Name of iPhone] phone is lost, and, to call a sample 10-digit telephone number (123-456-7890) to reach them. **Figure 11** looks nearly identical to the plaintiff’s example user interface, depicting, “if found, please call Jake at 866.555.1212.” in **1110**. The user herein is providing an honest finder with a name and telephone number to reach them; while they could also list alternate forms of contact, such as an email or instant messaging aliases. The fact Apple copied plaintiffs example interface so closely (using a name and telephone number as items of information which could be provided) as an example in response to the selected remote command is not accidental. Moreover, plaintiff discloses that, “[the] user record allows storage of device names and contact numbers” in circled text located next to his example user interface. The plaintiff thus shows irrefutably that the

user of the feature (the true owner of the lost device) is providing information in response to a remote command button, as depicted in **Exhibit 9**. In the plaintiffs example embodiment, the messaging on the display of the device (for an honest finder) clearly identifies that the user has provided both their name and telephone number; otherwise the message would contain nothing more than stating that's it's been lost—which doesn't solve the longstanding problem of an honest finder having a reliable method to return the device to its true owner. Therefore, the surrendering of some information from the user is necessary for both the claim (and elements of the feature itself) to indeed be novel. Lastly, plaintiff discloses in **Exhibit 8** that a, "custom user defined message" would be used under the "Messaging" section of the operative flowchart.

348. Additionally, plaintiff discloses the idea in **Exhibit 11** of disabling the true passcode; if the user requests it, after activating the remote command message for lost "discovery" mode. Plaintiff states, "we could lock the device and invalidate the true passcode while privileged mode is in-use." This is relevant because plaintiff demonstrates herein again a remote command message being sent to the lost mobile device requiring information from the true owner. In this secondary example, the user must provide their true passcode after disabling the remote command; to temporarily invalidate the true passcode, for extra security. Otherwise, there'd be no adequate assurance that the cloud server user account also belonged to the true owner of the device. The harmless example of a child's mobile phone's appropriate here; as the child has control of the passcode, but parents may have control of one or more cloud server accounts which have privileged access to the device. While juniors iPhone may need a passcode to keep other children at school from accessing the device contents, for instance, a parent may have the feature enabled to locate the device if they cannot locate their child; or, more commonly, the child loses the phone. Requiring a passcode to disable the true passcode while in lost "discovery" mode is helpful and necessary if a typical user misplaces the device, then finds it long before the true owner could disable the remote command message invalidating the true passcode. As discussed at **179**, the scenario of an employee being stuck without use of their device all day after misplacing it temporarily before their shift applies in this instance. While the device may have been recovered by the typical user a short time after declaring it lost to their boss, it might take an extended period before the true owner could intervene to disable lost "discovery" mode; thus, the importance of the user providing information in response to the selected remote command. Plaintiff had concerns about the features durable function making it impossible to defeat in cases where the device is recovered by the typical user before the true

owner can intervene and restore functionality. The accidental enabling scenario is thus mentioned in plaintiffs **Exhibit 11**, where he states, “we must allow the device to be unlocked due to accidental enabling, or, the phone being found.” These instances all necessarily require the typical user of the lost device to provide information to deactivate their enabling of remote commands.

349. **Claim 12**. The computer program product of claim 11, wherein the item of information comprises a message to be displayed on the mobile device.

350. Plaintiff provides examples of messages to be displayed on the mobile device as discussed *supra* in **Exhibit 8**, **Exhibit 11** and **Exhibit 12**, using an item of information; most especially the “Example Lock Screen When Lost” in **Exhibit 12**.

351. **Claim 14**. The computer program product of claim 9, further operable to cause data processing apparatus to perform operations comprising: retrieving a result message generated by the mobile device in response to the remote command from a result topic identified in the remote command message.

352. The results of executing the remote command operation is depicted in plaintiffs example user interface buttons in **Exhibit 9**; showing four buttons actuated by the user from the cloud server, capable of retrieving a result message generated by the mobile device in response to the remote command from a result topic identified in the remote command message. Retrieving a result message in one example embodiment is shown both in Apple’s application and by plaintiff is displaying the location of a device geographically on a map; as displayed in **Exhibit 9**, and, both **Figure 9** and **Figure 10**, respectfully. Herein the cloud server is responding to the remote command by displaying the location result transmitted by the processor of the mobile device.

353. **Claim 15**. A system comprising: a server hosting a plurality of command collection topics; and a computing system including an input interface, a display, and processor electronics configured to perform operations comprising presenting, on the display, a user interface listing one or more mobile devices associated with a remote management account; receiving, through the input interface, a user selection corresponding to one of the one or more mobile devices; presenting, in the user interface, one or more remote commands, wherein presenting the one or more remote commands includes identifying at least one of the one or more remote commands as enabled for execution by the selected mobile device; receiving, through the input interface, a user selection corresponding to one of the one or more remote commands; and transmitting a remote command message corresponding to the selected remote command to the server for publication in one of a plurality of command nodes included in a command collection

topic uniquely subscribed to by the selected mobile device, wherein the server notifies the mobile device that the remote command message is available for retrieval in the command node, and wherein, only in response to the mobile device accessing the command node, the server transmits the remote command message to the mobile device.

354. A server hosting a plurality of command collection topics, and, a computing system including an input interface is depicted in plaintiffs **Exhibit 10**. "Location data sent to Apple" as in **Exhibit 8**'s operative flow can only indicate that it's being transmitted to a cloud server. The instructive question for those unskilled in the art is, what other possibility could exist? It's beyond doubt a lost device wouldn't send location data by ringing Apple on a telephone, or, by attaching the coordinates to a paper message for a carrier pigeon to fly to Cupertino—from any point in the world, and, at any time.

355. The user interface listing one or more mobile devices associated with a remote management account; receiving, through the input interface, a user selection corresponding to one of the one or more mobile devices; presenting, in the user interface, one or more remote commands, wherein presenting the one or more remote commands includes identifying at least one of the one or more remote commands as enabled for execution by the selected mobile device, is depicted in **Exhibit 9**. While previously discussed, the "Example Process UI" depicts a user interface with four remote command buttons; all of which could potentially be executed by the selected mobile device from the plaintiff's device list. In plaintiff's embodiment, three iPhones are listed as being registered to him on the cloud server. The entry for "Darren's iPhone" is circled, indicating that the user has selected it with a mouse cursor; with the intent of issuing one (or more) remote commands. The specific remote command identified in **Exhibit 9** is executed first; enabling lost "discovery" mode, and, three additional remote commands buttons, *supra*.

356. The selected mobile device; receiving, through the input interface, a user selection corresponding to one of the one or more remote commands; and transmitting a remote command message corresponding to the selected remote command to the server for publication in one of a plurality of command nodes included in a command collection topic uniquely subscribed to by the selected mobile device, wherein the server notifies the mobile device that the remote command message is available for retrieval in the command node, and wherein, only in response to the mobile device accessing the command node, the server transmits the remote command message to the mobile device, is displayed in **Exhibit 12**. Herein, the lost iPhone has been issued a remote command message to locate itself from the command node using geographic

coordinates. After the “Find Device” button is actuated by the user in **Exhibit 9**, the cloud server transmits the remote command instruction to the iPhone; whereas the processor calculates this using its GPS circuit, transmitting the location data to Apple’s cloud server. In this example, the server has uniquely subscribed the iPhone to command collection topics. The “Presenting Data of Device Location” diagram show the server transmitting and receiving location data using a remote command instruction with a lost iPhone; which is accomplished over the available communication network. The receiving and transmission of remote command data may occur with a web browser, an application on a computer or mobile device tasked for the purpose (such as Find my iPhone) or even a custom interface built into all iOS devices as an extended part of the operating systems functionality; meaning that an application doesn’t have to be installed by the user for this task, for added convenience.

357. **Claim 16**. The system of claim 15, wherein the server is further configured to host a result topic.

358. The cloud server must host the result topic, else the resulting data from location changes (for example) would not be available to chart the lost devices past locations. The example user interface depicting a lost iPhone’s current (and past) geographical locations in plaintiffs **Exhibit 9** would not be possible, as would Apple’s example of “Jake’s iPad” in **Figure 9** and **Figure 10**. The obvious distinction between these examples is that Apple only shows “Jake’s iPad” in one location on a map overlay at **915** and **1025**; whereas plaintiff illustrates his lost iPhone in a current location, and, two previous ones. Further, there’s physically no other method to convey the result data reliably without using a server. While it’s technically possible to send the remote command message data directly to another peer device, it still must use the same communication network, but, is subject to several fatal deficiencies; resulting in a failure for the peer device to correctly (and reliably) receive or interpret the transmitted result data from the processor of the lost device. The reliability inherent in a cloud server predicates that it must receive message data from a lost mobile device, however, in both plaintiff’s original embodiment and Apple’s later application copy, the cloud server is also transmitting the remote command which produces the result message. This claims differentiation of the server (being further configured to host a result topic) is simply for application ambiguity.

359. If plaintiff had been properly joined to the patent, the application would actually explain the practical and programmatic reasoning inherent here; given he has 32 years of programming experience. Quite alarmingly, a search of the application produces zero results for

the word “array” and is disturbing for those skilled in the art. In programming, irrespective of the language used, an array is used to collect and manage responses deriving from an operation; most typically in modern times this involves a preference change or response to an action taken by the user with either a keyboard, or mouse cursor impression. Computers don’t know how to “remember” anything unless its stored in some form of array. Given plaintiff was working with arrays in Pascal before he was 10 years old, it’s extremely alarming (and further proof of Apple’s blatant misjoinder and nonjoinder) that Apple didn’t explain this as a non-transitory, standard part of receiving responses. The counsel retained to draft the application by Apple had to necessarily rely on the disclosures by the purported inventor(s); who don’t understand basic programming and were basing such disclosures on facsimile copies of plaintiff’s notes. On P4, L14, there should be a discussion concerning the result topic and the corresponding responses being organized into an array by the cloud server. Instead, it says, “The techniques also can be implemented such that the server is further configured to host a result topic. Also, the techniques can be implemented such that the processor electronics are further configured to perform operations including retrieving from the result topic a result message generated by the selected mobile device in response to the remote command message.” Processor electronics are “further configured” in most operations, so, what does this really mean? Any combination of magic and forced osmosis herein fail those skilled in the art. Again, as mentioned *supra*, the PTO obviously exercised arduous effort in deciding whether to approve the patent; which wouldn’t have occurred on a priority application if the correct inventor had provided disclosures. Explaining “how” a processor is configured for a method in a critical workflow of a patent is necessary for proper examination, but again, if you don’t really know and are copying others work, you make broad statements (as herein) which don’t make sense and raise the speculation of magic. If Apple had explained *how* and *why* the cloud server was configured to host a result topic, the patent application wouldn’t suffer from such lack of clarity, and, would have been approved sooner. Apple wanted to punish the plaintiff so badly for his responsible innovation that it inadvertently punished both the counsel drafting the application, and, both of the PTO examiners. Curiously, Apple *does* mention non-transitory elements in the other claims of this application. This indicates beyond doubt that even the patent counsel was confused with the misjoinder disclosures; as failing to use such language will almost certainly lead to the application being denied, or at the very least, the need to remove claims. This was only accomplished from the patent counsel by doing research; as they are skilled in the art, but not programmers. Therefore, the patent counsel “saved” the application from denial, but, missed the fundamental need to explain an array being necessary, and, why that predicates the necessity of using a cloud server to execute and interpret results from the remote commands.

360. **Claim 17**. The system of claim 16, wherein the processor electronics are further configured to perform operations comprising: retrieving from the result topic a result message generated by the selected mobile device in response to the remote command message.

361. The retrieving of a result message from a remote command is illustrated in plaintiffs **Exhibit 12**, where the location of the lost iPhone is transmitted to the server, and then, displayed on a map overlay with geographic coordinates in **Exhibit 9**. The example in **Exhibit 9** shows dynamic updating of the lost devices position on a map; with the current and former positions displayed. Each time the device processor detects an appreciable delta in physical location, it sends a corresponding result message containing location data to the server; which is depicted in this example with the plaintiffs iPhone being in Los Gatos, after being detected in Cupertino and Saratoga—when the remote command message to locate the device was issued from the command node. Alternatively, if the lost device cannot be located when the remote commands issued from the server, this failed result itself will be transmitted as a result message.

362. **Claim 18**. The system of claim 17, wherein the result message includes an execution time associated with the selected remote command.

363. Significant beforementioned discussion *in re* timestamps being associated with a selected remote command is contained and interrogated at **163**, **213**, **214**, **215**, **216**, **224**, **234**, **252**, **256**, **257**, **259** and **290**. The narrative in “Handling Device While Stolen” alone in **Exhibit 11** illustrates the importance of associating timestamps with remote commands; else it would be impossible for law enforcement to recover an abducted or kidnapped person, as described.

364. **Claim 19**. The system of claim 16, wherein the processor electronics are further configured to perform operations comprising: presenting, in the user interface, one or more disabled remote commands corresponding to the selected mobile device; receiving, through the input interface, a user selection corresponding to one of the one or more disabled remote commands; and transmitting a message to the selected mobile device enabling the disabled remote command for execution by the selected mobile device.

365. The plaintiff’s beforementioned example in **Exhibit 9** (of remote command buttons not becoming active in the user interface until lost “discovery” mode has been enabled by the true owner pushing the “Find Device button) is especially appropriate here. This claim deals

exclusively with the enablement of remote command messages, which are disabled by default. Once the mobile device in **Exhibit 9** receives the issued locate command, it enables the subscription of other remote commands; as well as the execution of the location command itself. This causes the remaining three remote command buttons (in plaintiffs example user interface) to become available for execution by the user. This is why plaintiff intentionally depicted the three additional remote command buttons as being present in the user interface, but inactive.

366. **Claim 20**. The system of claim 15, wherein the selected remote command comprises a locate command.

367. As previously discussed in **117**, the locate command as described in this claim is discussed by the plaintiff in operative detail in **Exhibit 8**, while the user interface example to actually locate a device is shown in **Exhibit 9**; whereas a "Find Devices" button than locates the devices current geographical position. This geographic position is then charted on a user interface element with map overlay, as illustrated *supra* in the same exhibit. Apple's **Figure 8** shows the receive locate command **805** denoting the impression of the "Find Device" button in plaintiffs **Exhibit 9**. After determining the location **810** and **815**, the resulting geographic coordinates are published as a result message **820**; which is received by the cloud server in plaintiffs **Exhibit 8** and **Exhibit 10**, which allow it to generate the position(s) of the lost device on a map overlay in **Exhibit 9**.

368. **Claim 21**. The system of claim 20, wherein the processor electronics are further configured to perform operations comprising: receiving a result message corresponding to the locate command, the result message including geographic coordinates associated with the selected mobile device; and presenting, on the display, a map depicting a location of the selected mobile device in accordance with the associated geographic coordinates.

369. Beginning in the Radar umbrella feature bug in **Exhibit 8**, plaintiff describes, "display[ing] phone location after translating to GPS location for web display. Show device in map on web app[location] or page." In the next "Progress" section, it states, "chart lost path since last activation by user (if running again, chart previous "check-in" spots for map" followed by, "display device movement with charting of each check-in." Clearly, this necessarily describes a result message including geographic coordinates associated with the selected mobile device presenting a map depicting a location of the selected mobile device in accordance with the associated geographic coordinates on a display.

370. **Exhibit 9** features a map overlay in an example user interface, which has text

stating, “Your iPhone has been found here.” On the resulting map overlay is a circle and pin, which denote the lost iPhones current geographical position in Los Gatos. Pictured in this same map overlay are the devices two previous locations in the cities of Cupertino and Saratoga; these geographic positions being reported as a result message to the cloud server since lost “discovery” mode was enabled. As previously discussed, pressing the “Find Device” button depicted in the “Example Process UI” executes the locate remote command; the corresponding result data is then presented on a map overlay with geographical coordinates.

371. **Exhibit 12** includes a previously discussed diagram, entitled “Presenting Data of Device Location” describing how the geographical coordinates contained in the result message of the locate remote command are presented on a map overlay using a plurality of methods; including a web browser, standalone applications for computers or mobile devices, and finally, using a custom interface element in the iOS operating system for embedded mobile devices.

372. **Claim 22**. A computer-implemented method of remotely commanding a mobile device, the method comprising: receiving input uniquely identifying a mobile device; presenting to a user one or more remote commands available to be performed by the mobile device; receiving user input selecting a remote command from the one or more presented remote commands; generating a remote command message instructing the mobile device to execute the selected remote command; transmitting the remote command message to a notification server for publication in a command node of a command collection topic uniquely subscribed to by the mobile device, where the command collection topic is one of a plurality of command collection topics hosted on the notification server and the command node is one of a plurality of command nodes included in the command collection topic; wherein the notification server notifies the mobile device that the remote command message is available for retrieval in the command node; and wherein, only in response to the mobile device accessing the command node, the notification server transmits the remote command message to the mobile device.

373. The server receiving input uniquely identifying a mobile device; presenting to a user one or more remote commands available to be performed by the mobile device, and, receiving user input selecting a remote command from the one or more presented remote commands is depicted exactly in plaintiffs **Exhibit 9**, which depicts three uniquely identified iPhones in a device list organized for the plaintiff as the cloud server user, who’s just finished authenticating. The user has selected “Darren’s iPhone” to signal the computer-implemented method that this particular device is desired for executing remote commands. By locating the

device initially by pressing the "Find Device" button, this transmits the remote command message to the cloud server for publication in a command node of a command collection topic uniquely subscribed to by the mobile device, where the command collection topic is one of a plurality of command collection topics hosted on the notification server and the command node is one of a plurality of command nodes included in the command collection topic; wherein the notification server notifies the mobile device that the remote command message is available for retrieval in the command node; and wherein, only in response to the mobile device accessing the command node, the notification server transmits the remote command message to the mobile device. The command node depicted in plaintiff's embodiment contains four buttons for executing remote commands. As discussed s*upra* at **133**, **Figure 3** shows login beginning the process at **305**; presenting a list of linked mobile devices in **310**, a user then selects a mobile device from managed devices in **315**, available commands for the selected device occur in **320**, and, finally, the true owner can select a remote command to be executed in **325**. The plaintiff discloses the same, identical process in his earlier embodiment. Lastly, the steps in **Figure 12** contain the same events in **1205** through **1225** as plaintiff describes in **Exhibit 8** and **Exhibit 9**.

COUNT 8 Patent 9,125,014

Location-based ticket books

374. *The '14 patent includes the following claims plaintiff invented, specifically 1, 6, 7, 8, 9, 13, 16, 17, 18, 19, 22, 25, 26 and 27 as listed below*. Evidence is supported by **Exhibit 7**, **Exhibit 17**, **Exhibit 18**, **Exhibit 19**, **Exhibit 20** and **Exhibit 21**.

375. **Exhibits 17-21** were created January 7, 2003 and thus well before plaintiff's 2006 employment at Apple, which later saw the '14 application submitted a decade later. Plaintiffs notes represent a crucial narrative to corroborate plaintiff's pre-employment Apple IPA in **Exhibit 7**. Given lab notebooks are admissible, the decade between its creation and Apple's patent application nonjoinder of plaintiff further reinforces his previous claim.

376. **Claim 1.** A method comprising: receiving, by a mobile device, a virtual ticket, the ticket comprising a signal source identifier and a message for accessing a service of a service provider, the signal source identifier identifying a signal source being associated with the service provider; providing the signal source identifier to a wireless subsystem of the mobile device, the wireless subsystem executing a procedure for monitoring wireless signals from signal sources using a wireless processor of the mobile device; receiving, by the mobile device and from the wireless processor, a notification that the signal source identifier is detected in a wireless scan,

indicating that the mobile device is located within a communication range of the signal source; and then in response to an input requesting access to the service, providing, by an output device of the mobile device, a representation of the message to the service provider, wherein providing the representation of the message comprises: generating a barcode image from the message; and providing the barcode image as the representation for display on a display surface of the mobile device.

377. A method comprising of receiving, by a mobile device, a virtual ticket is depicted in **Exhibit 17**. A notebook computer is displaying a ticket for a September 4, 2003 performance of *Hamlet*, the barcode for which is being presented for redemption. Illustrated in the same exhibit's a flowchart highlighting how a virtual ticket is conceived, managed, redeemed and sold digitally online. The means for payment for virtual tickets before redemption's also disclosed.

378. **Exhibit 19** further contains a personal digital assistant, or PDA. The PDA device is similarly demonstrating on its display a virtual ticket to the same *Hamlet* performance, which is waiting to be redeemed. Also depicted is another view of the display screen of a PDA. On this display screen's a virtual ticket example for an NCAA baseball game between Univ. of the Pacific and Cal State Fullerton on January 7, 2003. The necessary distinction of this particular virtual ticket is that it was generated as a free student ticket, but, still granted using digital means. A Cal State Fullerton student (who logged into the student web portal [14] for verification of their academic eligibility for the complimentary ticket) was granted a virtual ticket; the barcode of which is being displayed on their PDA, for redemption at Goodwin Field.

379. A message for accessing a service of a service provider is declared by plaintiff in **Exhibit 17**, whereas the ticket server sends a message for access with public sales online, and more importantly here, redemption events and validation. These two processes are identified with discrete boxes and are connected with a line to the ticket server, which than interfaces with other potential facets of a virtual ticket transaction. In **Sheet 2**, Apple lists a ticket server **206** connected with a communications network **202** supplying the mobile device in **102**; along with signals **210** and **212**, which correspond with the vendor of the virtual ticket, which is taco truck **214** and potentially other facets of a virtual ticket transaction in **216**. The two methods are identical, with both Apple's and plaintiffs similarly providing the same apparatus to issue virtual ticket message

[14] CSUF Student Portal https://shibboleth.fullerton.edu/idp/profile/SAML2/Redirect/SSO;jsessionid=94A8387E6A4267FE19E129843335B7E1?execution=e1s1

signals to mobile computing devices.

380. Moreover, **Sheet 4** depicts the signal interface **424** and registry **422** enjoying two-way communication with the ticket manager at **406**; exactly as plaintiffs ticket server and ticket management software does in **Exhibit 17**. The message continues to the ticket book at **408**, whereas the UI manager in **412** presents the virtual ticket; which has been encoded into a barcode at **414**. The messaging (beginning with the message and parameters being passed to the eventual mobile device display window) is nearly identical in focus and operative scope. Plaintiffs embodiment shows the ticket manager communicating with the ticket server, which then sends and receives messages from the service provider *in re* virtual ticket redemption events. The registry and signal interface obviously behave in the same manner as in plaintiff's embodiment.

381. **Sheet 5** depicts the message containing the virtual ticket and pass message from the vendor in **502** beginning an event cycle; where it concludes by providing an output device of the mobile device, a representation of the ticket pass message at **508**. The only delta in additional messaging parameters concerns location data other co-inventors provided; the method, process and requirements are identical between plaintiff's signal messaging disclosure and that of claim 1.

382. Without such messages transmitted to the mobile device, there would simply be no method of conveyance for the virtual ticket to emerge from the issuer (a taco truck or regional university theatre) to the mobile device belonging to the patron. This represents a crucial reason why this longstanding problem had yet to be solved; even by Ticketmaster, for instance. If for no other reason, plaintiff's *conception* is demonstrated exclusively with messaging in claim 1.

383. On June 9, 2014, Ticketmaster released its iOS application, which transfers purchased tickets into Apple's Passbook application for redemption. [15] This demonstrated that both plaintiff and Mr. Jobs were correct in their much previous assertions that mobile devices were the best method for solving the digital ticket redemption problem.

384. Providing, by an output device of the mobile device, a representation of the message to the service provider, wherein providing the representation of the message comprises: generating a barcode image from the message; and providing the barcode image as the representation for display on a display surface of the mobile device, is disclosed in plaintiffs **Exhibit 17**. A mobile computing device has a barcode image clearly represented on the display surface for redemption. The PDA example containing the barcode for the *Hamlet* ticket in

[15] App Review: Ticketmaster for iPhone Makes Buying Easy, Has Few Shortcomings https://www.sportsbusinessdaily.com/Daily/Issues/2014/07/29/Media/App-Review

Exhibit 19 is identical, as is the free student virtual baseball ticket barcode. One not skilled in the art can discern that the embodiments generating barcodes for ticket redemption are virtually identical between plaintiffs 2003 notes and those in Apple's much later June 9, 2013 application. The only delta is the decade which passed in-between.

385. **Claim 6**. The method of claim 1, wherein the input comprises: a user activation of a display surface of the mobile device using a home button of the mobile device; a user gesture on a touch-sensitive surface of the mobile device to lock or unlock the touch-sensitive surface; or a user selection, from a quick-access menu, of an option for presenting the message.

386. An input comprising of a user activation of a display surface of the mobile device using a home button of the mobile device is depicted in plaintiffs **Exhibit 19**; wherein a PDA has a home button for device navigation, and, a virtual ticket barcode for *Hamlet* is being presented for redemption on the display surface. The user had to activate the device using the home button; as well as launching the application associated with presenting the depicted virtual ticket barcode.

387. **Claim 7**. The method of claim 1, wherein the barcode image includes a linear barcode or a two-dimensional barcode.

388. Plaintiff depicts a barcode image including a linear barcode (or a two-dimensional barcode) in **Exhibit 17**, whereas a portable computer has a barcode for a virtual ticket for *Hamlet* presented on the display surface for redemption. **Exhibit 18** contains a barcode image for the same event. **Exhibit 19** contains three linear barcodes. One's again for Hamlet; with the second an NCAA Super Regional Baseball Tournament, and finally, a college baseball game between the Univ. of Pacific at Cal State Fullerton. Moreover, **Exhibit 21** shows a linear barcode being redeemed by the university as a service provider; with three possible redemption responses.

389. Common symbology exists for linear barcodes, with code 39, code 128, UPC-A, UPC-E, EAN-13 and EAN-8. [16] All five of plaintiffs' barcodes clearly match the imagery for the various linear barcodes in-use worldwide.

390. **Claim 8.** The method of claim 1, wherein the ticket is a ticket for boarding a vehicle or attending an event, a store card, or a coupon.

391. Plaintiff discloses an example ticket for a *Hamlet* theatre performance in **Exhibit 17,** an example ticket for the same performance in **Exhibit 18**, an example ticket for the same performance in **Exhibit 19**, an example ticket for a NCAA Baseball Super Regional Tournament,

[16] Linear Barcode Symbologies http://www.systemid.com/learn/linear-barcode-symbologies/

and finally, an example student ticket for a college baseball game. Clearly, plaintiff was concerned with devising digital tickets for attending events, as he discloses in **Exhibit 19** that, "UAF could sell the turnkey solution to airports and education." UAF denotes University Advancement Foundation; who's responsible for generating fundraising and revenue. Herein plaintiff's disclosure *in re* airport usage certainly represents boarding a vehicle. Many airlines now accept boarding passes using the Apple Passbook application. On October 12, 2012 United Airlines began using Passbook for boarding passes. The resulting article displays a barcode presented on the display screen of an iPhone, exactly as plaintiff does with *Hamlet*. [17] In plaintiffs IPA from 2005 in **Exhibit 7**, it mentions, "ticket sales, reporting and management" in a clear reference to attending ticketed events.

392. **Claim 9.** The method of claim 1, wherein: the ticket is associated with a timestamp specifying a time the service is available, and the method comprises, before providing the representation of the message, confirming that a current time is within a time window that is determined based on the timestamp, wherein providing the representation of the message occurs if the current time is within the time window.

393. Time is an important element of validation. If an aircraft isn't available to board, or, an event venue has yet to open the doors, it's no different than the taco truck not arriving. Plaintiff previously disclosed problems with multiple event performances occurring on the same day on-campus, as well as the generally accepted problems associated with ticket management; such as forgery, expired tickets which've been reissued, or complimentary tickets needing to be issued for privileged access, such as media or student passes. As such, plaintiff discloses an example redemption unit being used to redeem virtual tickets in **Exhibit 21**. An example of the time requirement for validation demonstrated by plaintiff appears at the bottom, where it states, "this stops ticket from being marked as redeemed if patron goes to the wrong venue by mistake." The logic here, as derived from the application's that the redemption site knows what tickets should be accepted for that venue at which time, which necessarily includes the date. A similar process is in-place at the airport; where the proximity fence detects that the virtual boarding pass has been presented at the correct time, causing the "Error" or "Not Valid" messages to flash when the virtual pass was presented for redemption.

[17] United Airlines app for iPhone gets Apple Passbook support for boarding passes https://thenextweb.com/apps/2012/10/08/united-airlines-app-for-iphone-gets-apple-passbook-support-for-boarding-passes/

394. Presenting a ticket attendant with a paper ticket for a previous flight produces the same response that the invention thus does here; it politely tells the user they presented the wrong boarding pass or ticket. All computer servers necessarily function based on the current local time setting. This is relevant because plaintiff draws a line to correlate the ticket server and the validation of redemption events. With this obvious model, the ticket server will not grant redemption to the presenter of the virtual ticket if the timestamp doesn't match what the event descriptor has been programmed for. Using *Hamlet* again as an example, if a zealous patron arrived early and wanted to attend the first showing (and not the later second showing they'd purchased) the generated barcode on their mobile device cannot be redeemed; the timestamp for the event doesn't match the threshold. Using time as validation also allows for the same identical seats to be sold for two performances on the same day, but, to different patrons.

395. The application uses the following narrative to explain how Apple is using the same method as plaintiff; in using a ticket server authority to modify and validate the data used to encode barcodes presented for redemption as virtual ticket books on mobile devices. "In some implementations, ticket manager **406** can determine time window **416** for associating with virtual ticket **208**. Ticket manager **406** can determine time window **416** based on an expiration time of virtual ticket **208** and a ticket type of virtual ticket **208**. Upon determining that time window **416** has closed, ticket manager **406** can delete virtual ticket **208** or mark virtual ticket **208** as invalid. Time window **416** can be a point in time (e.g., ending 23:59:59 on December 31, 20xx, at a given time zone) or a time period (e.g., beginning at 00:00:01 and ending at 23:59:59 on December 31, 20xx at a given time zone). If time window **416** is a time period, ticket manager **406** can register the signal source identifier with wireless subsystem **420** upon determining that a clock of mobile device **102** has reached the beginning time of the time period."

396. **Claim 13.** A system comprising: a mobile device; and a non-transitory computer-readable medium coupled to the mobile device, the non-transitory computer-readable medium storing instructions operable to cause the mobile device to perform operations comprising: receiving a virtual ticket, the ticket comprising a signal source identifier and a message for accessing a service of a service provider, the signal source identifier identifying a signal source being associated with the service provider; providing the signal source identifier to a wireless subsystem of the mobile device, the wireless subsystem executing a procedure for monitoring wireless signals from signal sources using a wireless processor of the mobile device; receiving, from the wireless processor, a notification that the signal source identifier is detected in a wireless

scan, indicating that the mobile device is located within a communication range of the signal source; and then in response to an input requesting access to the service, providing, by an output device of the mobile device, a representation of the message to the service provider, wherein providing the representation of the message comprises: generating a barcode image from the message; and providing the barcode image as the representation for display on a display surface of the mobile device.

397. Herein this claim refers to the mobile devices capability to receive a signal source identifier and messaging, as well as generating a barcode image from the ticket server messaging. As these topics were previously interrogated, this claim's simply adding ambiguity; in reinforcing the mobile devices capability to interface with the ticket server authority.

398. **Claim 16.** The system of claim 13, wherein the input comprises: a user activation of a display surface of the mobile device using a home button of the mobile device; a user gesture on a touch-sensitive surface of the mobile device to lock or unlock the touch-sensitive surface; or a user selection, from a quick-access menu, of an option for presenting the message.

399. Herein a user is selecting to use the Passbook application, which then allows the presentation of a barcode on the display of the computing device; which is a virtual ticket being granted for redemption. A user has chosen to have a virtual ticket barcode generated on the display of a mobile device in **Exhibit 17** and **Exhibit 19**. Similarly, in the flow diagram above, the ticket management software is disclosed. Such software may exist on the memory of a mobile computing device; such as a portable computer, personal data assistant or smartphone. Even if a third-party application instructs Apple's Passbook application to activate and launch a virtual ticket for barcode generation on the display, or, generate the barcode from within the third-party application (such as Ticketmaster's *supra*) it still presents the virtual ticket at the user's command.

400. As in plaintiffs' **Exhibit 17** example, the ticket management software is presenting a virtual ticket for a performance of *Hamlet* on a mobile device. In addition to developing software in Java to make the code largely reusable on a plethora of different devices in a university environment, plaintiff similarly considered having the ticket server simply embed the barcode generation data along other text parameters of the ticket metadata (such as seat location, event name, event venue, price, etc.) into a vCard file; which could then be emailed to the patron upon the completion of the financial transaction. The advantage to this method was that a user needed only to open the attached vCard file from the message (using their email software) and it

would then be presented on the display for redemption on-demand. In another embodiment, the vCard file could simply be loaded into the devices contact list for email, which all low-cost PDAs allowed in 2003. This allows the vCard to be launched on-demand by the user—without needing to have a network connection, or, launch an email application. When an event was over, a patron could simply delete the vCard from their device, or, keep the now redeemed ticket on their device; in the same fashion many keep paper tickets from past concerts as souvenirs.

401. Data in the vCard format is stored according to the vCard specification; the files use the .vcf extension and are properly recognized by most applications and mobile devices. Storing a barcode, or even photos for art direction in ticket backgrounds possible; since .vcf files are simple text files. This latter option avoids needing to develop and maintain an app for the mobile device; as the ticket server is embedding the barcode generation information into a file which can thus be opened using any number of applications for email or contacts. Such software comes pre-installed on most devices, including low-cost PDAs at the time plaintiff devised the invention. This avoids having to distribute software for any mobile device which one may wish to use for redemption; as well as ensuring most mobile devices from the past, present and future can easily participate in virtual ticket redemption. Irrespective of the implementation method, plaintiff clearly indicated in his previous disclosure in **Exhibit 17** that software *of some kind* would be used to actuate the virtual ticket for redemption by presenting a barcode on its display.

402. Given the plaintiff wrote Mac software for the university and worked with a cadre of Windows developers; creating a discrete application for redeeming virtual tickets could easily have been realized. By using Java as a development language, plaintiff could have used the same code for cross-platform application support; which would also work on older devices with less memory and processor speed. This means instead of just being available on iOS devices like Passcode today, plaintiffs virtual ticket redemption solution would have also had applications available for Windows and UNIX devices. Plaintiff had found he could use Java on Palm devices (like those pictured in the exhibits) using a required virtual machine and a J2ME Connected, Limited Device Configuration; simply by using the free PalmOS Emulator, which allows a 32-bit version of Windows to be used to program in Java, using the J2ME standard. [18] The decision to embed vCard's with barcode generation data; versus creating a cross-platform redemption application was solely a management decision not made because plaintiff went to Apple,

[18] Program your Palm in Java: The PalmOS Emulator https://www.javaworld.com/article/2076524/program-your-palm-in-java--the-palmos-emulator.html

however, *plaintiff performed a reduction to practice while still employed at CSU Fullerton*. The potential to generate profit for the university using either implementation method was sanguine.

403. **Claim 17**. The system of claim 13, wherein the barcode image includes a linear barcode or a two-dimensional barcode.

404. As stated for claims 7 and 13, plaintiffs established a plurality of linear barcode imagery throughout his ticketing exhibits; with three virtual ticket barcode examples disclosed.

405. **Claim 18.** The system of claim 13, wherein: the ticket is associated with a timestamp specifying a time the service is available, and the operations comprise, before providing the representation of the message, confirming that a current time is within a time window that is determined based on the timestamp, wherein providing the representation of the message occurs if the current time is within the time window.

406. Using a timestamp for auditing when a virtual ticket may become usable for redemption has been discussed *supra*, however, **Exhibit 21** depicts such a procedure in action at the theater. Three different responses can occur from the scanning of the barcode on the mobile device. If the timestamp in the barcode information corresponds with an acceptable variable the ticket server has established for this performance of *Hamlet*, then the "valid" option lights and sounds a particular beep. This allows the lone attendant depicted managing six entrance lines into the theater in **Exhibit 20** to ensure each patron virtual ticket has been properly accepted. The timestamp can also reveal that some kind of error occurred; which may result in a plurality of causes rooted either in hardware or software failure, or even an incorrectly generated performance barcode. An attempt to counterfeit a barcode from a colleague's virtual ticket would produce the third response based on the timestamp, which is a "not valid" response. The ticket could even be an exact facsimile taken of another legitimate ticket from the previous day's performance that was not marked as redeemed; however, the timestamp will reveal upon redemption that it's not valid because it has already been redeemed, or, the time threshold for the event has concluded. Hence, the service is not available in this claim's language, using the later example; because a message indicating that the timestamp information doesn't match had been sent from the ticket server to the mobile device in **Exhibit 17**, when redemption of the virtual ticket was attempted.

407. **Claim 19.** The system of claim 13, the operations comprising: determining that the signal source identifier is no longer detectable by the wireless processor; and in response, stopping providing the representation of the message.

408. Herein is a continuation of the previous discourse. In the usage case where an

event has been cancelled, a patron secured a refund of their ticket, or even theft causing the subsequent reissuance of a new virtual ticket, the previous one's now been invalidated by the ticket server; which is represented sending such a potential validation message when the now invalidated virtual ticket is presented for redemption in plaintiffs **Exhibit 17** and Apple's **Sheet 4**. A performance of *Hamlet* here shares commonality with the Apple taco truck. If the absentminded or dishonest patron alike tries to present an invalidated Hamlet ticket, the same response message is provided as if they attempted to use an invalidated pass for tacos sold the day before. Herein additionally, one could make the virtual ticket no longer display a barcode image after an unsuccessful attempt has been confirmed by the validation authority. This also presents a convenience for the user with a ministry of previous events saved on their device; particularly in the "collector" usage case described *supra*. This allows honest patrons collecting virtual tickets from past events to easily see they've been redeemed; since the barcode no longer appears, however, the other metadata is still intact and can be recalled. It also helps prevent an invalidated ticket from being presented a second time, after previously being deemed not redeemable by the ticket server authority.

409. The signal source identifier being used to validate multiple locations is also present in the information generated on the barcode that's presented on the display for redemption. This is an important distinction, because plaintiff illustrates how the barcodes generated by the ticket server for virtual tickets in **Exhibit 21** display nine different locations where tickets could be uniquely redeemed; corresponding with the different venues existing in 2003 at Cal State Fullerton. This differentiation (between locations of the same vendor offering virtual tickets) is no different than providing services both standardized and individualized, as in line 7 of the applications summary explanation:

410. "In addition, the features described in this specification can allow service providers to provide services that are both standardized and individualized. For example, if multiple coffee shops are in a franchised coffee shop chain, and the franchised coffee shop chain wishes to have a standardized promotion across the chain, the franchised coffee shop chain can provide a pre-configured wireless beacon to each franchised shop. Each pre-configured wireless beacon can broadcast a same signal source identifier. A mobile device can then display a virtual ticket to access the same promotion in each of the coffee shops. In addition, if the franchised coffee shop chain wishes to have an individualized promotion (e.g., an experimental promotion in a pilot program) in one or more franchised shops, the franchised coffee shop chain can distribute

wireless beacons having another signal source identifier to the one or more shops. The mobile device, upon entering these shops, can display tickets for accessing the individualized promotion."

411. From examining the coffee shop summary, plaintiff's disclosure in **Exhibit 21** is individualized (because different places at the same location can accept tickets for different events) as well as standardized—the different university events could still be purchased on the same website. A standardized promotion could thus exist for alumni, or, students who've purchased an activity card for the term; wherein they get free or reduced-price tickets for all events at the university. When plaintiff attended Cal State Fullerton as a student, he could attend all sporting events for free. Whereas men's baseball and women's softball events had discreet NCAA fields located next to each other, they both had different events and tickets; as well as the public having the ability to purchase paid tickets from the same issuing authority. Both complexes could also be hosting ticketed games at the same time, as well as potentially basketball or track and field. While all four sports complexes are visible from each other and can all have discreet events occurring at potentially overlapping times, they're all the same vendor, and additionally, exist in very close geographic proximity. While one popular coffee chain location is sometimes present within sight of a second location (especially in airports) they all sell the same products and accept the same pricing terms. If a coupon for a free coffee was issued by Starbucks, it'd be redeemable in either of its nearby locations; whereas the student with an activity card could watch both a baseball and basketball game in the same day for free. Moreover, if performing arts subscribers were given a free ticket for a theater performance as part of an experimental program to increase ticket sales, a free virtual ticket for *A Midsummers Night's Dream* would properly redeem at the theatre, but, wouldn't redeem if presented instead at a basketball game occurring at the same time; despite both events being located near each other on the same campus, and, using the same ticket server authority. In the location proximity beacon example, the logic which determines which location may redeem which virtual ticket is one and the same. While GPS circuits weren't yet available in 2003 mobile phones, the barcode information for even an experimental promotion in plaintiff's embodiment enforces the same proximate location auditing; whereas in Apple's much later iPhone embodiment, the GPS information is used for convenience—to prevent the user having to sort through virtual tickets in the Passbook application at redemption time. Herein the barcode still contains logic from the virtual ticket authority, by which the ultimate decision whether to redeem the virtual ticket is made. If a coupon

barcode for a free coffee was instead presented at a franchise location which was not participating, it would not redeem when scanned by the store employee. Similarly, if that same virtual coupon was presented at a competitor located across the street who also accepts virtual tickets, it would similarly not redeem. This is because the logic in the barcode ultimately is the final authority in whether a coupon or virtual ticket can be redeemed. The signal source in this example would stop providing the representation of the message because it wasn't redeemable at that location. The "not valid" condition would be displayed when an employee scanned the barcode presented by the device, as shown in **Exhibit 21**.

412. This distinction's important as the applications background states that the operator of the mobile device is still ultimately selecting and presenting the virtual ticket barcode for redemption, it's not being done autonomously by location data, which would be imperfect and not cognizant of practical deltas; such as changed boarding gates with airline tickets. The background text also stipulates the user is selecting a virtual ticket to present on the device from the ticket book, which is a collection of virtual tickets the user has accumulated. The user still has the correct burden of deciding upon the correct virtual ticket to present for redemption. "The ticket book can store a user's various virtual tickets, e.g. boarding passes, movie tickets, retail coupons, loyalty cards on the mobile device. When the person arrives at a place where the virtual ticket can be used, e.g. a flight gate, a movie theater entrance, or a shop, the person can launch an application program that manages the ticket book service. The mobile device can display all virtual tickets stored in the ticket book for selection. The user can select a relevant ticket. The application program can display the user-selected ticket on the mobile device, for inspection by a ticket reader machine or person."

413. The correlation is further solidified at P8, L10 of the application, wherein it states *in re* signal source identifiers, "In some implementations, the ticket can be associated with a timestamp (e.g. expiration time **320**) specifying a time the service from the provider is available or will expire." The provider herein is Cal State Fullerton, with the timestamp being enforced both with the generated barcode data, and also the ticket server authority. A rescheduled performance constitutes an example where the timestamp information originally generated for the barcode is no longer valid, but, a new time has been substituted by the ticket server; which handles the logic necessary for the association and causes the "old" virtual ticket to be redeemed at the "new" time; despite the timestamp potentially changing considerably. In P4, L57, it states, "the signal source identifier can be provided by ticket server **206**. At least a portion of the signal

source identifier can match the signal source identifier included in virtual ticket **208**. Signal source **210** can be mobile." Thus, one may safely conclude that plaintiff's novel method of embedding timestamp information into the barcode generated when the virtual tickets sold is no different than the signal source identifier being included either in the virtual ticket, or, being provided by the ticket server.

414. Plaintiff also specifically mentions in **Exhibit 17** that, "a barcode generated when ticket sold, so it is unique." This agrees with the signal source identifier being included in the virtual ticket. On P6 exists a description of the diagram in **Figure 3B**, which illustrates the exemplary structure of virtual ticket **208**. "Virtual ticket **208** can be stored in location-based ticket book of mobile device **102**. The location-based ticket book can include multiple virtual tickets. Virtual ticket **208** can include signal source identifier **304**, expiration time **320** and payload **322**. Expiration time **320** can specify a time that virtual ticket **208** expires (e.g. ceases to be valid at a service provider) Payload **322** can include a message provided by the service provider, an encoding indicator and a ticket type." This agrees with the operational flowchart in **Exhibit 17**, which shows such a virtual ticket being created and redeemed through its potential lifecycle. Even if the patron wants to change their seat and get a new ticket, or, the venue wants to make a new seating plan and release tickets not included in the original seating plan, this accommodating change to the virtual ticket is also represented. Thus, the venue could also expire a virtual ticket before an event; so that a patron could upgrade their seat and the venue may release the once-claimed virtual ticket back into general availability for assignment; to be sold once again. The payload information changes in such cases, with the audit of the ticket server acting as an authority to the updated encoding in the barcode presented for redemption.

415. It's beyond doubt if plaintiff had been included in the claim disclosures, that Apple's application would include far more explanatory detail *in re* why different scenarios common in the ticketing realm make such a difference in the implementation. The greater purpose for the signal source identifier and the ticket server's role in encoding, for example, would become easier to understand. It's helpful to understand the many ticketing problems and workflow scenarios plaintiff solved a decade previous to Apple, which remain germane today. A taco truck illustration and the discussion of coupons for franchises doesn't help explain to one unskilled in the art why the solution's necessarily novel, save for repeated mentions of the location based GPS in the device predicating what virtual tickets to present; as choices for the user to then decide to present a corresponding barcode for redemption. The user still must unlock

their mobile device, launch the Passbook application and then choose a virtual ticket for redemption on the display screen. Since Passbook isn't autonomous and doesn't grant the user admission solely from a smartphone being in their pocket when they arrive to an event, the greater stated use of storing the event information in the virtual ticket itself is obscured. One additional way to understand this is from the discussion of UUID's (or universally unique identifiers) in the page 5 discussion of **Figure 3A** under Exemplary Data Structures.

416. "FIG. 3A illustrates an exemplary structure of a signal source identifier as used in a location-based ticket book service. Signal source **302** can be a signal source configured to broadcast signal source identifier **304** in a beacon signal. Signal source **302** can be signal source **110** of FIG. 1, or signal source **210** or **212** of FIG. 2. Signal source identifier **304** can be a programmable data structure having multiple portions. A first portion of signal source identifier 304 can include a universally unique identifier (UUID). The UUID can be a number having a specified size (e.g., 128 bits). The UUID can be unique for a group of signal sources designated to represent service provider **314**, and uniform among the signal sources in the group. For example, the UUID can correspond to a business operating food truck **214** and restaurant **216**. A mobile device (e.g., mobile device **102**) that has detected signal source identifier **304** broadcast by any signal source and identified the UUID corresponding to the business can present a virtual pass to obtain service from the business." It's easiest to think of a UUID as a potentially very long number; with portions corresponding to different values, which, is similar also to how a conventional linear barcode's encoded. Herein the UUID corresponds to Cal State Fullerton as a business selling tickets; instead of a taco truck, as well as other variables. The locations of each venue on campus in **Exhibit 21** represent another portion of this value by using one of nine digits. Using *Hamlet* as an example from **Exhibit 17**, another value denotes that the event is theater; as opposed to a concert, convention, festival or sports. A further sub-value of theater, *Hamlet* has been denoted as event 7 of the season by the ticketing server, as the performance calendar dictates. The date and time for the event are represented as themselves, with the section, row and seat as discrete constants—such as 15 for orchestra, 1 for the first section inside of orchestra, 1 for row one, followed by a 2 for seat 2. Other numbers may be used for additional metadata variables as needed, however, in this example, a UUID of 1,1,7,09042003,1,15,1,1,2 would than produce that unique encoding scheme for the barcode of the virtual ticket. Another patron sitting in seat 7 for a second performance the same day would a UUID of 1,1,7,09042003,2,15,1,1,7. A patron attending the NCAA Baseball Tournament in **Exhibit 19** would receive a virtual ticket

with a UUID of 1,3,1,00002003,001,001 before the date was later announced. The ticket server would honor the 00002003 value for the game on May 1, 2003—despite this logic being added to the signal source post-sale. Since the ticket is general admission and has no assigned row or seat, seating code 001 has been defined as the GA seating section for students, with 001 being the first of up to 540 tickets which may be sold. Another patron who got a similar GA ticket, but after the date had been announced might have a UUID of 1,3,1,05012003,001,213. The concept of unique values being used in the encoding of the barcode, but, subject to audit and modification *ex post facto* by the ticket server is now plainly disseminated herein for those unskilled in the art; with the necessary similarities so predicated to emulate the environment and flexibility paper tickets have enjoyed for over a century.

417. The application continues to discuss how different parameters of virtual tickets have unique identifiers used in the signal source; which now reads plainly as a narrative describing plaintiffs ticketing exhibits. “Signal source identifier **304** can have a second portion and a third portion for storing labels for tiered services. Service provider **314** may issue virtual tickets that are customized based on locations having tiered granularities. Each of the second portion and third portion of signal source identifier **304** can represent a tier. Each tier can have a different geographic granularity. For example, service provider **314** may have multiple physical presences in multiple regions. Signal source identifier **304** can have a second portion and a third portion for storing information related to the multiple regions and multiple physical presences. The second portion of signal source identifier **304** can store label **308** that corresponds to region **316** (e.g., California) where service provider **314** has one or more physical presences. The third portion of signal source identifier **304** can store label **310** that corresponds to physical presence **318** (e.g., food truck **214**) located in the region. Labels **308** and **310** can cause mobile device **102** to present different virtual passes at different locations. For example, mobile device **102** can present a store card valid in shops in California upon detecting a signal source identifier that includes label **308** representing California.”

418. **Claim 22**. A non-transitory computer-readable medium coupled to a mobile device, the non-transitory computer-readable medium storing instructions operable to cause the mobile device to perform operations comprising: receiving a virtual ticket, the ticket comprising a signal source identifier and a message for accessing a service of a service provider, the signal source identifier identifying a signal source being associated with the service provider; in response to an input requesting access to the service, providing, by an output device of the mobile

device, a representation of the message to the service provider, wherein providing the representation of the message comprises: generating a barcode image from the message; and providing the barcode image as the representation for display on a display surface of the mobile device.

419. Herein this claim's reinforcing the primary method; further reinforcing with ambiguity (for the application) that the mobile device performing the ticket redemption operation on its display surface receives a signal source from the ticket server—which necessarily displays the barcode generated by the virtual ticket message. The non-transitory computer-readable medium here is the ticket server. Those unskilled in the art shall take notice this means that the ticket server is not by-itself patentable, however, it works in-conjunction with the mobile devices for the purposes of the novel invention. Since plaintiff discloses the necessary ticket server interaction with mobile devices (which are eligible to generate a barcode of a virtual ticket) in **Exhibit 17,** it proves beyond doubt this claim also mirrors plaintiffs much earlier disclosures.

420. **Claim 25.** The non-transitory computer-readable medium of claim 22, wherein the input comprises: a user activation of a display surface of the mobile device using a home button of the mobile device; a user gesture on a touch-sensitive surface of the mobile device to lock or unlock the touch-sensitive surface; or a user selection, from a quick-access menu, of an option for presenting the message.

421. Plaintiff depicts a user activation of a display surface of a mobile device using a home button of the mobile device using a PDA in **Exhibit 19**. The home button is being used to launch the application which then generates a barcode, which is pictured on the display screen of the device as a *Hamlet* performance. Moreover, **Exhibit 17** depicts a portable computer; which has used keyboard or mouse events initiated by the user to launch an application, which also generated a virtual ticket barcode on the display for redemption at the *Hamlet* performance.

422. **Claim 26**. The non-transitory computer-readable medium of claim 22, wherein the barcode image includes a linear barcode or a two-dimensional barcode.

423. The linear barcodes already disclosed and previously interrogated also apply to this non-transitory computer-readable medium claim.

424. **Claim 27.** The non-transitory computer-readable medium of claim 22, wherein: the ticket is associated with a timestamp specifying a time the service is available, and the operations comprise, before providing the representation of the message, confirming that a current time is within a time window that is determined based on the timestamp, wherein

providing the representation of the message occurs if the current time is within the time window.

425. Herein the virtual ticket being associated with a timestamp that's used for validating the redemption (when presented on the display of a mobile device as a barcode) has been previously interrogated. This claim adds ambiguity to the application; by reinforcing that both the mobile device and the applications which may communicate with a validation authority are capable of interpreting and processing timestamps. The ticket server being depicted as communicating with a mobile device presenting a virtual ticket for a *Hamlet* performance in **Exhibit 17** visually represents the purpose of this claim.

COUNT 9 Patent 10,104,495

Location-based ticket books

426. *The '495 patent includes the following claims plaintiff invented, specifically 6, 7, 8, 9, 16, 17, 18, 22, 25, 26 and 27 as listed below*. Evidence is supported by **Exhibit 7**, **Exhibit 17**, **Exhibit 18**, **Exhibit 19**, **Exhibit 20** and **Exhibit 21**.

427. As stated at **375**, **Exhibits 17-21** were created January 7, 2003 and well before plaintiff's 2006 employment at Apple, which later saw this '495 application submitted a decade later. Plaintiffs notes represent a crucial narrative to corroborate plaintiff's pre-employment Apple IPA in **Exhibit 7**. Given lab notebooks are admissible, the decade between its creation and Apple's patent application nonjoinder of plaintiff further reinforces his previous claim.

428. **Claim 6**. The method of claim 1, wherein the input comprises: a user activation of a display surface of the mobile device using a home button of the mobile device; a user gesture on a touch-sensitive surface of the mobile device to lock or unlock the touch sensitive surface; or a user selection, from a quick access menu, an option for presenting the message.

429. Plaintiffs novel invention for digitally redeeming tickets involves a user activating a display of a digital electronics device and actuating an instruction which causes the processor to present messaging onscreen which includes a linear barcode. This barcode is presented for redemption using the display of the electronic device. Such messaging is depicted in **Exhibit 17** and again with various other embodiments in **Exhibit 19**.

430. Additionally, the example digital redemption device used for auditing the digital ticket embedded in the linear barcode of the patron's device in **Exhibit 21** features a home button on a mobile device. Messaging *in re* the ticket book redemptions success, failure or being invalid is presented on the display surface of a mobile device. One example of such a device are PDA's, which were common before the invention of Apple's iPhone, or even touch capacitance.

431. **Claim 7**. The method of claim 1, wherein providing the representation of the first message comprises: generating a barcode image from the first message; and providing the barcode image as the representation for display on a display surface of the mobile device.

432. As discussed in the preceding claim, providing a barcode image as the representation for display on a display surface of the mobile device is depicted in plaintiffs **Exhibit 17** and **Exhibit 19**.

433. Generation of the barcode image itself is depicted in the block diagram of **Exhibit 17**. Plaintiff explains the "barcode generated when ticket sold, so it is unique." Moreover, a processor and server being used to not only generate the unique barcode image, but also in the redemption itself is demonstrated in the connected block diagrams. The ticket server is connected to redemption events and validation.

434. **Claim 8**. The method of claim 1, wherein the first ticket is a ticket for boarding a vehicle or attending an event, a store card, a boarding pass, a movie ticket, a loyalty card, an employee pass, a gymnasium access pass, a library card, a discount coupon, a retail coupon, or another kind of coupon.

435. **Exhibit 17** depicts a digital, linear barcode for a production of *Hamlet*. A secondary example for *Hamlet* using a different kind of digital mobile device is presented in **Exhibit 19**, along with a digital, liner barcode for a college baseball sporting event. **Exhibit 21** discusses using linear barcodes for a plurality of other event and venue types, including conferences, festivals, performing arts and sporting events. These are featured under the "Example Redemption UIDs" section.

436. **Claim 9**. The method of claim 1, wherein: the first ticket is associated with a timestamp specifying a time the first service is available, and the method comprises, before providing the representation of the first message, confirming that a current time is within a time window that is determined based on the timestamp, wherein providing the representation of the message occurs if the current time is within the time window.

437. Validating timestamps is a critical element of enforcing digital tickets. The linear barcode in plaintiffs much earlier novel invention. Plaintiff discusses this at **394**; using the example of how the barcode information was encoded so that two different performances of *Hamlet* at different times of the day were mutually exclusive. Moreover, a timestamps element in the ticket audit process was discussed at **406**. The ability to re-issue or change the logic in a barcode to accommodate a different timestamp later was discussed at **413**.

438. **Claim 16**. The system of claim 13, wherein the input comprises: a user activation of a display surface of the mobile device using a home button of the mobile device; a user gesture on a touch-sensitive surface of the mobile device to lock or unlock the touch sensitive surface; or a user selection, from a quick access menu, an option for presenting the message.

439. This claim simply describes the systemic aspect of claim 7, which has already been interrogated. Plaintiff explains the systemic concept of using a digital device to display a linear barcode; which has been generated using a processor and in-concert with the logic that's been established for the event using a digital ticket book. The block diagrams in **Exhibit 17** show the systemic process in great detail. All of the digital devices depicted by plaintiff in **Exhibit 17** and **Exhibit 19** utilize the equivalent of a home button and processor—which interpret user actuation to launch a linear barcode on the display surface.

440. **Claim 17**. The system of claim 13, wherein providing the representation of the first message comprises: generating a barcode image from the first message; and providing the barcode image as the representation for display on a display surface of the mobile device.

441. This claim also describes the systemic aspect of claim 7, which has already been interrogated. Plaintiff explains the systemic concept of using a digital device to display a linear barcode; which has been generated using a processor and in-concert with the logic that's been established for the event using a digital ticket book. The block diagrams in **Exhibit 17** show the systemic process in great detail. Herein, this claim focuses on presenting the linear barcode itself on the digital device display, as opposed to the (broad) ability to present messaging for digital tickets using the device itself in claim 16.

442. **Claim 18**. The system of claim 13, wherein: the first ticket is associated with a timestamp specifying a time the first service is available, and the operations comprise, before providing the representation of the first message, confirming that a current time is within a time window that is determined based on the timestamp, wherein providing the representation of the message occurs if the current time is within the time window.

443. The systemic ability and function of embedding and using timestamps in linear barcodes for ticket books was previously interrogated in great detail at **394**, **406**, **413** and **437**.

444. **Claim 22**. A non-transitory computer-readable medium coupled to a mobile device, the non-transitory computer readable medium storing instructions operable to cause the mobile device to perform operations comprising: accessing a storage medium storing one or more tickets, each ticket including a respective signal source identifier and a respective message for

accessing a service of a respective service provider, the signal source identifier identifying a signal source being associated with the service provider; detecting a first signal source identifier based on data received from a first signal source by a wireless subsystem of the mobile device, the wireless subsystem executing a procedure for monitoring wireless signals from signal sources using a wireless processor of the mobile device, the first signal source identifier comprising at least two portions; determining that one of the at least two portions represents a first location having a first geographic granularity and another of the at least two portions represents a second location having a second geographic granularity; determining, based on the data received from the first signal source, that the mobile device is located within a communication range of the signal source; identifying, among the one or more stored tickets, a first ticket that includes a signal source identifier that matches the geographic granularity of at least one of the two portions and includes a first message for accessing a first service of a first service provider; and then in response to an input requesting access to the first service, providing, by an output device of the mobile device, a representation of the first message to the first service provider.

445. The method and system for establishing a digital ticket has now been heavily interrogated in previous claims. The purpose of this claim is differentiating geographical proximity to the user's device presenting the barcode for redemption. This problem is explained in **Exhibit 21**, wherein plaintiff lists longstanding with paper tickets and had to be solved by plaintiff before any novel solution could be realized, and, before GPS was physically available in a digital device, such as a mobile phone. Fortunately, plaintiff recorded extensive detail *in re* geography of event redemption areas. Nine different venues at different physical locations at Cal State Fullerton are listed, along with an example representation of each venue being encoded as a universal identifier. This ensures the signal source identifier in Apple's cribbed application matches the location which the user is attempting to present the digital ticket. A redemption device at Goodwin Field will not accept a digital ticket barcode for a performance of *Hamlet* at the Little Theater, even if the event time is the same as the baseball game scheduled. Since Apple's struggled mightily to understand how its own Passbook feature works in earlier pleadings, a less technical example are airport plane tickets. Several planes may leave from the same gate at the same airport each day, with only different timestamps. Each plane's going to a much different geographical location. This data is encoded as part of the destinations in the flight number of the user's ticket. It's thus necessary to encode the correct geography data in the barcode, just as paper tickets list each connecting flight and final destination. We also must

ensure any connecting flights in different geographical locations are correct. If a passenger deplaned at San Francisco instead of Reno on their way to Chicago, neither a paper, nor digital barcode would allow this passenger to then board a flight to Chicago from San Francisco. In this sense, GPS is not always reliable with precision inside an airport or university yet was still solved by plaintiff before they existed in mobile devices.

446. This is why Apple's application relies on a secondary signal source identifier when considering geographic proximity. Moreover, this is why plaintiff's invention also allows linear barcodes to be printed on paper as a backup alternative, and, for patrons who don't own a digital device but still have purchased tickets from the same issuing authority who offers digital tickets. Most airport passengers still use paper tickets today, which utilize plaintiff's invention and are commonly used when a flight or seat change occurs at the gate. This same barcode ensures the geographical departure location matches what's expected by the redemption device in the same manner as it does when presented on a display surface. This is why GPS cannot be used exclusively for linear barcodes and only presents a casual convenience for the user; by not displaying other tickets for redemption which aren't located near the user. GPS is also subject to signal availability—coupled with its difficulty resolving precise determinations for multiple ticketing scenarios, which paper tickets could already avoid with certainty. Anybody who's flown into England or Ireland's familiar with frequent gate changes and gates not being announced until boarding. These issues are still solved using plaintiff's novel invention, which allows for such granularity; even when used on paper and not electronically. In such examples, the flight number and airport location are encoded into the barcode. Having an issuing authority allows for real-time changes after a ticket may have been encoded, which plaintiff illustrates in the block diagram in **Exhibit 17**. The redemption problems discussed herein are also explained from the audit perspective in **Exhibit 20** and **Exhibit 21**. The problem of a patron presenting their digital ticket at the wrong gate. but at the correct airport is shown with the "valid, error or not valid" choices on the display surface of the redemption device scanning the ticket book. While multiple aspects of the event are audited, the approval beep heard when boarding aircraft in today's airports largely is indicating that the location identifier (among other things, like timestamp) is correct. When a digital ticket doesn't redeem when scanned at the gate, it's typically because the wrong flight/gate was visited and not because the passenger arrived at the wrong airport, or, a day before or after their scheduled flight. Plaintiff solved this longstanding problem by replacing human-readable text requiring exactly this to be replaced with a barcode that audits the information automatically.

This is why airline and concert tickets still printed on paper can be scanned with the same accuracy, precision and speed as those presented on the display surface of a digital device; with the same confidence in the interpreted result. It should be remembered that one longstanding problem plaintiff had to solve was reducing the labor required to manage entry for a ticketed event, as depicted in **Exhibit 20**. Today, the plaintiff's model has been adopted by most airlines. One gate attendant can ensure hundreds of passengers are correctly boarded without needing more staff, time and introducing human audit error. Now, the passenger who mistakenly boards the wrong flight is detected before they ever board the aircraft and are finally discovered; when two people have the same seat…or, they arrive at the wrong city if the seat wasn't already sold. While Apple (sadly) doesn't understand this in this matter, the world's airlines and plaintiff do.

447. **Claim 25**. The non-transitory computer-readable medium of claim 22, wherein the input comprises: a user activation of a display surface of the mobile device using a home button of the mobile device; a user gesture on a touch-sensitive surface of the mobile device to lock or unlock the touch sensitive surface; or a user selection, from a quick access menu, an option for presenting the message.

448. This non-transitory claim is for ambiguity. The use of a digital device to present messaging related to digital ticket books has already been interrogated in great detail, particularly in **428-430** and in the '14 patent narrative.

449. **Claim 26**. The non-transitory computer-readable medium of claim 22, wherein providing the representation of the first message comprises: generating a barcode image from the first message; and providing the barcode image as the representation for display on a display surface of the mobile device.

450. This non-transitory claim is for ambiguity. Generating and presenting a barcode image related to digital ticket books has already been interrogated in great detail, particularly in **429-435**, **439-441** and in the '14 patent narrative.

451. **Claim 27**. The non-transitory computer-readable medium of claim 22, wherein: the first ticket is associated with a timestamp specifying a time the first service is available, and the operations comprise, before providing the representation of the first message, confirming that a current time is within a time window that is determined based on the timestamp, wherein providing the representation of the message occurs if the current time is within the time window.

452. This non-transitory claim is for application ambiguity. Determining when an event or service is available using a timestamp and then verifying that the current time matches the data

encoded in the linear barcode has already been interrogated in great detail, particularly in **394**, **406**, **413**, **436**, **437**, **442** and **443**.

Count 10 Patent 9,037,513

System and method for providing electronic event tickets

453. *The '513 patent includes the following claims plaintiff invented, specifically 6, 7, 8, 9, 10, 14, 15, 17, 19, 20 and 21 as listed below*. Evidence is supported by **Exhibit 7**, **Exhibit 17**, **Exhibit 18**, **Exhibit 19**, **Exhibit 20** and **Exhibit 21**.

454. As stated at **375** and **427**, **Exhibits 17-21** were created January 7, 2003 and well before plaintiff's 2006 employment at Apple, which later saw this '13 application submitted a decade later. Plaintiffs notes represent a crucial narrative to corroborate plaintiff's pre-employment Apple IPA in **Exhibit 7**. Given lab notebooks are admissible, the decade between its creation and Apple's patent application nonjoinder of plaintiff further reinforces his previous claim.

455. **Claim 6**. The method of claim 1, wherein the electronic device comprises a handheld device and the handheld device comprises a portable phone.

456. As discussed at **429** and **430**, plaintiff's novel invention for digitally redeeming tickets involves a user activating a display of a digital electronics device and actuating an instruction which causes the processor to present messaging onscreen which includes a linear barcode. This barcode is presented for redemption using the display of the electronic device. Examples are depicted in **Exhibit 17** and again with various other embodiments and a different digital device in **Exhibit 19**. A portable phone wasn't capable of housing a computer processor and display surface which could present liner barcodes. This was the principal reason Apple's CEO initially tabled plaintiffs' idea; as PDA and laptop computers comprised the bulk of devices which could support the resolution necessary. No mobile phone in existence when invented could yet support the technical requirements necessary. Some handheld devices (such as PDAs) could support such functionality and are depicted in **Exhibit 17** and **Exhibit 19**.

457. **Claim 7**. The method of claim 1, wherein the electronic device comprises a handheld device, the handheld device having a height less than approximately 5.0 inches, a width less than approximately 2.5 inches, and a depth less than approximately 0.5 inches.

458. PDA's in-use at the time of plaintiff's invention resembled the approximate size of a handheld device described in claim 7.

459. **Claim 8**. The method of claim 1, wherein the electronic device comprises a

handheld device, the handheld device weighing less than approximately 5.0 ounces.

460. The weight of most PDA's and even digital music players was near this threshold during the time of plaintiff's invention—some weighed up to 16 ounces and had the ability to run the mobile equivalent of a full-size computers operating system, such as Microsoft Windows. Apple's music players at this time weighed more than 5 ounces; some iPhones today still weigh more. The weight has no relevance for presenting digital tickets on a display surface and varies considerably based on the size/type of the battery.

461. **Claim 9**. The method of claim 1, wherein the ticket is an electronic ticket stored on a physical ticket and the physical ticket is configured to transmit the electronic ticket to the electronic device after the physical ticket is tapped to the near field communication interface of the electronic device.

462. Plaintiffs invention covers both electronic tickets and physical tickets. Both use a linear barcode to present for redemption using either the image on the display surface of a digital device, or via printed means. Several additional embodiments disclose using a ticket authority on a remote server to validate, invalidate or edit barcodes already generated as mobile ticket books. Payment and purchase system interactions with electronic and physical digital ticket books are also disclosed in **Exhibit 17**. The barcode in plaintiff's invention has logic for ensuring the correct geographic location and date + timestamp is enforced, with the flexibility to make changes post sale if circumstances require. Plaintiff discloses examples of electronic and physical tickets ready for redemption, as well as disclosing methods and apparatuses to audit and redeem both types. Moreover, both electronic and physical tickets work in-conjunction together using unique barcodes, which allows for flexibility and convenience for redemption sites. Both electronic and physical tickets have a barcode scanned identically by the same redemption site. This allows a ticket to be modified or transferred to another patron, irrespective of whether they have a mobile device to use for redemption.

One example of the interaction in this claim as invented by plaintiff doesn't require near field communication. An event could be postponed for later in the season for weather reasons. Some fans may opt to use their original electronic or physical tickets at the later date. Their tickets are modified by the redemption authority to work on the new date, without barcodes needing to be regenerated. Some fans may opt to sell or transfer their electronic or physical tickets to another party, including transferring them back to the seller for a refund. In each instance, the electronic or physical barcode is transferred to another party, with the new barcode being re-generated in some cases to prevent fraud when re-sold; mostly in cases where the seller is issuing a refund and then relisting the ticket for sale. In all cases, tickets can be transferred from electronic to physical mediums, using the same universal identifier encoding. In this claim, a physical ticket is generated and issued electronically to a digital device, which is then presenting the barcode image on the display surface when prompted by the user. Using NFC to transfer the ticket book is no different than scanning the tickets using a barcode reader or camera. The inherent method and process remain identical.

463. **Claim 10**. The method of claim 1, wherein the event comprises a concert.

464. Plaintiff depicts a concert in **Exhibit 21**, wherein a festival is shown as a unique identifier type. A concert was the example former CEO Jobs used with plaintiff when he presented the idea to him. Two of the theatres depicted and the associated student union venue regularly host concerts.

465. **Claim 14**. The method of claim 1, comprising displaying information from the ticket on a screen of the electronic device.

466. Plaintiff clearly depicts displaying information from the ticket on a screen of the electronic device in **Exhibit 17**, **Exhibit 19** and **Exhibit 21**.

467. **Claim 15**. The method of claim 1, wherein the electronic device comprises a handheld device and the handheld device comprises a portable media player.

468. Plaintiff clearly depicts both a handheld device and a portable media player in **Exhibit 17** and **Exhibit 19**.

469. **Claim 17**. The method of claim 1, wherein the event comprises a play.

470. Plaintiff clearly depicts displaying information from the ticket of *Hamlet* in a screen of the electronic device in **Exhibit 17** and again in **Exhibit 19**.

471. **Claim 19**. The method of claim 1, wherein the event comprises an opera.

472. Plaintiff clearly depicts two theatre types in **Exhibit 21**. Both of the listed venues at Cal State Fullerton (Little and Big Theater) have hosted operas.

473. **Claim 20**. The method of claim 1, wherein the event comprises a sporting event.

474. Plaintiff clearly depicts baseball, basketball, soccer, softball and volleyball as event types in **Exhibit 21**. Plaintiff clearly depicts a physical ticket for an NCAA baseball game against Univ. of Pacific and also an electronic ticket for an NCAA Super Regional baseball tournament at Goodwin Field in **Exhibit 19**.

475. **Claim 21**. The method of claim 1, wherein the event comprises a school-related

event.

476. Plaintiff clearly depicts a plurality of university events—featuring concerts, performing arts, and both men's and women's NCAA athletics in **Exhibit 17**, **Exhibit 19**, **Exhibit 20** and **Exhibit 21**.

Count 11 Patent 9,277,530

Delivery of push notifications to an inactive computing device

477. *The '530 patent includes the following claims plaintiff invented, specifically 10, 14, 15, 18, 20 and 23 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

478. Despite the focus and title, this patent discloses necessary art plaintiff invented pertaining to reliably finding a lost smartphone or device. The term "lost device" is found directly in four instances throughout the application.

479. In **Background**, it states, "Examples of such applications are calendar app, contacts app, image library organizer app, *lost-device locator app*, voice over Internet protocol (VoIP) app, video conference app, etc." Herein, plaintiff's invention is represented by the lost-device locator app, which utilizes the cloud server to allow for finding devices location and remote command event execution. Note using a web page to connect to the cloud server's no different than using the same login credentials with a lost-device locator app. They equally are capable of enabling lost "discovery" mode on the true owner's device and executing remote commands that use notifications to communicate with the typically inactive, lost device.

480. In **Summary**, it states, "Examples of such non-red-listed apps are *lost-device locator app*, voice over Internet protocol (VoIP) app, video conference app, etc." The use of "red-listing" an application simply entails *not* preemptively securing their limited usage when lost "discovery" mode has been enabled by the true owner—using either a lost-device locator app or a web page to enforce the restriction on the lost device. This allows for the telephony app to be used to contact the true owners unique contact number by an honest finder, as well as for accessing the device and reporting its geographical movement while the display is inactive, and, without the knowledge of a thief; if the device was stolen and not misplaced. Thus, a lost-device locator app is added to a special "white" list, to ensure its special availability; when lost "discovery" mode has been enabled.

481. In **Detailed Description Overview**, it states, "Examples of apps installed on the device that typically are not red-listed are *lost-device locator app*, voice over Internet protocol

(VoIP) app, video conference app, etc." As discussed *supra*, the lost-device locator app and telephony apps aren't red-listed, so as to ensure their usage when lost "discovery" mode has been enabled by the true owner.

482. In **Activating Recipient Device to Deliver Push Notifications Using Notification Handler Run by Recipient Device**, it states, "As such, the app **158** a is referred to as a red-listed app. A second category of apps, e.g., *lost-device locator app*, voice over Internet protocol (VoIP) app, video conference app, etc., is such that delivery of push notifications for an app **158** b from the second category would cause the recipient device **150** to transition from the inactive state into the active state prompting the user to interact with the app **158** b. Identifiers of the apps from the second category are left out of the red list **154**." Herein, the notification server is the cloud-based server being used to manage lost devices. The only way the inactive device can become active for an honest-finder to attempt reaching the true owner is for such processes to not be added to a red-list.

483. **Claim 10**. The method of claim 1, wherein the recipient device is in the inactive state when at least a display of the recipient device is dark while the recipient device is running on battery power.

484. Plaintiff discusses in great length the benefits of maximizing the battery life of the lost device; in order to increase the opportunity for the true owner to locate the device. In addition to **Exhibit 8**, see **165**, **170**, **185**, **208-211**, **241** and **244-248**.

485. **Claim 14**. A computing device comprising: one or more hardware processors; and non-transitory computer readable medium encoding instructions that, when executed by the one or more hardware processors, cause the one or more hardware processors to emulate a notification handler that performs operations comprising: receiving a first push notification for a first application while the computing device is in an inactive state; determining that delivery of the first push notification would cause the computing device to transition into an active state without prompting a user associated with the computing device to interact with the first application, wherein the operation of determining that delivery of a push notification would cause the computing device to transition into the active state without prompting the user to interact with the first application comprises: parsing a record of restricted application identifiers stored at the computing device, the restricted application identifiers corresponding to applications installed on the computing device for which delivery of push notifications would cause the computing device to transition into the active state without prompting the user to interact with the respective

installed application, and finding an identifier of the first application among the restricted application identifiers of the stored record; in response to determining that delivery of a push notification would cause the computing device to transition into the active state without prompting the user to interact with a particular application, storing the first push notification at the computing device; receiving a second push notification for a second application while the computing device continues to be in the inactive state; determining that delivery of the second push notification would cause the computing device to transition into the active state prompting the user to interact with the second application; and in response to determining that delivery of the second push notification would cause the computing device to transition into the active state prompting the user to interact with the second application, delivering the received second push notification to the second application and the stored first push notification to the first application.

486. Causing a computing device to transition into an active state without prompting a user associated with the computing device to interact with the first application is the only method to ensure both that the location of a lost device can continue for recording geographic charted timestamps while a thief has possession, as well as allowing an honest finder to contact the true owner if found. It has the benefit of reducing battery drain by only allowing a few processes and applications to execute when the device is in lost "discovery" mode, as characterized by plaintiffs **Exhibit 8** and also **165**, **170**, **185**, **208-211**, **241** and **244-248**. This is why plaintiff described using the battery sparingly until it's depleted for sending location updates, while trying indefinitely if it's detected that a power adaptor is connected.

487. Parsing a record of restricted application identifiers stored at the computing device, the restricted application identifiers corresponding to applications installed on the computing device for which delivery of push notifications would cause the computing device to transition into the active state without prompting the user to interact with the respective installed application, and finding an identifier of the first application among the restricted application identifiers of the stored record is depicted with identifiers communicating with the server in **Exhibit 12**.

488. Storing the first push notification at the computing device; receiving a second push notification for a second application while the computing device continues to be in the inactive state (determining that delivery of the second push notification would cause the computing device to transition into the active state prompting the user to interact with the second application) is shown by plaintiff in the example lock screen with messaging for an honest finder to reach the

true owner in **Exhibit 12**. Image 1 & 3 of the application show this same process outlined.

489. **Claim 15**. The computing device of claim 14, wherein the operation of determining that delivery of a push notification would cause the computing device to transition into the active state prompting the user to interact with the second application comprises parsing the record of restricted application identifiers stored at the computing device without finding the identifier of the second application among the restricted application identifiers of the stored record.

490. Parsing restricted application identifiers is accomplished by user record mapping against the cloud or notification server in **Exhibit 10** and **Exhibit 12**. Similarly, if an honest finder pressed an operative button to interact with the device, it would cause an active state to present the lost messaging screen and telephony application process on the device display—and because the first push notification has already enabled lost "discovery" mode on the device, thus causing the restricted application identifiers to be enforced. A less technical explanation is that the device doesn't begin to enforce the restricted application identifiers until being declared lost by the true owner and lost "discovery" mode is enabled using the notification server. This claim is explaining that when the device is in normal operation, it's not necessary to enable the restricted application identifiers, while explaining how its necessarily enabled. In theory, while not as practical, another application (not for finding lost devices) could send a similar instruction to an inactive device using a cloud server to manage such notifications. This is why a VOIP application is another example cited by the application.

491. Communicating with an honest finder follows this process as outlined by plaintiff generally, as even their intervention with the buttons on the physical device has already been predicated by restricted application identifiers. Otherwise, the honest finder could potentially activate other applications or processes on the device; including access to the contents of the device if no passcode had been enabled by the true owner.

492. **Claim 18**. The computing device of claim 16, wherein the operations further comprise transmitting, via a public IP connection to a notification server associated with the notification handler process, a copy of the record of restricted application identifiers generated by the computing device, such that the notification server can (i) temporarily store, at the notification server, push notifications for applications installed on the computing device that have application identifiers included on the copy of the record of restricted application identifiers, and (ii) opportunistically push the stored notifications along with a notification for an application

installed on the computing device that has an unrestricted application identifier.

493. The notification server represents the conduit which a public IP connection is derived by the device, with appropriate notification handlers. This is explained in the process of **Exhibit 8**, the connection path and user record mapping in **Exhibit 10** and in practical operation in **Exhibit 12**. It's important to note opportunistically pushing notifications can also occur as a result of executing remote command instructions, as previously detailed in previous patents.

494. **Claim 20**. The computing device of claim 14, wherein the operations further comprise removing the record of restricted application identifiers from the computing device's storage upon detecting that the computing device transitions from the inactive state to the active state.

495. When the true owner of a device disables lost "discovery" mode, it requires that the cloud server depicted in plaintiffs notes signal to the lost device that applications and regular functionality can now be re-established. Herein, the list of application identifiers is thus no longer enforced and normal operation is restored to the active device. This also restores normal notification service for applications and the OS itself, as such functionality is (primarily, but not limited) to locating the device geographically and communicating with an honest finder.

496. **Claim 23**. The computing device of claim 14, wherein the computing device is in the inactive state when at least a display of the computing device is dark while the computing device is running on battery power.

497. This is the state most devices are in when declared lost by the true owner and discussed in **Exhibit 8**, with an example depicted in **Exhibit 9**. The iPhone is obviously operating on battery power while moving through three cities after being declared lost. Even the "Example Process UI" generally assumes the device is inactive when the true owners attempting to locate it.

Count 12 Patent 8,670,748

Remotely locating and commanding a mobile device

498. *The '748 patent includes the following claims plaintiff invented, specifically 1, 5, 6, 8, 9, 11, 12, 14, 16, 17 and 28 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

499. **Claim 1**. A computer-implemented method performed by a specified mobile device, the method comprising: accessing, by the specified mobile device, a notification service on a server separate from the specified mobile device, the notification service hosting a plurality

of command collection topics, where each of the plurality of command collection topics is subscribed to by a unique mobile device; accessing, by the specified mobile device, one of the plurality of command collection topics that is subscribed to by the specified mobile device, the accessed command collection topic including a plurality of command nodes, each corresponding to a remote command type; polling, by the specified mobile device, each of the plurality of command nodes of the accessed command collection topic to determine whether one or more new remote command messages have been received by the accessed command collection topic; retrieving, by the specified mobile device, from a remote lock command node included in the plurality of command nodes, a remote lock command message, where the remote lock command message comprises a lock command and a locking passcode; locking, by the specified mobile device, the specified mobile device using the locking passcode in response to the lock command; setting, by the specified mobile device, an unlock passcode associated with the specified mobile device; and generating, by the specified mobile device, an acknowledgement message in response to the remote lock command message.

500. The cloud server depicted and described in plaintiffs **Exhibit 1**, **Exhibit 8**, **Exhibit 9** (which shows the server login page) and also directly shown in **Exhibit 10** is, “a notification service on a server separate from the specified mobile device.” **Exhibit 12** shows the server in a block diagram. The cloud-based recovery server handles a plurality of programmatic functions for accomplishing the goal of reliably locating a lost device and giving an honest finder a reasonable opportunity to contact the owner. The notifications discussed throughout this patent generally relate to notifications for the lost device or server relating to the dynamic location of the device, including the remote command functions the true owner may execute using another computer (or device) logged into the server with the same credentials being used on the lost device when lost “discovery” mode’s actuated.

501. A plurality of command collection topics relate to functions a true owner may utilize after declaring their device lost and using the cloud server to actuate lost “discovery” mode. An example function is shown in **Exhibit 12**, with the server in a block diagram, and, an example remote command shown in an example screenshot—the ability to allow an honest finder to call the owner using the device. **Exhibit 9** also shows a remote command notification in-action—the dynamic traveling location of the lost device is presented on a map in real time.

502. Moreover, **Exhibit 9** shows the remote lock command message comprises a lock command and a locking passcode; locking, by the specified mobile device, the specified mobile

device using the locking passcode in response to the lock command; setting, by the specified mobile device, an unlock passcode associated with the specified mobile device; and generating, by the specified mobile device, an acknowledgement message in response to the remote lock command message. Under "Example Process UI" is depicted a function to locate and lock a missing device, with the corresponding example UI for the lost device shown in Exhibit 12, noting the field explicitly for a locking passcode. **Exhibit 11** is mostly concerned with a narrative of how to lock and unlock a device when stolen. The first note by plaintiff reads, "we could lock the device and invalidate the passcode while privileged mode is in-use." Even a Hollywood film doesn't contain the element of copy Apple's misjoinder applications do, especially in yet another example herein. It's clear to an objective bystander not skilled in the art that it was largely impossible to file this and the other applications without working from a photocopy of plaintiffs notes—which were available to many Apple employees, including the two executives responsible for implementing the feature, and, who had no knowledge of this concept, idea or implementation before plaintiffs disclosures in **Exhibit 3** and **Exhibit 5**.

503. **Claim 5**. The computer-implemented method of claim 1, wherein generating an acknowledgement message further comprises: including a time stamp indicating a time at which the remote lock command message was retrieved.

504. The importance of using a remote command message with a time stamp indicating (for example) when a lost device was physically at given locations on a map was disclosed by plaintiff as discussed at **67**, **111**, **159**, **163**, **214-216**, **218**, **234**, **252**, **257-260**, **290**, **361**, **363**, Using a time stamps for querying battery life operations was disclosed by plaintiff at **170**, **173**, **208**, **210** and **230**.

505. **Claim 6**. The computer-implemented method of claim 1, wherein locking the specified mobile device further comprises: locking a display associated with the specified mobile device such that access to one or more of information stored on the specified mobile device and functionality of the specified mobile device is blocked.

506. Plaintiffs narrative explaining his function of locking access to information stored on the specified mobile device and limiting (or otherwise suppressing) its functionality has been *extensively* discussed throughout the complaint, including at **150**, **172**, **175**, **177**, **183**, **187**, **206**, **264**, **275** and **281**.

507. **Claim 8**. A non-transitory computer-readable medium, tangibly encoding a computer program product comprising instructions operable to cause a data processing apparatus

comprised within a specified mobile device to perform operations comprising: accessing, by the data processing apparatus, a notification service on a server separate from the mobile device, the notification service hosting a plurality of command collection topics, where each of the plurality of command collection topics is subscribed to by a unique mobile device; accessing, by the data processing apparatus, one of the plurality of command collection topics that is subscribed to by the specified mobile device the accessed command collection topic including a plurality of command nodes, each corresponding to a remote command type; polling, by the data processing apparatus, each of the plurality of command nodes of the accessed command collection topic to determine whether one or more new remote command messages have been received by the accessed command collection topic; retrieving, by the data processing apparatus, from a remote lock command node included in the plurality of command nodes, a remote lock command; locking, by the data processing apparatus, the specified mobile device in response to a lock command included in the remote lock command message; and publishing, by the data processing apparatus, an acknowledgement message to the notification service.

508. A notification service on a server separate from the mobile device, the notification service hosting a plurality of command collection topics, where each of the plurality of command collection topics is subscribed to by a unique mobile device; accessing, by the data processing apparatus, one of the plurality of command collection topics that is subscribed to by the specified mobile device the accessed command collection topic including a plurality of command nodes, each corresponding to a remote command type is depicted and/or disclosed by plaintiff in **Exhibit 1**, **Exhibit 3**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10** (wherein the server is depicted being separate from mobile devices in an example flow diagram) **Exhibit 11** and finally **Exhibit 12**, which features a beforementioned flow diagram depicting the server hosting the notification service with a unique mobile device, and, a remote command type actively being executed in the "Example Lock Screen When Lost" UI.

509. **Claim 9**. The computer-readable medium of claim 8, further operable to cause data processing apparatus to perform operations comprising: identifying a passcode specified by the remote lock command message; detecting that the specified passcode does not comply with a security constraint implemented by the specified mobile device; and determining, in response to the detecting, not to reset an unlock passcode associated with the specified mobile device.

510. Enforcing a passcode as specified by a remote command instruction has already been interrogated in previous claims, with examples at **44**, **114**, **157**, **172**, **175**, **180**, **181**, **264**,

275, **278**, **285**, **297** and **348**.

511. **Claim 11**. The computer-readable medium of claim 8, further operable to cause data processing apparatus to perform operations comprising: locking the specified mobile device by locking a display such that access to one or more of information stored on the specified mobile device and functionality of the specified mobile device is blocked.

512. This is the computer-readable version of **Claim 8** that was added for patent application ambiguity and was interrogated *supra* at **508**.

513. Moreover, when lost "discovery" mode has been enabled via the cloud server, it locks the device until it's been disabled. This prevents both an honest finder and thief alike to be unable to access the information stored on the specified mobile device, or, enable functionality on the device. An honest finder would only be able to contact a privileged contact that's been enabled previously. No other functionality can be realized and even the telephone cannot be used to dial any other number *except* the privileged contact. While lost "discovery" mode's enabled, the device is effectively disabled and cannot be restored to escape this restriction. The data processing apparatus on the cloud server enforces this policy and is a solution to the longstanding problem of preventing mobile device theft—the resulting device has no effective value when it's been disabled using a cloud server by the true owner; even if the device was just purchased new the day before.

514. **Claim 12**. The computer-readable medium of claim 8, further operable to cause data processing apparatus to perform operations comprising: including a time stamp in the acknowledgement message indicating a time at which the remote lock command was executed.

515. This is both demonstrated and necessary in order to (using one common, simple example herein) chart the location dynamically of a lost device over time and present a trail showing past static movements. In plaintiff's original embodiment, the lost device is locked when lost "discovery" mode has been enabled by the true owner, using the cloud server listed as a notification server in the application. See **17**, **36**, **67**, **150**, **159**, **160**, **163**, **183**, **201**, **210**, **213-216**, 2**18**, **222**, **234**, **252**, **256-260**, **264**, 2**90**, **361** and **363**.

516. **Claim 14**. The computer-readable medium of claim 8, further operable to cause data processing apparatus to perform operations comprising: establishing a connection to the notification service over a wireless data connection.

517. Plaintiff clearly depicts a computer readable medium establishing a connection to the notification service over a wireless data connection in **Exhibit 10**. Further, plaintiff depicts an

iPhone using both cellular and switched Internet wirelessly to connect to the notification service in separate flow diagrams. **Exhibit 12** also depicts a notification server being connected to a lost iPhone via both cellular (denoted by baseband) and wireless Internet mediums.

518. Plaintiff's **Exhibit 10** clearly depicts establishing a connection to the notification service over a wireless data connection. "iPhone Wi-Fi" is depicted being connected to the Internet and cellular service providers networks and the cloud server; depicted in the application as a notification service.

519. **Claim 16**. A mobile device comprising: processor electronics; a storage medium storing instructions executable by the processor electronics to cause the processor electronics to: access the notification service on a server separate from the mobile device, the notification service including a plurality of command collection topics, where each of the plurality of command collection topics is subscribed to by a unique mobile device; access one of the command collection topics of the plurality of command collection topics hosted on the notification service and subscribed to by the mobile device, the command collection topic subscribed to by the mobile device including a plurality of command nodes, each command node corresponding to a remote command type; poll each of the plurality of command nodes of the accessed command collection topic to determine whether one or more new remote command messages have been received by the accessed command collection topic; retrieve, from a remote lock command node included in the plurality of command nodes, a remote lock command message; open the remote lock command message, the remote lock command message comprising a lock command and a locking passcode; lock the specified mobile device in response to the lock command; set an unlock passcode associated with the specified mobile device to the locking passcode; and publish an acknowledgement message to the notification service.

520. Each of these elements have previously been heavily interrogated *supra*, including throughout previous counts and featuring block diagrams, flow diagrams and example UI's.

521. **Claim 17**. The mobile device of claim 16, wherein the instructions are further executable by the processor electronics to cause the processor electronics to: include in the acknowledgement message an indication confirming that the unlock passcode has been set to the locking passcode and a time stamp identifying a time at which the remote lock command message was retrieved.

522. The lost device that's been placed in "discovery" mode has to acknowledge to the cloud server at what time it was successfully placed in a locked state. This allows for the starting

point for forensic and map location charting by the cloud server, as well as providing a data point for remote command instructions; so, they can be properly executed. The discussion supporting **Claim 12** explains the importance of the timestamp from the missing device being recorded using the cloud server. It's also practical for calculating whether to disable login attempts, or, wipe the memory contents of the lost device—from a predefined amount of incorrect passcode attempts.

Count 13 Patent 9,277,530

Remotely receiving & communicating commands to mobile device

523. *The '7530 patent includes the following claims plaintiff invented, specifically 6, 7, 10, 11, 20, 22 and 23 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

524. Apple filed for this patent on May 30, 2014—which was 5 months before wrongfully terminating plaintiff. It's impossible to argue plaintiff abandoned his much earlier invention, as documented in his lab notebook. Plaintiff was unaware of this patent's existence until a recent trip to the PTO in Alexandria, and, over a year after filing this litigation. Apple failed to join the plaintiff to this application, which thus would've ensured his inclusion on the other phone-finding patents. This last patent plaintiff discovered was the first phone finding patent Apple should have properly enjoined him—while still employed there.

525. **Claim 6**. The method of claim 4, further comprising transmitting, via a public IP connection to a notification server associated with the notification handler process, a copy of the record of restricted application identifiers generated by the recipient device, such that the notification server can (i) temporarily store, at the notification server, push notifications for applications installed on the recipient device that have application identifiers included on the copy of the record of restricted application identifiers, and (ii) opportunistically push the stored notifications along with a notification for an application installed on the recipient device that has an unrestricted application identifier.

526. Herein, this claim (as many others) uses a cloud-based notification server to enable, disable and execute remote commands to a lost device when the true owner's enabled lost "discovery" mode—which uses a pubic IP connection from the cloud server. The application identifiers described are simply used to denote a whitelist when lost "discovery" mode is enabled, necessarily placing all other applications on a blacklist; which continues temporarily until the true owner has deactivated the feature. This allows the lost phone to have the selective use of partial functionality, useful for an honest finder attempting to communicate with the true owner, the

ability to record device location and movement, conserve battery resistance and remotely erase the contents of memory. An unrestricted application identifier (as described in Apple's application) simply pertains to applications which have previously been programmatically set to be on a whitelist. Push notifications aren't novel herein and are listed simply because that's the programmatic method (using the Objective C programming language unknown to the misjoinder inventors) for drawing a message on a device in response to an action. This occurs also without requiring a user notification, which is useful when a remote command instruction is executed to a remote device. It allows for a transaction to be executed without being stuck in a continual event loop, which is helpful for conserving resistance and memory footprint.

527. The use of software industry terms like blacklist and whitelist for referring to rights differentiation is present in many Apple software patents, but, doesn't appear even once in this application. Yet again, Apple was confused and couldn't understand plaintiff's invention (a common theme among the phone-finding patents) notes correctly, just salvaging the overall concept without describing correct detail.

528. This was confusingly declared again by Claim 16 in the patent in count 6. Claim 4 discusses the notification server and the concept of notification events and explains (yet again) why the phone-finding patents took so long to approve and required additional examiners, which is very rare with such simple patents. Again, this stems from misjoinder inventors not understanding the plaintiffs notes they copied, or, how object-oriented programming works.

529. The narratives supporting the patent in count 6 apply herein to this claim and have been interrogated *ad nauseum*—it's not necessary to burden the Court with duplication.

530. **Claim 7**. The method of claim 2, further comprising removing the record of restricted application identifiers from the recipient device's storage upon detecting that the recipient device transitions from an inactive state to an active state.

531. This is simply terminating the event loop which programmatically allows the device to enforce the application whitelist when lost "discovery" mode is active, meaning when the device is found, the instruction from the cloud-server (called a notification server herein) which disables the lost functionality also returns the applications on the device to normal operation. It wouldn't be novel to still have a device which doesn't allow the applications to ever function again after its been found by the true owner. Even the overall concept of ambiguity in patent applications (which adds some duplication of function to other things) questions the need for this claim, however, the one technical reason for this not disclosed by Apple is that a device

which isn’t located (a hopefully rare proposition) needs to remain locked so that its resale value is null, and, so that a determined thief cannot access the application data if they’re able to later compromise the passcode and gain access to the device.

532. Given plaintiff worked [redacted] on issues potentially involving national security interests (not appropriate for disclosure per NDA) it was a realistic concern [redacted] and increasing success of [redacted] could allow application data (normally encrypted when connected to another device) to be accessible and potentially cause unnecessary loss of life, or, serious national security implications. If an iPhone was lost with sensitive information [redacted], they could very easily gain access to classified information.

In plaintiffs’ case, his vehicle was often followed in hopes a prototype would be left unattended, which resulted in security providing escort during certain conditions; even when travelling between Apple buildings. [redacted].

Thus, the application blacklist has critical reason for being enforced in perpetuity; until the true owner deems it found. Apple doesn’t explain that enforcing the application blacklist is a key method of ensuring a device has no resale value even when new; which is a longstanding problem plaintiff solved, among others. Even in a usage case where there was no passcode, having the applications inaccessible can still prevent the device being reimaged and repurposed.

Given plaintiff doesn’t know and never worked with the misjoinder inventors, its beyond doubt they likely had no experience or familiarity with the espionage potential with a stolen device including national security information, or, highly confidential codeword intelligence. In contrast, plaintiff has found few computing devices over 30 years he *couldn’t* compromise. Even before plaintiff worked for Apple, he had considerable experience dealing with users losing devices with confidential information, particularly for the California State University. Moreover, plaintiff had recovered data from lost devices that had failed or were even seized by law enforcement. In one case, plaintiff defeated Apple File Security encryption on a PowerMac G4 and recovered intentionally encrypted data resulting in a serious criminal conviction for the accused employee. If restrictive application identifiers had been invented by the plaintiff back then, it’s uncertain the same methods he used to defeat Apple’s own encryption would have worked, for example.

533. **Claim 10**. The method of claim 8, further comprising transmitting, via a public IP connection to a notification server associated with the notification handler process, an instruction to remove a copy of the record of restricted application identifiers—provided by the recipient device—from storage at the notification server.

534. Herein the true owner is disabling lost “discovery” mode using the cloud server they had previously used to enable the functionality. This results in no need to continue enforcing restricted application identifiers and a return of normal functionality for the device.

535. **Claim 11**. The method of claim 2, wherein the recipient device is in the inactive state when at least a display of the recipient device is dark while the recipient device is running on battery power.

536. Herein this claim means that the ability to use restricted application identifiers (i.e. a blacklist) and execute remote command instructions does not require the display to be active. This makes sense, as otherwise a person would have to push buttons on the device after finding it before any restriction policy or remote command enforcement occurs. It also helps retain resistance while still allowing remote command interaction, which is an important consideration not disclosed by Apple. A device may have low resistance when lost, or, may potentially never be found before depletion. This was an important consideration plaintiff explained previously and stemmed from his previous work on battery signaling across different power states at Apple.

537. **Claim 20**. The computing device of claim 18, wherein the operations further comprise transmitting, via a public IP connection to a notification server associated with the notification handler process, a copy of the record of restricted application identifiers generated by the computing device, such that the notification server can (i) temporarily store, at the notification server, push notifications for applications installed on the computing device that have application identifiers included on the copy of the record of restricted application identifiers, and (ii) opportunistically push the stored notifications along with a notification for an application installed on the computing device that has an unrestricted application identifier.

538. This claim simply discusses the lost device sending a list of its application restrictions to the cloud-based notification server when lost “discovery” mode has been enabled. As already explained in great detail, part of the novelty of plaintiff’s invention is that the cloud-server ensures enforcement of policies on the lost device; largely so a thief has no method of disabling or otherwise overcoming such protection after being enabled by the true owner. It also ensures that the device cannot be restored and overridden by connecting it physically to a

computer device using a physical cabled connection. While restoring an iPhone using iTunes on a computer is what most users might identify as the typical means of repurposing a device, both historically and currently hackers, intelligence services and rogue actors also use physically connected devices to overcome and use "brute force" to defeat the Apple password mechanism. Around the time of this cases initial filing, law enforcement agencies began purchasing expensive devices which circumvent Apple's device passcode policy. While breaking into the device has become easier (despite Apple's rancor in adverts about iPhone security) an application restriction policy enforced by a cloud-server helps ensure the local data is inaccessible when local sabotage has been successful and isn't disclosed in Apple's application. From a user perspective, attempts to launch a Mail or Notes application will still fail and wrongfully "bounce" when launched; as if the application was already open when it's not. This preserves the encryption of the application data hives; while preventing the saboteur from simply copying the application bundle to another similar device and again attempting to access the data. Its unfortunate Apple failed to disclose or explain this; the patent would've been issued much sooner, and, wouldn't be both confusing and incomplete.

539. **Claim 22**. The computing device of claim 20, wherein the operations further comprise transmitting, via a public IP connection to a notification server associated with the notification handler, an instruction to remove a copy of the record of restricted application identifiers—provided by the computing device—from storage at the notification server.

540. Again, lost "discovery" mode's herein been disabled by the true owner, using the cloud-based notification server. The notification server than removes the event loop for remote command instructions, which removes the application blacklist initially received from the device.

541. **Claim 23**. The computing device of claim 16, wherein the computing device is in the inactive state when at least a display of the computing device is dark while the computing device is running on battery power.

542. Herein is another case of ambiguity for the application, simply reinforcing the principle discussed *supra* at 537 applies to inactive states after initial detection and blacklisting.

Count 14 Patent 10,257,709

Bypassing security authentication scheme on a lost device to return the device to the owner

543. *The '709 patent includes the following claims plaintiff invented, specifically 1, 4, 8, 9, 13, 14, 15, 19 and 20 as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and

Exhibit 12.

544. Apple filed for this patent on August 18, 2017—almost 3 years after the 2014 wrongful termination of plaintiff.

545. Fittingly, Apple's misjoinder inventors didn't properly understand plaintiff's copious invention notes and narratives for reliably finding a lost device, and, communicating with an honest finder. This required a subsequent continuation application of this patent from 2014/0199966 originally filed in 2013, and, disclosed previously in earlier causes of action. Much of the reason for this continuation patent was the apparent realization third-party applications could be used by an honest finder to communicate with a privileged contact—telephony and messaging apps by third parties could be used in-place of the default ones provided by Apple.

546. Plaintiff mentioned this important distinction to several people while advocating his invention and explained his usage case of using VOIP service when traveling on Apple business with his iPhone and Mac. The concern was that a lost device could be configured by default to receive telephony communication over a wireless connection; even if the wireless network was available and connected, it needed to be established as a communication mechanism in the software for locating a lost device. Otherwise, an honest finder could attempt to dial a privileged contact when lost "discovery" mode had been enabled and effectively not have the equivalent of dial tone.

The cellular baseband connection on mobile devices is always the default for telephony service, irrespective of the hardware vendor. The software controlling the devices operation (i.e. the operating system or telephony app) must be cognizant of this secondary communication lane; this allows the incoming and outgoing telephony to occur independent of just the cellular baseband, allowing just a wireless internet connection to receive and transmit. The risk plaintiff feared was a lost device being able to connect to a wireless network for VOIP that was bypassed, in favor of the traditional baseband connection all mobile phones use by default. A particular usage case was the iPhone user traveling overseas; who'd switched over to VOIP calling to avoid costly international roaming from their home cellular provider, or, those who had access to Wi-Fi but had unsustainable cellular service in outlying areas. Several national parks in America have had this issue, for example. A strong Wi-Fi connection might be available from a building with a wired connection, but cellular service for some (or all) carriers may be nonexistent or unreliable. Sadly, this reasoning for the need of this continuation patent wasn't explained in the application by Apple; not surprisingly because he was no longer at Apple.

A new misjoinder inventor (not present on other phone-finding patents) is named solely herein; further explaining the confusing narrative and Apple's unnecessary need for continuation. Instead of making this simple explanation, Apple instead discloses the need to switch to text messaging apps for communicating with an honest finder. This is the exact same problem with augmenting the devices use of Wi-Fi instead of the cellular baseband controller, but unexplained. Apple continues to beat a drum discussing change between using telephony and text messaging in the user interface instead.

Understandably, when one copies something given them, they miss important distinctions about the architecture and functionality inherent to the original creator. Since none of the misjoinder inventors actually *invented* anything, their application disclosures were confusing and uncertain. The heart of this patent lies in allowing the mobile device to use a different transport, other than cellular baseband. Most text messaging apps (including Apple's) allow for sending messages using wireless Internet, instead of available cellular connections. Apple's rampant narrative in the application (*in re* switching between telephony or text (SMS) messaging in the user interface) necessarily depends on the device being configured to bypass the baseband connection. Otherwise, nothing works; the device continues to be lost and unreachable. Nothing is novel about switching communication methods and Apple didn't apply for a design patent for the GUI declared. Given design patent applications for GUI and user interface element design at the PTO have grown considerably and allows for overcoming the hurdle of 35 USC § 101, this further exposes the confusion Apple's misjoinder inventors invoked upon its own PC; from not understanding innovation it wholly stole from plaintiff.

547. **Claim 1**. A method comprising: displaying, while a device is in a locked mode in which a plurality of services are unavailable on the device, a selectable user interface (UI) item on the device for enabling a person to operate the device to communicate with a privileged contact while the device is in the locked mode; upon selection of the UI item, displaying a list of privileged contacts while the device is in the locked mode, wherein the list of privileged contacts includes an owner of the device; displaying, upon selection of the privileged contact from the list of privileged contacts, a list of different communication types for initiating a communication with the privileged contact, the displayed list of different communication types being exclusive of communication types for initiating a communication with other privileged contacts; and initiating the communication with the privileged contact selected from the list of privileged contacts, the communication having a communication type selected from the list of different communication

types.

548. This claim discusses the basic function and operability of the plaintiff's phone-finding invention. Operating the device in a manner that an honest finder can have a reasonable chance to reach the true owner is evident repeatedly in plaintiffs' description and invention notes. Plaintiff has a "messaging" section in **Exhibit 8** to specifically deal with this problem; allowing the true owner to (at the very least) display contact information on the device when lost "discovery" mode has been enabled by the true owner. Moreover, plaintiff declares and differentiates between using a default lost contact message and a custom message that's defined by the true owner with additional parameters. Even one not skilled in the art can easily discern plaintiff's invention was copied wholly herein by examining the example UI in **Exhibit 12**. It's obvious that Apple's later UI matches plaintiffs much earlier invention notes. Plaintiff also mentions in a note that, "user record allows storage of device names and contact numbers." This is so that the cloud-server can enforce this message without intervention locally by a thief, who wishes to instead maintain the false image the device wasn't stolen. It also allows the true owner to dynamically change their contact info. A perfect example is the lost phone during a vacation; the addition of the local hotel number may be easier for the true finder to dial— particularly if either the lost phone or honest finders' phone don't dial internationally. Curiously, but not surprisingly, Apple fails to disclose this important consideration and focus simply on the privileged contact statically declared by the true owner previously, before the device is lost. Pushing the telephone number onscreen is useless if the device cannot reach that number as the service provider has configured the device for that region.

549. **Claim 4**. The method of claim 3, wherein the selected privileged contact is the owner of the device.

550. This is patently obvious amongst plaintiff's invention notes and example UI, particularly in **Exhibit 8** and **Exhibit 12**. The example UI in Exhibit 9 for locating and communicating with a lost device also shows the default privileged contact is the true owner. The discussion *in re* law enforcement tracking in **Exhibit 11** even (to a lesser extent) reinforces this concept, especially the sixth bullet point concerning the privileged user being used to then attempt to track the lost device. The true owner thus has no issue verifying their displayed contact info on the lost device for law enforcement if recovered.

551. **Claim 8**. A non-transitory machine readable medium storing a program which when executed by at least one processing unit of a device bypasses a device security protection to

communicate with a privileged contact, the program comprising sets of instructions for: displaying, while the device is in a locked mode in which a plurality of services are unavailable on the device, a selectable user interface (UI) item on the device for enabling a person to operate the device to communicate with the privileged contact while the device is in the locked mode; upon selection of the UI item, displaying a list of privileged contacts while the device is in the locked mode, wherein the list of privileged contacts includes an owner of the device; upon selection of the privileged contact from the list of privileged contacts, determining a communication type of a plurality of different communication types for initiating a communication to the privileged contact based at least in part on a configuration of the device; and initiating the communication with the privileged contact selected from the list of privileged contacts, the communication having the determined communication type.

552. Again, plaintiff demonstrates an example UI in his **Exhibit 12** invention notes which shows this claim in-action. The "Example Lock Screen When Lost" shows a lost iPhone with a dialog box allowing only the operation of the telephone to reach the true owner, using the devices baseband connection to dial 123-456-7890. An option for the true owner to unlock the device is depicted below, however, a plurality of services is unavailable on the device is clearly occurring. Herein, a picture is worth a significant amount of words. Plaintiff also illustrates in **Exhibit 9** how the true owner of one device could theoretically also be defined as a privileged contact on yet another device, illustrating the example of a minors phone listed along with the plaintiffs and his spouse as available devices using plaintiffs credentials; registered with the cloud server; which Apple sometimes also calls a notification server. This shows in reverse action how the plaintiff could be the privileged contact which appears on Junior's iPhone (as depicted in the **Exhibit 12** example UI) when lost "discovery" mode has been enabled.

553. **Claim 9**. The non-transitory machine-readable medium of claim 8, wherein the program further comprises a set of instructions for receiving the selection of the privileged contact from the list of privileged contacts before initiating the communication.

554. As previously interrogated *supra*, the list of privileged contacts is received by the lost device when lost "discovery" mode has been enabled using the cloud-based server. Unlike Apple's unsure application, plaintiff allows for the information to be changed dynamically, provided the proper credentials are used. **Exhibit 10** shows in more detail how a user record is joined with the device's hardware identifier in the "User Record Mapping" block diagram. When plaintiff performed his *reduction to practice* over two years before the misjoinder inventors

named on the phone-finding patents using the same two servers internally called [redacted] and [redacted] ([redacted]) he resolved the media access control (MAC) address with his .Mac user account. This allowed the server to "understand" which device the provided credentials should attempt to access. It has the added protective benefit of eliminating unsolicited traffic from Apple's servers (and flustered users who just lost an expensive device) to devices which don't belong to the true owner. This also ensures accidental lockout doesn't occur from a cloud-server user account (iCloud today) from too many attempts to discover a lost device. It should be noted that none of the misjoinder inventors had access to the two servers called [redacted] and [redacted] which were necessary to develop this feature, and, for its everyday use; particularly as security restrictions forbid anyone without a need to know being involved with production assets handling sensitive user data.

Some misjoinder inventors are not thought to even been employed at Apple when plaintiff invented Find my iPhone. Not one misjoinder employee had a required nondisclosure on file for [redacted] and [redacted] or anything in the .Mac or Mobile Me infrastructure; none of them ever attended cross-functional meetings, their managers weren't disclosed or involved, and, they had no access to use, work-on or otherwise *even know* what assets were necessary. Nobody who worked with plaintiff during his nearly decade tenure at Apple (hundreds of employees) knew any of these misjoinder inventors; any suggestion to the contrary would necessitate investigation by the DOJ, as it would raise serious questions about espionage with Apple infrastructure. Plaintiff was so concerned about tracking customer devices without their knowledge, he refused to demonstrate the feature to others using devices other than his own, despite the zeal of several employees who were very excited about his invention and wanted to see their own device be located on-screen.

While Apple's current adverts feature iPhone privacy at the moment, Apple's counsel state in their demurrers that the misjoinder inventors performed an earlier reduction to practice, which wasn't physically possible, and, would necessitate investigation by the DOJ if it were true. Having members of the public who're not employees (or employees breaking into extremely sensitive production servers) accessing extremely sensitive assets such as [redacted] and [redacted] necessitates espionage investigation and is obviously untrue, however, this reality may become necessary if Apple continues to disregard undisputed facts. Moreover, plaintiff was the only person in his entire division who had access to [redacted] and [redacted], and, it was granted by engineering management specifically so he could help VIPs like Rush Limbaugh, those working with the Office of the CEO including journalists, and, to substantiate reports of emerging user issues (bugs and service disruptions) before they became an impact to most users. Plaintiffs management didn't have access to [redacted] and [redacted] and the misjoinder inventors weren't even known to the engineering team responsible for this infrastructure, and, may not have been employed by Apple in some cases. In this country, a much earlier *reduction to practice* is recognized in patent law, however, Apples repeated oppositions demonstrate they *still don't know how this feature works*; an embarrassment for customers, employees and shareholders. If Apple could substantiate even a portion of its untrue defense, they'll need to contact the DOJ and temporarily shut down iCloud worldwide; a very serious forensic proposition that may require disclosing a privacy breach to all Apple customers. There's otherwise no method to argue that plaintiff did NOT perform a much earlier *reduction to practice* involving assigning a privileged contact to a potentially lost device and using remote commands, etc. To be clear, again, there's no way to assign a privileged contact at Apple without [redacted] and [redacted].

555. **Claim 13**. The non-transitory machine-readable medium of claim 8, wherein the list of privileged contacts comprises a set of automatically generated privileged contacts.

556. As discussed *supra*, the cloud server is capable of automatically generating the list of privileged contacts for the lost device when lost "discovery" mode has been enabled by the true owner, as described in **Exhibit 10**.

557. **Claim 14**. A device comprising: a memory; and at least one processor configured to: display while the device is in a locked mode in which a plurality of services are unavailable on the device, a selectable user interface (UI) item on the device for enabling a person to operate the device to communicate with a privileged contact while the device is in the locked mode; upon selection of the UI item, display a list of privileged contacts while the device is in the locked mode, wherein the list of privileged contacts includes an owner of the device; and initiate a communication with the privileged contact selected from the list of privileged contacts, the communication having a communication type selected from a plurality of different communication types determined based on a configuration of the device.

558. Herein this claim describes the lost device after lost "discovery" mode has been enabled by the true user using a cloud server. The device becomes unusable and communication with the true owner is the only operation either an honest finder or thief alike may utilize. The "Example Lock Screen When Lost" UI in **Exhibit 12** shows this occurring in-action.

559. **Claim 15**. The device of claim 14, wherein the at least one processor is further

configured to receive a selection of the privileged contact from the list of privileged contacts before initiating the communication.

560. This claim is merely for application ambiguity and is reinforcing that a lost device (with at least one processor) may receive the privileged contact from the list of privileged contacts before an honest finder initiates any communication with the true owner.

561. **Claim 19**. The device of claim 14, wherein the configuration of the device does not support cellular communications.

562. As discussed several times *supra*, a lost device may simply use Wi-Fi to connect to the cloud server and doesn't require a cellular baseband connection. An iPod Touch, many iPad tablets and Mac computers all fall into this category. In practice, even a Bluetooth PAN connection to an active switched network could be used, however, it'd need to be configured before it was lost in most cases, as the ability to use a VNC is disabled proactively using application identifiers; meaning the true owner couldn't reach the window server to actuate the new PAN connection if the device had been moved after being declared lost.

563. **Claim 20**. The device of claim 14, wherein the at least one processor is further configured to initiate the communication comprises a set of instructions for using a third-party application to initiate the communication.

564. As discussed several times *supra*, a third-party application may be used to initiate the communication between an honest finder and the true owner of a lost device using plaintiffs' invention. The biggest example is VOIP applications, which are particularly cost effective when traveling abroad in foreign countries; where a separate (and more expensive) calling plan is required by the true owner's cellular service provider. When plaintiff travelled on Apple business internationally, he used a VOIP application, as Apple forbid him from having an AT&T international calling plan. This claim simply allows a VOIP application that's been configured to re-route normal telephony from the cellular baseband to (typically) a Wi-Fi network. While this is the common usage case. It's entirely possible a third-party application could also be used for telephony.

Count 15 Patent 10,447,839

Device Locator Disable Authentication

565. *The '839 patent includes the following claims plaintiff invented; specifically, all twenty claims as listed below*. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**, **Exhibit 5**, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

566. **Claim 1**. A method comprising: receiving, by a mobile device, a request to disable a device locator mode of the mobile device in which an authorized requesting device can receive location information related to the mobile device; upon receiving the request to disable the device locator mode of the mobile device: sending, by the mobile device, a hardware identifier of the mobile device to a server; in response to a determination that the hardware identifier is associated with a user account, receiving, by the mobile device, from the server, a challenge for credentials of the user account; receiving, by the mobile device, credentials through a user interface; sending, by the mobile device, the received credentials to the server; receiving, by the mobile device, from the server, an indication that the received credentials match the credentials of the account associated with the hardware identifier; and in response to receiving the indication, disabling the device locator mode; and following disabling the device locator mode, entering, by the mobile device, an activation operating mode, wherein the mobile device is configured to enable one or more functions in the activation operating mode.

567. Disabling the device locator herein is akin to disabling lost “discovery” mode in plaintiffs original phone finding invention. This is akin to the “end process” described in the “Progress” section of **Exhibit 8**. Moreover, the most common way of disabling the device locator (which assumes the lost devices true owner has regained possession; or at least knowledge from an honest finder the device is safe) can be accomplished using the cloud server, with an example UI present in **Exhibit 9**. Once the device locator has been successfully enabled, the “Find Devices” radio button changes state to then allow cessation of the locator and resulting event loops corresponding with it; including remote command execution.

568. This claim then discusses the method and apparatus used for configuring the device locator; both initially and then when its functionality is desired by the true owner of s lost computing or telephony device. **Exhibit 9** shows an example UI which has been wholly duplicated by Apple; both functionally, and, herein the patent application claims. The user authenticates with the cloud server and then has access to other devices which have been registered using the same credentials. This comprises the device list shown by plaintiff in his much earlier “Example Process UI” already heavily interrogated.

569. **Exhibit 10** shows a block diagram that shows how the hardware identifier that’s unique for a corresponding computing or telephony device is joined with the true owner’s user account, creating a unique user record as depicted. Moreover, plaintiff discloses both IMEI and MAC address reconciliation with the true owner’s user record. **Exhibit 12** alternatively discloses

the cloud server using various communication network types to communicate the information contained in the unique user record essential for solving this longstanding problem—which plaintiff did much earlier than Apple, while Apple claims his sole invention (they wholly copied) was prior art disclosed by Nokia, which couldn't be further from the truth. Apple's patent illustrations continue to use many of the same images from previous applications; all of which are very similar (or identical) copies of plaintiffs dated, much earlier invention notebook entries from August 30, 2008.

570. **Claim 2**. The method of claim 1, comprising: while in the activation operating mode, sending, by the mobile device, to the server, information related to a new account to be associated with the hardware identifier and credentials corresponding to the new account.

571. As defined *supra*, **Exhibit 9** depicts three iPhones which have been associated with the hardware identifier and credentials corresponding to the new account. This is further reinforced by the user record mapping block diagram in plaintiffs **Exhibit 10**.

572. **Claim 3**. The method of claim 2, comprising: while in the activation operating mode, enabling the device locator mode.

573. As also defined *supra*, **Exhibit 9** illustrates this exact moment; when the device locator mode (or lost "discovery" mode) is enabled by the true owner of a lost device, using their user account also registered with the cloud server. In the plaintiff's example, the inventors iPhone has been selected in the device list; with the user about to press the "Find Devices" radio button. This is the same process adopted later by Apple, both with the UI and programmatic function. The device locator mode is thus enabled when the "Find Devices" radio button's depressed. This same button than changes states and reveals messaging akin to stopping the device locator mode and ending lost "discovery" mode.

574. **Claim 4**. The method of claim 3, wherein the device locator mode is enabled using the information related to the new account and the credentials corresponding to the new account.

575. As also defined *supra*, **Exhibit 9** illustrates this exact moment about to occur.

576. **Claim 5**. The method of claim 1, wherein the request to disable the device locator mode is received through a user interface element displayed on the mobile device.

577. In plaintiffs **Exhibit 12**, a method for disabling the device locator by using the correct passcode is clearly depicted in the "Example Lock Screen When Lost" UI. This user interface element is presented on the display of the lost iPhone, as the messaging above the passcode field states that the users iPhone is lost, and, provides a telephone number for an honest

finder to communicate with the true owner.

578. Alternatively, plaintiff also discloses the ability and need (for some users) to disable the correct passcode of the lost device temporarily when they have enabled lost "discovery" mode and are attempting to locate it. Under "Handling Device While Stolen" plaintiff states in the first item that, "we could lock the device and invalidate the true passcode while privileged mode is in-use."

579. **Claim 6**. The method of claim 1, wherein the hardware identifier includes a hash generated based on one or both of a media access control (MAC) address and an international mobile equipment identity (IMEI) of the mobile device.

580. As stated *supra* at **569** for claim 1, **Exhibit 10** shows a block diagram that shows how the hardware identifier that's unique for a corresponding computing or telephony device is joined with the true owner's user account, creating a unique user record as depicted. Moreover, plaintiff discloses both IMEI and MAC address reconciliation with the true owner's user record. To further reinforce the point of a MAC address being used to enjoin a hardware device with a user account in a nonvolatile record, plaintiff represented the cloud server as the "Recovery User MAC" to indicate that the true owner and their lost devices hardware address were enjoined and known by the cloud server, as the sole point of authority.

581. **Claim 7**. The method of claim 1, comprising: before receiving the request to disable the device locator mode, enabling the device locator mode using the credentials of the account associated with the hardware identifier.

582. **Exhibit 9** depicts this in-action, whereas the "Example Process UI" has the plaintiffs iPhone chosen from a list of devices; which also have him designated as a privileged contact. The "Find Devices" radio button impression is shown as connected to the workflow with an arrow, as it enables and executes the device locator when actuated by the true owner. The same credentials for this account are described in the "User Record Mapping" block diagram contained in **Exhibit 10**. One *not* skilled in the art can easily discern that the lost iPhone and known computer share the same common user record.

583. **Claim 8**. A non-transitory computer-readable medium comprising code that, when executed by a processor, causes a device to perform operations comprising: receiving, by a mobile device, a request to disable a device locator mode of the mobile device in which an authorized requesting device can receive location information related to the mobile device;
upon receiving the request to disable the device locator mode of the mobile device: sending, by

the mobile device, a hardware identifier of the mobile device to a server; in response to a determination that the hardware identifier is associated with a user account, receiving, by the mobile device, from the server, a challenge for credentials of the user account; receiving, by the mobile device, credentials through a user interface; sending, by the mobile device, the received credentials to the server; receiving, by the mobile device, from the server, an indication that the received credentials match the credentials of the account associated with the hardware identifier; and in response to receiving the indication, disabling the device locator mode; and following disabling the device locator mode, entering, by the mobile device, an activation operating mode, wherein the mobile device is configured to enable one or more functions in the activation operating mode.

584. This long-winded claim simply states that when the true owner wishes to disable lost "discovery" mode (called device locator mode herein by Apple) the programmatic process is ensuring that the unique user account credentials are validated for the correct lost device, using the hardware identifier to confirm the operation. In other words, the process ends by reversing path exactly as it was enabled.

585. The reasoning isn't clear in Apple's application narrative as to why it might be necessary to perform the same audit to end the process as is used to begin it—not surprising when disclosures don't include the putative inventor. The most obvious concern is to ensure that the feature remains actively locating a lost device; even if another person with physical access to another one of the true owners' computers or devices could accidentally or intentionally disable the locator. Cases involving espionage can mirror themselves technically as being no different than an angry (or intoxicated) acquaintance turning off the device locator after hiding the original device from the true owner. Another consideration relates to a longstanding problem with lost devices plaintiff solved; that a thief will surely attempt to disable any locator to ensure they can repurpose the device to mirror goods not stolen. A consumer who purchases a device with plaintiff's locator invention still running is akin to buying seemingly "new" equipment used that's vacuum sealed, but, contains rocks instead of the actual product inside matching the box. Moreover, this claim prevents a thief (or unscrupulous finder) from simply disabling the locator as soon as its been wrongfully converted.

586. **Claim 9**. The computer-readable medium of claim 8, wherein the operations comprise: while in the activation operating mode, sending, by the mobile device, to the server, information related to a new account to be associated with the hardware identifier and credentials

corresponding to the new account.

587. Herein this claim simply reinforces that when the device locator service is setup for the first time by the devices true owner, that the user name credentials they present to the cloud server are enjoined with the hardware identifier present on the device they choose to add—in which they have also authenticated with the same user account credentials. The unification of these two data types into the user record managed by the cloud server happens initially when adding a new device to that users registered device list. Future successive operations simply audit this information from the cloud server record, until such time as the true owner may decide to remove the device from their registered device list. This allows the now "freed" device to be re-purposed into a new true owner's registered device list with their own user account, for example.

588. **Claim 10**. The computer-readable medium of claim 9, wherein the operations comprise: while in the activation operating mode, enabling the device locator mode.

589. This claim reinforces the concept that the functionality may be used as soon as the user record and hardware identifier have been reconciled into the cloud server record by the true owner, and, that activation operating mode is necessary before any device locator mode (or other remote command execution) may occur. The true owner of the device must declare their chosen device lost and login to the cloud server with the correct credentials, before they can then enable the device locator by utilizing the "Find Devices" radio button in **Exhibit 9**. Fundamentally, the true owner could setup a new device to use the device locator with the cloud server and then utilize the functionality immediately afterwards. A great example herein is setting up a new device and then having it disappear under a sofa cushion a short time later. There's nothing to stop the true owner from immediately using a secondary computer or device to login to the cloud server (effectively iCloud herein) and execute the device locator—which'll show the lost device at the true owners own location.

590. **Claim 11**. The computer-readable medium of claim 10, wherein the device locator mode is enabled using the information related to the new account and the credentials corresponding to the new account.

591. This concept has largely been interrogated *supra* at **589** and elsewhere. This claim was not necessary (like some others amongst the phone-finding patents) and involves duplication; from confusion by the misjoinder "inventors" giving unsure disclosures to PC, as opposed to conventional application ambiguity to reinforce multiple "methods" or "sides" of operation. The extended period of examination time for some of these patents highlights examiner confusion.

592. Moreover, anybody with even a basic understanding of relational databases that's *not* particularly skilled in the art can easily discern that user record commits are available instantaneously; even by multiple handlers or sources making similar requests. If plaintiff had been correctly joined, this claim would've been omitted; with either of the previous two claims simply reinforcing that the device locator could be activated and deactivated immediately after being setup. That's a technically accurate, simple narrative that's easy to understand; not an exercise to create as many claims as possible for something the misjoinder "inventors" were unsure. One of many scary trends emerges from such examples, herein those misjoinder are unsure of how databases have worked for decades and make the assumption to the PTO that plaintiffs invention is (also) novel because the user record can be used after its created, which does explain the serious quality and usability of Apple's software in recent years. One cannot know what they don't understand. Experience has no substitute in software innovation.

593. **Claim 12**. The computer-readable medium of claim 8, wherein the request to disable the device locator mode is received through a user interface element displayed on the mobile device.

594. Herein, this claim reinforces for ambiguity that a request to disable device locator can be received through a user interface element displayed on the lost mobile device. This is useful for situations where a lost device is located by the true owner, especially in close proximity. A lost device discovered under a sofa or table in the true owner's home or office is a perfect example of a usage case whereas disabling lost "discovery" mode is easier without returning to another device and using the cloud server. The earlier example at **179** of a worker discovering their lost device in their vehicle after already enabling the device locator, but, after leaving the office computer area for the near future is a common reason for this functionality. Another example would be a mobile device being used to execute a device locator on another mobile device. Each of these usage cases was carefully considered by plaintiff in 2008 and can be seen by examining "Presenting Data of Device Location" in **Exhibit 12**. Plaintiff depicts using a web browser on a computer, an application on mobile devices, an application on computers and finally, using a custom user interface inside a devices own system preference choice. The latter is how Apple implemented Find my iPhone, with the ability also being always available from the iCloud server using a web browser, and later, Apple developed a discrete application for Find my iPhone; which was then morphed into use with Mac computers and other iOS devices, such as iPad tablets. Yet again, Apple followed plaintiff's invention blueprint precisely.

595. **Claim 13**. The computer-readable medium of claim 8, wherein the hardware identifier includes a hash generated based on one or both of a media access control (MAC) address and an international mobile equipment identity (IMEI) of the mobile device.

596. As stated *supra* at **569** and again at **580** for claim 1, **Exhibit 10** shows a block diagram that shows how the hardware identifier that's unique for a corresponding computing or telephony device is joined with the true owner's user account, creating a unique user record as depicted. Moreover, plaintiff discloses both IMEI and MAC address reconciliation with the true owner's user record. To further reinforce the point of a MAC address being used to enjoin a hardware device with a user account in a nonvolatile record, plaintiff represented the cloud server as the "Recovery User MAC" to indicate that the true owner and their lost devices hardware address were enjoined and known by the cloud server, as the sole point of authority.

597. **Claim 14**. The computer-readable medium of claim 8, wherein the operations comprise: before receiving the request to disable the device locator mode, enabling the device locator mode using the credentials of the account associated with the hardware identifier.

598. This was already interrogated *supra* at **584** and **585** for claim 8. As also discussed *supra* with claim 11 at **592**, this claim (again) provides unnecessary duplication *in re* disabling the device locator, or, lost "discovery" mode.

599. **Claim 15**. A system comprising: a processor; and a non-transitory computer-readable medium comprising code that, when executed by the processor, cause the processor to perform operations comprising: receiving, by a mobile device, a request to disable a device locator mode of the mobile device in which an authorized requesting device can receive location information related to the mobile device; upon receiving the request to disable the device locator mode of the mobile device: sending, by the mobile device, a hardware identifier of the mobile device to a server; in response to a determination that the hardware identifier is associated with a user account, receiving, by the mobile device, from the server, a challenge for credentials of the user account; receiving, by the mobile device, credentials through a user interface;
sending, by the mobile device, the received credentials to the server; receiving, by the mobile device, from the server, an indication that the received credentials match the credentials of the account associated with the hardware identifier; and in response to receiving the indication, disabling the device locator mode; and following disabling the device locator mode, entering, by the mobile device, an activation operating mode, wherein the mobile device is configured to enable one or more functions in the activation operating mode.

600. Herein we encounter another claim with unsure duplication; with information already contained in the first claim. Apple again reinforces that the cloud server user record containing the user account credentials and the corresponding hardware address identifier are audited when the true owner enables or disables device locator mode; or, lost "discovery" mode.

601. **Claim 16**. The system of claim 15, wherein the operations comprise:
while in the activation operating mode, sending, by the mobile device, to the server, information related to a new account to be associated with the hardware identifier and credentials corresponding to the new account.

602. This is simply the ability to register a new device and user with the cloud server; from the device itself, as opposed to using a separate computer or device that's different. Potentially, this could also be used to change or supplement a privileged user contact with an existing one that's recently been established from the device. In **Exhibit 9**, a user could theoretically add or change user entries using a user interface on the device itself, as previously discussed *supra* at **594**.

603. **Claim 17**. The system of claim 16, wherein the operations comprise:
while in the activation operating mode, enabling the device locator mode.

604. Herein we see more claim duplication. The device locator mode can be enabled once the true owner has authenticated with the cloud server; using the same user account they have previously registered their computers and devices with. Additionally, as previously discussed *supra* at **591** and **592**, Apple didn't reinvent the relational database—as soon as the user record has been established in the cloud server's database, the device locator may immediately be enabled and then disabled by the true owner. The UI usability is also no different; as soon as the device hash and password have been provided to the server, the user record exists and can be used. In theory, a user could setup the device locator on a mobile device and then enable it using a computer (or other mobile device in their other hand) before they even finish setting it down.

605. **Claim 18**. The system of claim 17, wherein the device locator mode is enabled using the information related to the new account and the credentials corresponding to the new account.

606. We see yet more duplication herein; enough plaintiff was concerned there were typos in the online PTO application. As discussed *supra*, plaintiffs "Example Process UI" in **Exhibit 9** show this claim in-action. While its clear that two other iPhones have already been configured for device locator support using a unique user record, it's possible that the inventors

iPhone could have just been entered and setup. Since the “Find Devices” radio button is in an active state and accepting user impression, it means the user record was created successfully and that the device locator service is active. In theory, the plaintiff could also have had all three devices already setup previously, having simply logged into the cloud server. Since the devices aren’t ordered alphabetically in the device list (which plaintiff did intentionally in his diagram) it is showing instead the order that the devices were initially registered with the cloud server. In the event that a user had a large plurality of devices (particularly in an institutional or lab environment) they could discern what the “newest” device added was. This is especially helpful if the device names have little human-readable differences between them, such as one number. Anybody who has been an administrator of large numbers of institutional devices is very familiar with such issues identifying unique units. One may not want to stop a computer or telephony device from working normally (as a result of the device locators purposeful design) because they really meant to enable the device locator for another unit.

607. **Claim 19**. The system of claim 15, wherein the request to disable the device locator mode is received through a user interface element displayed on the mobile device.

608. Disabling the device locator mode is possible from both the lost mobile device, and, using another mobile device in which the true owner has the dame user account registered with the cloud server. This has been interrogated *supra* at **594** and **602**.

609. **Claim 20**. The system of claim 15, wherein the hardware identifier includes a hash generated based on one or both of a media access control (MAC) address and an international mobile equipment identity (IMEI) of the mobile device.

610. Apple’s final claim enjoys the most unnecessary duplication again herein. As defined supra, **Exhibit 9** depicts three iPhones which have been associated with the hardware identifier and credentials corresponding to the new account. This is further reinforced by the user record mapping block diagram in plaintiffs **Exhibit 10**. Moreover, this has already been heavily interrogated at **569** for Claim 1, **571** for Claim 2, **580** for Claim 6, **582** for Claim 7, **587** for Claim 9, **592** for Claim 11, **596** for Claim 13, **598** for Claim 14, **600** for Claim 15 and **602** for Claim 16.

Count 16 Patent Application 2018/0337974

Remotely Locating and Commanding a Mobile Device

611. *The ‘974 patent application includes the following claims plaintiff invented, specifically 2, 3, 5-20 and 21 as listed below*. Claim 1 was cancelled by the defendant, pursuant to the PTO application. Evidence is supported by **Exhibit 1**, **Exhibit 2**, **Exhibit 3**, **Exhibit 4**,

Exhibit 5, **Exhibit 6**, **Exhibit 8**, **Exhibit 9**, **Exhibit 10**, **Exhibit 11** and **Exhibit 12**.

612. Plaintiff's counsel propounded a demand letter to Apple including plaintiffs' phone-finding patent misjoinder/nonjoinder on November 2, 2016. Plaintiff filed litigation August 13, 2018. Apple filed for this patent May 17, 2018 and it was published on the PTO website November 22, 2018.

613. Not only was it impossible for plaintiff to have known about this patent directly relating to his phone-finding patents when he filed litigation, Apple intentionally filed for this patent in bad-faith—having distinct evidence and prior knowledge that it concerned plaintiff's novel invention at the heart of this litigation.

614. Surprisingly, Apple previously objected to both this patent application (and 6 other patents) included in this amended complaint on the grounds plaintiff exercised bad faith; despite not revealing them (or any information, whatsoever) period during discovery, and, with full knowledge that there was no way plaintiff could have known about this patent application; since it was not yet published on the PTO website when plaintiff filed his *pro se* complaint. It's instructive and worrisome to note Apple doesn't even treat Samsung as poorly as it's esteemed former employee and plaintiff; who simply seeks correction of ownership to help repair his reputation Apple damaged.

615. The first claim is typically the most important in any patent and was cancelled by Apple. This further reinforces plaintiff's argument that Apple didn't fully understand plaintiff's invention when it wholly copied it. No PC submits an application to the PTO that sees them cancel the very first claim. All the other phone-finding patents heavily utilize the first claim as a foundation for the overall patent; hanging successive claims off of the success of the first claim. Given this application was wrongfully submitted after Apple had already been put on notice of plaintiffs much earlier invention and nonjoinder, it's no surprise the most important claim was than cancelled by Apple. One doesn't know what they cannot understand. The previous phone-finding patents feature unnecessary duplication and an excessive reliance on the same diagrams and illustration sheets; all of which closely resemble plaintiffs 2008 invention notebook enough that anybody *not* skilled in the art can discern the misjoinder "implementors" were working from a facsimile.

616. It's no surprise Apple continues to defy federal law *in re* plaintiff, with this application filed after it knew full well it was plaintiff's sole invention in **Exhibit 13**. Moreover, Apple is continuing the remarkable precedent it set for intentionally exercising bad faith in patent

cases earlier this year in *Apple Inc. v. Qualcomm Inc.*, 3:17-CV-00108 (S.D. Cal 2019) wherein Apple's unethical and wrongful tactics (also being inflicted against the plaintiff herein) were exposed by the *Washington Post*. "Apple's criticism of Qualcomm underpinned more than 80 lawsuits around the world and influenced governments to change laws and regulations in Apple's favor. The documents also raise questions about the methods Apple used to inflict pain on Qualcomm and whether Apple really believed its own arguments to lawmakers, regulators, judges and juries when it tried to change not just its long-standing business agreement with Qualcomm but the very laws and practices that have allowed inventors to profit from their work and investments. The real pain, according to Qualcomm, came when Apple instructed its contract manufacturers, which build its iPhones, computers and other products, to stop paying Qualcomm royalties for patent licensing agreements. Qualcomm argued Apple had also planned this move ahead of time and had even laid out the possible legal scenarios. "Apple will be at risk for infringement, tortious interference and full royalties (plus any interest, penalties, etc.)," Apple wrote in its royalty reduction plan." [19] Apple clearly cannot believe its own arguments presented already in this case, which is evident in not understanding how either the phone-finding or Passbook inventions work in their pleadings.

617. More problematic for Apple is the duty of disclosure to the PTO Apple willfully ignored; both generally with all patents in-question and more specifically, with this application. Apple knew it had intentionally committed nonjoinder of plaintiff for two years before submitting this application. 37 C.F.R. § 1.56 was seemingly codified for just such rare instances: "Each individual associated with the filing and prosecution of a patent application has a duty of candor and good faith in dealing with the Office, which includes a duty to disclose to the Office all information known to that individual to be material to patentability as defined in this section. The duty to disclose information exists with respect to each pending claim until the claim is cancelled or withdrawn from consideration, or the application becomes abandoned." Apple instead exercised bad faith and considerably poor judgment; understanding Apple can afford the very best PC in the world and still intentionally abandoned its duty to disclosure. Moreover, this proves Apple's hodgepodge counsel from external firms retained to defend it clearly do not communicate with Apple's own counsel. This reality continues with the misjoinder engineers and even

[19] Apple said Qualcomm's tech was no good. But in private communications, it was 'the best.' Washington Post. April 19, 2019 https://www.washingtonpost.com/technology/2019/04/19/apple-said-qualcomms-tech-was-no-good-private-communications-it-was-best/

executives named in the original complaint. Thus, Apple has no defense and cannot plead ignorance to its continued bad faith herein.

618. Apple intentionally and willfully submitted this patent nonjoinder of plaintiff. Worse, Apple spent a considerable amount of time reviewing plaintiffs' counsels much earlier demand letter; which was delivered with disclosed electronic read receipts in **Exhibit 23** to the former GC Bruce Sewell. **Exhibit 24** shows forensically the 11 computers inside Apple (see **Exhibit 25**) which reviewed the demand letter (dated November 2, 2016) from **Exhibit 13**. Apple filed this patent application May 17, 2018 with full gross implied malice. Given Senior Director Deborah Rice promised an investigation on November 17, 2016 Apple never performed as promised in **Exhibit 14**, there can be no confusion or misunderstanding.

619. The putative inventor and plaintiff describe a preferred method of practicing the invention not realized in Apple's application. This is one reason why so much uncertainty and unnecessary duplication occurs; the misjoinder implementors are not inventors and couldn't understand the breadth and underlying purposes for the invention. 35 U.S.C. § 112 was disregarded by Apple, whereas they failed to, "include an inventor's subjective beliefs about which mode is best and any aspect of making or using the invention that materially affects the properties of the claimed invention." *Bayer AG v. Schein Pharmaceuticals, Inc.*, 301 F.3d 1306 (Fed. Cir. 2002) at 1320. This is noteworthy because Apple should have included the plaintiff for the declarations and disclosures of this application and willfully refused.

620. Worse, Apple didn't reveal the existence of this application (or several others during discovery. This amended complaint wouldn't even be necessary if Apple followed the law. Similarly, if the plaintiff had even *accidentally* failed to disclose even the most moot minutia, Apple would cause the Court to both compel and sanction him. Apple remains immune to the interests of justice and continues to intentionally prejudice the plaintiff, PTO, other employees (current and former) and patent law itself. If Apple was willing to follow laws wherein it does business, this litigation and much of what it regularly faces would be nonexistent.

621. **Claim 2**. A computer-implemented method of commanding a remote device, the method comprising: authenticating a credential associated with a user account; determining a remote device associated with the user account, wherein the remote device is uniquely identified; presenting one or more remote command enabled for execution at the remote device, wherein at least one command of the one or more remote commands has been enabled for execution by input at the remote device; receiving input selecting a remote command from the one or more remote

commands; and transmitting, to the remote device, an instruction to execute the selected remote command.

622. This claim contains materials from previous phone-finding patents that's already been heavily interrogated. The reasoning for duplication herein is questionable, and, reflects poorly on the fact Apple withdrew the first (and most important) claim. This second claim reads like it should be the first claim but is contained in other previously granted phone-finding patents contained in this complaint.

623. Authenticating a credential associated with a user account was covered in Count 1, Claim 12 at **124** and **125**. Count 4, Claim 1 at **262-264** also previously discuss this, as well as Claim 8 at **277** and **278**. Count 7, Claim 1 at **332-335** also discuss this exact concept already.

624. Determining a remote device associated with the user account, wherein the remote device is uniquely identified has also been duplicated unnecessarily in previous patents, *supra*. Count 4, Claim 1 already covers this at **262-263**, Claim 6 at **271-273**, Claim 8 at **276-278** and Claim 10 at **279-282**, as well as generally at **334**, **340-341**. The duplication continues with Count 15, Claim 1 at **566-569**, Claim 2 at **570-571**, Claim 6 at **57**9-5**80**, Claim 8 at **583-585**, Claim 13 and 14 at **595-598**, Claim 15 and 16 at **598-602**, and, Claim 20 at **609-610**.

625. The ability and usage of one or more remote commands has also been duplicated unnecessarily in previous patents, *supra*. Count 1, Claim 1 covers this already at **110-115**, as well as Claim 18 at **132-133**, and, Claim 19 at **134-135**. Count 6, Claim 1 covers this at **306-307**, as well as Claims 2-4 at **308-313**, Claim 10 at **320-322** and Claim 16 at **327-328**. Count 7, Claim 1 covers this at **332-336**, Claim 9 at **344-345**, **348**, Claim 15 at **353-359**, Claim 17-19 at **360-365**, **371** and Claim 22 at **372-373**.

626. **Claim 3**. The computer-implemented method of claim 2, wherein the transmitting comprises concurrently transmitting multiple commands to the remote device.

627. Multiple commands can be transmitted to the remote device when lost "discovery" mode is active. This has been heavily interrogated *supra* in the 12 different entries at **625**. Moreover, plaintiff shows visual evidence of at least three remote commands which could be transmitted simultaneously in the "Example Process UI" contained in **Exhibit 9**.

628. **Claim 5**. The computer-implemented method of claim 2, further comprising receiving location information from the remote device in response to a locate command.

629. The "Example Process UI" contained in **Exhibit 9** depicts both the radio button for the locate command, and, the "Example UI" contains a map overlay with the plaintiffs lost iPhone

being “found” in Los Gatos; after having been also previously located in Saratoga and Cupertino. Moreover, in **Exhibit 8** the plaintiff explains how “location data [is] sent to Apple” directly after the true owner “[user] activates discovery mode” in the process flow diagram. Under the “Message” section, plaintiff states, “Display phone location after translating GPS location for web display. Show device in map on web app or page.” Plaintiff then further describes how the path of the lost device will be charted whenever it moves.

630. **Claim 6**. The computer-implemented method of claim 2, wherein a command of the one or more remote commands is enabled by default for execution by the remote device.

631. Once lost “discovery” mode has been enabled by the true owner of a lost device using the cloud server, the supported remote command messages are then enabled for use, as they are not enabled otherwise for execution. A user could not utilize any of the other 3 remote command radio buttons depicted in plaintiffs **Exhibit 9** until lost “discovery” mode has been enabled; which is accomplished by pressing the “Find Devices” radio button in the “Example Process UI” example. In the plaintiff’s example, the other 3 remote command radio buttons are disabled, because the “Find Devices” radio button hasn’t yet been pressed. This visual depiction shows this claims default execution by clear example. Alternatively, once the locator has been enabled using the radio button, the plaintiffs example UI depicts three additional remote commands, which may even be transmitted simultaneously.

632. **Claim 7**. The computer-implemented method of claim 2, wherein the remote device is selected from a plurality of remote devices associated with the account.

633. Plaintiffs “Device List” in **Exhibit 9** visually depicts three eligible devices associated with the true inventor’s cloud server account. One may observe that “Darren’s iPhone” has been selected amongst the two others, specifically “Nicole’s iPhone” and “Junior’s iPhone.”

634. **Claim 8**. The computer-implemented method of claim 2, wherein presenting one or more remote commands enabled for execution at the remote device comprises presenting only commands that are enabled for execution by the remote device.

635. **Exhibit 9** shows four total remote command radio buttons in the “Example Process UI” example, which are specifically designed to execute on the lost remote device. One aspect of not showing all possible commands that the cloud server could send to a registered device when the locator is active is processor limitation and battery depletion considerations. Herein again the pitfalls of experience benefit plaintiff and befall Apple.

636. In addition to the device locator not yet being actuated yet in plaintiff’s UI

example, another reason relating to this claim is the problem of not overly exerting the processor with multiple commands unnecessarily, as it can potentially deplete the available resistance quicker than a standard predictive demand curve. Its unknown what apps / processes are active in volatile memory or what resistance threshold a lost device may have when the locator is enabled or lost “discovery” mode. Allowing a true owner (who may be understandably quite anxious) to potentially deplete the available resistance on the lost device prematurely and thus increasing the percentage it’ll become unresponsive before recovery is not an acceptable user experience. The method of allocating the space for remote commands and not having them all active in plaintiffs’ notes were a memory pointer to remind him of the all-important user experience problem. While the iPhone in 2008 could handle multiple commands, there was a very real possibility that some older devices could have difficulty (for various technical reasons) and might support most or some potential commands instead. Plaintiff had experience disabling features for performance on the Mac and knew that not presenting some commands in this interface was the correct programatic method to accomplish this.

637. This was also a consideration for discrete event handlers—so a device which had reached a predetermined resistance parameter could disable intensive command workloads. This problem is exasperated when a cellular connection is being used instead of wireless Internet, which causes command interactions to potentially deplete resistance faster. Since the hardware identifier is known by Apple for all its products, this allows the cloud server to be cognizant of devices whose processors don’t support certain commands, or which have a smaller battery being allowed to perform repetitive commands after a certain threshold. It also helps differentiate devices which only have a cellular connection, as opposed to just wireless Internet, or both. The hardware identifier recorded during manufacturing is matched to the bill of materials and also handles custom configurations. A customer who ordered a cellular circuit for a computer or mobile device which wasn’t included in the default configuration for sale, for example, would still be accounted for programmatically. This also allows Apple to adjust the parameters of the remote command functions with a non-customer facing software update on the cloud server whenever desired. It might be discovered that some remote commands need performance or security changes in production that don’t itself necessitate issuing a standalone update for all supported devices. One distinct possibility was that future devices would have better power management capability and larger batteries, performing more intensive execution potential. In the last decade, mobile phone batteries have increased considerably, with many devices now having

low power performance states as part of their CPU directives. Plaintiff was concerned with the ability to add support for newer devices without compromising any chance of public exposure before announcement by Apple. By enforcing the execution state of remote commands using the cloud server the true owner was already interacting with, it allowed for support to be handled completely on the server. It also allows a method to issue corresponding support for new remote commands on older devices, without needing to qualify a dedicated local update. This was important for iPhone, as firmware updates are only applied going forward and not backwards in version. Having the ability to augment remote command support from the cloud server was thus critically important for supporting older devices with new functionality.

638. **Claim 9**. The computer-implemented method of claim 2, wherein the selected remote command causes the remote device to generate an output.

639. The most obvious example of this claim herein is presenting a map overlay with the dynamic location of the lost device, which is performed as a remotely executed command after the lost devices true owner enabled lost "discovery" mode by pressing the "Find Devices" radio button in the "Example Process UI" example in **Exhibit 9**. The "Example UI" visually depicts location output generated by the true inventors lost iPhone, which has been present in the cities of Los Gatos, Saratoga and Cupertino. Note the images in **Sheet 10** and **Sheet 11** both depict a near identical copy of plaintiffs "Example Process UI" example in **Exhibit 9**. Jake's iPod is no different than Darren's iPhone in this embodiment, or, in others.

640. **Claim 10**. The computer-implemented method of claim 2, wherein the output comprises a message to be presented on the display or a sound to be output from a speaker.

641. The "Example Lock Screen When Lost" in **Exhibit 12** contains an actual user interface example of a message being presented on an iPhone display that is lost. Note that the message presented much later by Apple in **Sheet 12** uses near identical messaging.

642. **Claim 11**. The computer-implemented method of claim 2, wherein the selected remote command causes the remote device to be locked or to be wiped.

643. The "Example Lock Screen When Lost" in **Exhibit 12** shows the iPhone being locked, with an opportunity to enter a 4-digit passcode to unlock it. Apple's much later locked iPhone in **Sheet 13** is identical to plaintiffs. Additionally, in **Exhibit 11**, under "Handling Device While Stolen" the first point says, "we could lock the device and invalidate the passcode while privileged mode is in-use."

644. **Claim 12**. A computing device comprising: an input interface; an output interface;

a wireless network connection; a processor coupled to cause the computer apparatus to: authenticate a credential associated with a user account; determine a remote device associated with a user account; wherein the remote device is uniquely identified; present, via the output interface, one or more remote commands enabled for execution at the remote device, wherein a command of the one or more remote commands has been enabled for execution by input at the remote device; receive, via the input interface, input selecting a remote command from the one or more remote commands; and transmit, via the wireless network connection, an instruction for the remote device to execute the selected remote command.

645. Plaintiffs **Exhibit 10** clearly depicts three computing devices (two iPhone's and a Mac computer) comprising: an input interface; an output interface, and the network connection, the processor configured to cause the computer apparatus to: authenticate a credential associated with a user account; determine a remote device associated with the user account, wherein the remote device is uniquely identified; present, vis the output interface, one or more remote commands enabled for execution at the remote device, wherein a command of the one or more remote commands has been enabled for execution by input at the remote device; receive, vis the input interface, input selecting a remote command from the one or more remote commands; and transmit, vis the wireless network connection, an instruction for the remote device to execute the selected remote command. They are connected to the cloud server using a cellular connection, the Internet, or both. The user record required to both authenticate and provide the other necessary tasks is depicted below using a block diagram. Each of the three devices is uniquely identified to denote their using hardware identifiers in the corresponding user records.

646. **Claim 13**. The computing device of claim 12, wherein the processor is further configured to cause the computing device to transmit, concurrently with the instruction for the remote device to execute the selected remote command, an instruction for the remote device to execute one or more additional remote commands.

647. As stated at **627** for Claim 3 *supra*, multiple commands can be transmitted to the remote device when lost "discovery" mode (or the device locator) is active. This has been heavily interrogated *supra* in the 12 different entries cited at **625**. Moreover, plaintiff shows visual evidence of at least three remote commands which could be transmitted simultaneously in the "Example Process UI" contained in **Exhibit 9**.

648. **Claim 14**. The computing device of claim 13, wherein the selected remote command and the one or more additional remote commands are associated with a predetermined

order of execution.

649. Lost “discovery” mode must be enabled (the device locator) to find the device before one or more additional remote commands can be executed. Until the lost device has been initially located using the “Find Devices” radio button in plaintiffs’ original embodiment (depicted in **Exhibit 9**) the plurality of other remote commands cannot physically execute—even if they were instead transmitted first. In other words, a true owner cannot lock the passcode or remotely wipe the contents of a lost device until it has been located, for example. Once the locator has been successfully actuated, successive remote commands may than be executed.

650. **Claim 15**. The computing device of claim 12, wherein the processor is further configured to cause the computing device to receive location information from the remote device in response to a locate command.

651. Herein we observe unnecessary duplication of Claim 9, as interrogated *supra* at **638-639**.

652. **Claim 16**. The computing device of claim 12, wherein a command of the one or more remote commands is enabled by default for execution by the remote device.

653. The only remote command which is enabled by default for execution by the remote device is the device locator, which is known as plaintiffs lost “discovery” mode—that’s depicted using the “Find Devices” radio button in **Exhibit 9**. While it’s entirely possible the suite of other remote commands which may be executed after the lost device has been located may all contain the same commands that a true owner sees available in the user interface, there’s the possibility of enforced performance directives potentially being enforced for some devices, as characterized at **636**.

654. **Claim 17**. The computing device of claim 12, wherein the remote device is selected from a plurality of remote devices associated with the account.

655. Herein we observe unnecessary duplication of Claim 12, as interrogated *supra* at **644-655**. Plaintiff demonstrates a plurality of remote device associated with the same user account in **Exhibit 9** and also in **Exhibit 10**. Note plaintiff also included a computer in his much older, original invention entries. Apple only used his invention initially for iPhone; before later expanding it to iPad and then Mac computer support much later.

656. **Claim 18**. The computing device of claim 12, wherein presenting one or more remote commands enabled for execution at the remote device comprises presenting only remote commands that are enabled for execution by the remote device.

657. This is largely duplication of Claim 8 and reinforcing (seemingly) that this happens on the remote device, as opposed to only with the cloud server. This has been interrogated *supra* at **635-638**. The mechanism for showing which remote commands are executable on the lost device is programmatic and cognizant of the current event loop restrictions (if any) and the unique hardware properties inherent from the unique identifier; which is stored in the user record. As such, ambiguity is necessarily duplication herein and moot.

658. **Claim 19**. A non-transitory computer-readable medium, storing instructions executable to cause one or more data processing apparatus to: authenticate a credential associated with the user account, wherein the remote device is uniquely identified; present one or more remote command enabled for execution at the remote device, wherein a command of the one or more remote commands has been enabled for execution by input at the remote device; receive input selecting a remote command from the one or more remote commands; and transmit an instruction for the remote device to execute the selected remote command.

659. This is near exact, unnecessary duplication of Claim 12 at **644-655**.

660. **Claim 20**. The non-transitory computer-readable medium of claim 19, wherein the transmitting comprises concurrently transmitting multiple commands to the remote device.

661. This claim simply states the programmatic instructions for remote command instruction execution have been stored in memory.

662. **Claim 21**. The non-transitory computer-readable medium of claim 19, wherein a command of the one or more remote commands is enabled by default for execution by the remote device.

663. As with the previous claim, this claim simply states the programmatic instructions for any default remote command instruction execution have been stored in memory.

ARGUMENT

A. Nonjoinder Claims Are Factually Plausible

664. Without plaintiff's novel innovation the patents in question wouldn't exist—his claims represent *enabling* technology. Without plaintiffs' notes attached to Radar #6262545, there would be no interest, or, methods for Apple to have pursued; notwithstanding the topic of patent applications. Two executives thought it was an excellent idea, with Apple having no proof to explain where its purported innovation otherwise derived. Simultaneous innovation's always a possibility *in re* patents—one needs dated evidence, which Apple cannot provide. *Lindemann Maschinenfabrik GMBH v. American Hoist and Derrick Co.* 730 F.2d 1464 (Fed. Cir. 1984).

Plaintiff provides dated, written evidence which validates the dates necessary for *conception*.

665. Plaintiff presents undisputable evidence that both the remote command functionality and overall method and apparatus for reliably finding a lost mobile device was his novel and original work product; occurring before Apple ever filed for the said patents.

666. Plaintiff presents a plurality of strong evidence demonstrating he devised a method and apparatus to redeem virtual tickets on a mobile device; a full decade before Apple had any interest in development, or, filing a patent. Both the death of Mr. Jobs and lack of an Apple smartphone when plaintiff originally explained his idea to him explains this timeline well. Additionally, plaintiff had been devising a solution for mobile ticketing problems as a direct result of his former job; for an extended period of time.

667. Plaintiffs complaint states valid claims for patent misjoinder, nonjoinder and reputational damage on its own merits; even without being viewed in the customary light most favorable to the plaintiff. *Great Plains Trust Co. v. Morgan Stanley Dean Witter* 313 F.3d 305, 312 (5th Cir. 2002). As such, the Court need not strain to find inferences favorable to the plaintiff, and also, doesn’t need to accept conclusory allegations, unwarranted deductions, or legal conclusions in its determination. *R2 Invs. LDC v. Phillips* 401 F.3d 638, 642 (5th Cir. 2005) (citations omitted). Accordingly, plaintiffs’ arguments and evidence are more than adequate to state a claim upon which relief can be granted. *Mann v. Adams Realty Co*. 556 F.2d 288, 293 (5th Cir. 1977) and *Doe v. Hillsboro Indep. Sch. Dist*. 81 F.3d 1395, 1401 (5th Cir. 1996), reversed on other grounds, 113 F.3d 1412 (5th Cir. 1997) (en banc).

668. Apple’s negligence in never reviewing its IPA’s is inexcusable. Apple’s contractually bound to recognize claims in the Passbook patent plaintiff declared and invented prior to joining Apple. Otherwise, Apple’s breached its own contract; which is an undisputable.

669. While plaintiff has compelling, dated proof of his innovations before the patent applications were filed, obviousness can also assist the Court. Such consideration can serve as relevant “*indicia of obviousness or nonobviousness*” and might be utilized “*to give light to the circumstances surrounding the origin of the subject matter to be patented*.” *Graham v. John Deere Co. of Kansas City* 383 U.S. 1, 17-18 (1966). The Court further stated this may include commercial success, long-felt but unsolved need, and the failure of others. Other factors recognized by the Federal Circuit after *Graham* include whether the prior art teaches away from the invention, whether others have copied the invention, and whether the invention has received industry acclamation. *Ecolochem, Inc. v. Southern California Edison Co*. 227 F.3d 1361 (Fed.

Cir. 2000) with cert. denied, (2001) 532 U.S. 974. It's undisputable that plaintiff himself resolved a long-felt but unsolved need; both for phone-finding and redeeming virtual tickets. Google copied plaintiff's invention; with industry acclamation for Apple deterring mobile device theft being quite high—as well as for Google locating abduction and kidnapping victims.

670. Google subsequently released a "Find my Device" Android OS feature in August 2013—it too wasn't possible without plaintiff's novel invention. There's no question of anticipation by equivalents. *Tate Eng. Inc. v. U.S.* (1973) 477 F.2d 1336, 1342. Apple has "*unclean hands*" from willfully preventing plaintiff from learning about the patents. *Yeda v. ImClone Systems Inc*. 443 F. Supp. 2d 570, 630 (S.D.N.Y. 2006). Rather than solicit multiple patent disclosures from plaintiff, it was decided to wrongfully terminate him instead.

671. Plaintiff never deceived Apple and encouraged patent protection. *Stark v. Advanced Magnetics Inc*. 119 F.3d 1551, 1556 (Fed. Cir. 1997). Plaintiffs burden of proof has clearly been met. *Pannu v. Iolab Corp*. 155 F.3d 1344 (Fed. Cir. 1998). Accordingly, the Court may order the PTO to issue a certificate of correction under § 256. This is an undisputable fact.

B. Plaintiffs' Prior Disclosures Are Enablement

672. Plaintiffs' contributions included the means of implementation to solve the problem of reliably locating a lost smartphone; while giving an honest finder an opportunity to return the device to the true owner. The remote commands and their accompanying results executed on a mobile device processor (in-conjunction with a cloud server) are also novel. Further, plaintiff laid the necessary foundation for the redemption of virtual tickets using a mobile device. Without such disclosures of *enablement*, Apple couldn't have developed such functionality for its products; nor could it have filed for patent protection.

673. 35 U.S.C. § 112(a) requires an inventor to describe it in such terms that one skilled in the art can make and use the claimed invention, as in plaintiffs' disclosures. The standard for determining whether the enablement requirement's been met comes from *Minerals Separation Ltd. v. Hyde* 242 U.S. 261, 270 (1916), which asked if the experimentation needed to practice the invention was undue or unreasonable. Even today, this standard applies, and, plaintiffs' claims support it. One skilled in the art of software engineering could easily implement plaintiffs' novel methods and apparatuses; in fact, this is exactly what happened. Nine different engineers the plaintiff has never met implemented his novel phone-finding invention, calling it their own. Three other different engineers the plaintiff has never met implemented his novel invention for redeeming virtual tickets on a mobile device. If twelve engineers Apple obviously considers

skilled in the art couldn't implement plaintiff's novel invention using his narrative and notes, this litigation would have no purpose for existing; the patents would not have been filed and neither innovative feature would Apple have released. Irrespective of research, it's impossible to find a nonjoinder case where the enablement was so clear and convincing it represents theft. Good artists copy, great artists steal. Apple did both.

674. It's been interpreted to require that the claimed invention be enabled so that any person skilled in the art can make and use the invention without undue experimentation. *In re Wands* 858 F.2d 736 (Fed. Cir. 1988) and *United States v. Telectronics, Inc.* 857 F.2d 778, 785 (Fed. Cir. 1988). ("The test of enablement is whether one reasonably skilled in the art could make or use the invention from the disclosures in the patent coupled with information known in the art without undue experimentation."). Apple proved this by easily implementing plaintiff's invention from his disclosures, and then, filed for patents without him. The method of finding and reporting the location of a lost smartphone, as well as creating and redeeming a digitally issued ticket on a computing device were not in previous art, or, known to Apple before plaintiffs' disclosures.

675. It's beyond doubt to those both skilled and unskilled in the art, that Apple prosecuted the phone-finding patents solely from poor interpretations of his notes. The persistent inconsistencies and unsure statements in the applications stem from the misjoinder inventors lacking a fundamental understanding of how authentication, basic arrays for storing responses of commands, batteries and unique hardware identifiers necessarily operate. Working from five pages of notes shows itself throughout thirteen patent applications; which, among other things, share nearly the same exhibits for drawings and figures, even using the same numerical order.

676. Any part of the specification can support an *enabling* disclosure—even a background discussing (or disparaging) the subject matter disclosed therein. *Callicrate v. Wadsworth Mfg., Inc*. 427 F.3d 1361, 77 USPQ2d 1041 (Fed. Cir. 2005). The test of enablement is not whether any experimentation is necessary, but whether, if experimentation is necessary, was it undue? *In re Angstadt* 537 F.2d 498, 503 (C.C.P.A. 1976). Per § 2164.01(a), *undue experimentation* suggests whether undue experimentation factors are undue. Many factors are to be considered when determining whether sufficient evidence to support a determination that a disclosure does not satisfy the enablement requirement applies, and, whether any necessary experimentation was "undue." These factors include:

(A) The breadth of the claims;

(B) The nature of the invention;

(C) The state of the prior art;

(D) The level of one of ordinary skill;

(E) The level of predictability in the art;

(F) The amount of direction provided by the inventor;

(G) The existence of working examples; and

(H) The quantity of experimentation needed to make or use the invention based on the content of the disclosure. *In re Wands* 858 F.2d 736 (Fed. Cir. 1988).

677. The breadth of the claims is related to a singular objective and doesn't apply. The nature of the invention is software engineering and thus a matter of following the steps outlined in plaintiffs art. Predictability doesn't apply, as otherwise both longstanding problems would have already been solved. Plaintiff had already completed a reduction to practice using Apple's existing cloud server infrastructure, so, this could be heralded as a working example, however, this existence may not have been known by those named in the said patents. The level of experimentation required by the misjoinder implementors was thus nonexistent, or at the very least, extremely minimal. More experimentation was afforded the three co-inventors of the '14 patent; as its evident they were not aware of plaintiffs previous disclosures and incredible diligence in solving the longstanding virtual ticket redemption problem. Herein it could be reasonably assumed that parallel inventorship occurred by two parties who worked for Apple at the same time, but, didn't know or otherwise work together. The difference alone in the quality of the background narrative in the '14 patent shows this. In the former case of the phone-finding patents, those listed on the patent applications are misjoinder; based largely on the fact they simply implemented plaintiffs' previous disclosures and then swore in the patent affidavits that the invention was solely theirs, when in fact, they simply copied plaintiffs work. As explained *supra*, the inconsistencies and instances of assumed magic in the several applications help credibly establish this reality. Moreover, despite his very best efforts, plaintiff was unable to provide direction to anybody *except* the two responsible executives, and, the Apple counsel who decides what Apple will consider for patent protection. While none of the implementors received any direction from the plaintiff, *those most responsible* for the features development and patent applications *did*. The considerable delta herein is difficult to argue and suggests intentional malice by Apple. Herein this example of "direction" sheds much light on the puzzle as to why Apple allowed this matter to go to litigation; after much good faith by plaintiff and his counsel over an extended period of time. Otherwise, Apple's actions (collectively) could have been

interpreted as an admission of guilt with malice, as opposed to an honest mistake. The three individuals (whom plaintiff issued direction) served in the very highest positions at Apple; with absolute authority to make any decision *in re* what Apple develops and patents. There can otherwise be no misunderstanding as to the predicament Apple caused to the plaintiff and putative inventor of two hugely successful innovations; which have brought Apple great acclaim.

678. No prior art existed for the patents in question, as nobody had solved either the problem of reliably finding and retrieving lost smartphone, or, redeeming virtual tickets. The PTO wouldn't have granted said patents otherwise and would have rejected Apple's thirteen applications. Several software engineers of (at least) ordinary skill named on said patents followed the steps in plaintiff's plurality of disclosures to solve both problems at different times. Predictability isn't an issue in this matter; as plaintiff provided enough direction in his disclosures to make solving the problems easily possible by others, else considerable evidence disputing this from those named on the patents would've been included as declarations in Apple's previous objections.

679. No experimentation was necessary after a successful reduction to practice occurred by plaintiff. The determination that "undue experimentation" would have been needed to make and use the claimed invention is not a single, simple factual determination. Rather, it is a conclusion reached by weighing all the above noted factual considerations, as in § 2164.08, 2164.05(a), 2164.05(b), 2164.03, 2164.02 and 2164.06. *In re Wands* 858 F.2d 736 (Fed. Cir. 1988). Apple cannot meet the steep burden required herein; it's clear to even one unskilled in the art that plaintiffs' previous disclosures were simply copied by others. The striking similarity in the example user interfaces and diagrams show that even the third-party professional draftspersons who were given a description based on the plaintiff's example *still* depicted the same things; hence being clear and convincing, especially given in many instances, there's simply no other manner from which to accomplish the said goal of the patents. The brilliance of plaintiff's novel method shouldn't be here overshadowed; there's no other reliable method to accomplish said goals, hence no purpose or reason for undue experimentation to exist.

680. If at least one method is declared for making and using the claimed invention that bears a reasonable correlation to the entire scope of the claim, then the enablement requirement of 35 U.S.C. 112(a) is satisfied. *In re Fisher* 427 F.2d 833, 838 (C.C.P.A. 1970). One skilled in the art should be able to make and use the claimed invention using the disclosures as a guide, which Apple clearly did here. *In re Brandstadter* 484 F.2d 1395, 1406-07 (C.C.P.A. 1973). In essence,

Apple was able to work from photocopies and derive the same results; using many of the exact same user interface examples plaintiff depicts in his disclosures for both the feature itself, and, the corresponding patent applications it filed nonjoinder.

681. Moreover, nothing in plaintiff's lab notebook or previous disclosures to Apple constituted prior art; as no novel solution to solve either the problem of reliably finding lost mobile devices, or, redeeming of virtual tickets previously existed. *Coda Dev. S.R.O. v. Goodyear Tire & Rubber Co.* No. 5:15-CV-1572 (N.D. Ohio 2016), opinion corrected on other grounds on denial of reconsideration…where the patent in question "actually identifies the very concepts [alleged as plaintiff's contribution] as 'prior art').Thus, a strong case exists for plaintiffs' inventions to be constituted as *enabling* technologies; so excellent in quality that the implementors were falsely credited as inventors.

C. <u>Laches Don't Apply</u>

682. § 256 claims have *six-year laches* from discovery of nonjoinder. *Advanced Cardiovascular v. Scimed Life Systems, Inc*. (1993) 988 F.2d at 1161, 1163. Plaintiff didn't discover such patents until November 2014, as Apple's PC would've otherwise contacted him for application disclosures. *Hor v. Chu* (2012) No. 11-1540. Another case and poverty precluded filing even sooner than three years, eleven months. *Vaupel Textilmaschinen KG v. Meccanica Euro Italia SPA* (1991) 944 F.2d 870.

683. Apple won't incur damages any differently than if the action had been brought sooner, despite *laches* not applying. *A.C. Aukerman Co. v. R.L. Chaides Const. Co*., 960 F.2d 1020, 1032-33 (1992). Apple would have, in fact, incurred far less damages if it would have filed a certificate of correction in the nearly two years plaintiff and his counsel were trying to negotiate a settlement in good faith. Since none of the law *in re* patents and § 256 matters would have been applied any differently two years ago, no claim for laches can be made.

684. The Court removed *laches* as a defense in 2017 against claims for patent cases brought within the six-year damages period in 35 U.S.C. 256 in *SCA Hygiene Products Aktiebolag v. First Quality Baby Products, LLC* (2017) 580 No. 15-927, slip opinion. Even if the plaintiffs time period hadn't been met, *laches* cannot apply in this matter.

D. <u>Conception & Patent Inventorship</u>

685. A coinventor need not contribute to every claim of a patent; contribution to one claim's enough. "The contributor of any disclosed means of a means-plus-function claim element is a joint inventor as to that claim, unless one asserting sole inventorship can show that the

contribution of that means was simply a reduction to practice of the sole inventor's broader concept." *Ethicon Inc. v. United States Surgical Corp.*135 F.3d 1456, 1460-63 (Fed. Cir. 1998). The electronics technician in *Ethicon*, who contributed to one of the two alternative structures to define "the means for detaining" in a claim limitation was held to be a joint inventor. In *Tucker v. Naito* 188 USPQ 260, 263 (1975) it was found inventors need not "personally construct and test their invention." Further, "it is not essential for the inventor to be personally involved in carrying out process steps…where implementation of those steps does not require the exercise of inventive skill." *In re DeBaun* 687 F.2d 459, 462 (C.C.P.A. 1982).

686. Herein the plaintiff did test his own phone-finding invention before revealing it to other Apple employees. As such, plaintiff contributed to a plurality of the claims in all sixteen patents. Even in the case of the '14 patent, the only claims plaintiff cannot attach some previous conception to involve the proximity fence. The proximity fence exists only from smartphones, which didn't exist a decade earlier; when plaintiff originally devised the solution for the problem.

687. As a general matter, patents must list all of the true inventors. *Trovan Ltd. v. Sokymat SA, Irori* 299 F.3d 1292, 1301 (Fed. Cir. 2002). "Conception is the touchstone of invention, and it requires a definite and permanent idea of an operative invention, including every feature of the subject matter sought to be patented." *In re VerHoef* 888 F.3d 1362, 1365 (Fed. Cir. 2018) (internal quotation and citation omitted). A definite and permanent idea, in turn, exists "when the inventor has a specific, settled idea, a particular solution to the problem at hand, not just a general goal or research plan." *Id.* (citation omitted).

688. Herein it's been well established plaintiff had a specific, settled idea for both the longstanding phone-finding problem, and, the problem of redeeming virtual tickets with a mobile device. A particular solution existed enough that Apple filed for thirteen patents for phone-finding, with only three for the unique solution of redeeming virtual tickets on a mobile device. Curiously, many Apple employees (potentially several hundred, or more) saw plaintiffs' phone-finding notes, whereas only Mr. Jobs saw plaintiffs more extensive notes *in re* virtual ticket redemption. One couldn't help wondering if more patents would have been filed by Apple; if the plaintiff had been properly joined for the application disclosures already at-hand.

689. In order for an invention to have co-inventors, they "need not 'physically work together or at the same time,' 'make the same type or amount of contribution,' or 'make a contribution to the subject matter of every claim of the patent.'" *Vapor Point LLC v. Moorhead* 832 F.3d 1343, 1349 (Fed. Cir. 2016), quoting 35 U.S.C. § 116. Rather, a joint inventor must:

> (1) contribute in some significant manner to the conception or reduction to practice of the invention, (2) make a contribution to the claimed invention that is not insignificant in quality, when that contribution is measured against the dimension of the full invention, and (3) do more than merely explain to the real inventors' well-known concepts and/or the current state of the art. *In re VerHoef* 888 F.3d 1362, 1365 (Fed. Cir. 2018) (quoting *Pannu*).

690. Herein plaintiff contributed in the most significant manner possible to both the *conception* and *reduction to practice* for the phone-finding patents, enough the misjoinder inventors can no more claim that they were explaining the plaintiff's invention to patent counsel. Anything Apple could argue as being added to his inventions are insignificant in quality, especially with respect to the phone-finding patents, and, the dimension of the full invention.

691. Finally, the issuance of a patent "creates a presumption that the named inventors are the true and only inventors." *Ethicon*, 135 F.3d at 1460. But "a person who alleges that he is a co-inventor of the invention claimed in an issued patent who was not listed as an inventor on the patent may bring a cause of action to correct inventorship in a District Court under 35 U.S.C. § 256." *Vapor Point LLC*, 832 F.3d at 1348 (quoting *Eli Lilly & Co.* 376 F.3d 1352, 1357 (Fed. Cir. 2004)). To overcome a presumption of correctness, "the alleged co-inventor or co-inventors must prove their contribution to the conception of the claims by clear and convincing evidence." *Ethicon*, 135 F.3d at 1461.

692. Clear and convincing evidence cannot elsewhere be derived that's any more helpful for identifying a previous disclosure later being copied by others as their own invention. Plaintiff has dated, written communication from two vice presidents and the senior patent counsel at Apple, with respect to the phone-finding patents. Having evidence with written responses and positive remarks, followed by very strong, dated lab notebook entries pairs well with the narrative provided in the introduction; especially concerning Mr. Jobs and the '14 patent. The seminal question the overwhelming evidence plaintiff demonstrates asks is why Apple still chose to develop and patent "Find my iPhone" if his very first disclosures were not adequate or compelling? Acknowledgment from both the principal vice presidents (later involved in the features development) that the plaintiff had a very good idea make Apple's overall defense seem questionable, both in scope and purpose. In other terms, it's harder to identify a bigger contradiction in any § 256 case; irrespective of the industry, or, product at-hand.

693. The legal definition of an "inventor" has been interpreted by the courts to be

dependent on the *conception* of the patented idea. Any individual who contributes to the conception of patented ideas should be listed as a patent inventor. What qualifies as *conception* has been specifically outlined by the COA: *Conception* is the touchstone of inventorship, the completion of the mental part of invention. It is "the formation in the mind of the inventor, of a definite and permanent idea of the complete and operative invention, as it is hereafter to be applied in practice." *Conception* is complete only when the idea is so clearly defined in the inventor's mind that only ordinary skill would be necessary to reduce the invention to practice, without extensive research or experimentation. Because it is a mental act, courts require corroborating evidence of a contemporaneous disclosure that would enable one skilled in the art to make the invention. Thus, the test for *conception* is whether the inventor had an idea that was definite and permanent enough that one skilled in the art could understand the invention; the inventor must prove his conception by corroborating evidence, preferably by showing a contemporaneous disclosure. "An idea is definite and permanent when the inventor has a specific, settled idea, a particular solution to the problem at hand, not just a general goal or research plan he hopes to pursue. The *conception* analysis necessarily turns on the inventor's ability to describe his invention with particularity. Until he can do so, he cannot prove possession of the complete mental picture of the invention. These rules ensure that patent rights attach only when an idea is so far developed that the inventor can point to a definite, particular invention." *Burroughs Wellcome Co. v. Barr Laboratories, Inc*. 40 F.3d 1223, 1227-28 (Fed. Cir. 1994) and *Bard Peripheral Vascular, Inc. v. W.L. Gore & Assocs., Inc*. 79 F.3d 1572 (Fed. Cir. 1996).

694. Plaintiff passes the test for *conception* with both his inventions by a wide threshold given his corroborating evidence—proving he made a significant contribution to both patented inventions. *Coleman v. Dines* 754 F.2d 353, 224 (Fed. Cir. 1985) and *General Electric Co. v. Wilkins* 750 F.3d 1324, 1330 (Fed. Cir. 2014). Examples of successful corroborating evidence include, but aren't limited to charts, drawings, lab notebooks, graphs, communications, and invention disclosure forms. Plaintiff herein includes a plurality of charts, drawings, his dated notebook entries, diagrams, dated email communications, and, his invention disclosure form with Apple, or IPA. Plaintiffs significant contributions to both inventions can thus be measured with confidence from the strong corroborating evidence. Plaintiff reduced to practice before Apple.

695. The plaintiff must pass the "*rule of reason*" test. *Mahurkar v. C.R. Bard* 79 F.3d 1572 (Fed. Cir. 1996), *Price v. Symsek* 988 F.2d 1187, 1195 (Fed. Cir. 1993). The rule of reason test is clearly passed here by the plaintiff; as his significantly earlier disclosures, narratives and

reduction to practice have demonstrated. In securing *conception*, plaintiff had to work through the experimental problems associated with the actions and determinations both an honest finder and thief simultaneously might take when finding a lost computing device. The plaintiff's strong evidence also helps to pass overwhelmingly the "*rule of reason*" test.

696. With the Passbook patent, plaintiff had to similarly resolve the experimental problems inherent with both managing the several technical needs and requirements for independent ticket sales online, but more importantly, the problems associated with redeeming such tickets with autonomous authority; which could reduce labor constraints and speed up the duration needed to seat a waiting audience for an event. Having a regional university with extensive free and paid performing arts and music performances; juxtaposed with sporting events and third-party events such as NCAA baseball tournaments and festivals was the ideal lab from which to observe and study the inherent problems associated with virtual ticket redemption. Even a stubborn protagonist must concede plaintiff passes the "*rule of reason*" test merely from working in such an ideal environment for developing a reduction to practice leading to actual use; and not simply for personal achievement. It's important to reinforce plaintiff was working in a role to actively support the electronic needs of the performing arts and sports ticketing operations already in-use. In this sense, plaintiff's innovation's akin to developing a new hand tool to fix a longstanding carpentry issue he encountered from regularly building intricate woodworks. Such innovation in the focused area of ticketing is no different than inventing a hammer or screwdriver.

697. Herein plaintiff is no different than an engineer (Mr. Sturman) who worked on residual magnetism in the 1960s and secured an '898 patent in 1973 for related technology. He then worked for Cummins Engine Co. on fuel injector valves, where he developed one using residual magnetic latching. While under a more lucrative IP agreement than plaintiff had at Apple, Mr. Sturman had contracted with Caterpillar and then sketched an idea for an integrated spool valve employing residual magnetic latching. Caterpillar rejected the idea; however, Caterpillar engineers *did* recognize that Sturman's integrated spool valve design had "tremendous potential." Contemporaneous memoranda indicated Caterpillar began instead exploring the idea of using an integrated spool valve for a hydraulically actuated injector. Unlike Apple, Caterpillar requested Mr. Sturman be included in one of two patents their counsel filed with the PTO, but, a breakdown over royalties caused him to refuse signing the disclosure. One application was abandoned (since it correctly had no joint inventors) while the other was granted as a '901 patent. The appeals court reversed summary judgment for Caterpillar (among other things) and affirmed

Mr. Sturman as the sole inventor of two patents. *Caterpillar Inc. v. Sturman Ind.* 387 F.3d 1358 (Fed. Cir. 2004). Plaintiff was already a subject matter expert in battery and networking technology before inventing the phone-finding patents; having been awarded a battery patent by Apple. He similarly and solely developed a novel solution for virtual ticket redemption a decade before Apple in his previous employment with the California State University; which, he declared in his IPA—not unlike Mr. Sturman's joint development agreement with integrated spool valves. While Apple executives had positive feedback *in re* the phone finding patents as a feature, the CEO had taken the additional step of brainstorming the virtual ticket redemption problem with plaintiff. Plaintiff never heard from any stakeholders again, with his only discovering he was nonjoinder on sixteen patents after being wrongfully terminated, and, his counsel recommending research. At least Caterpillar tried to include Mr. Sturman on one patent. Plaintiff didn't expect to derive royalties from said patents as Mr. Sturman's situation warranted; he simply wanted a correction of ownership; which Apple promised his counsel to investigate and was then ignored for two years.

698. Nonjoinder reputational damage alone satisfies constitutional standing. *Faryniarz v. Ramirez* No. 3:13-CV-01064-CSH (D. Conn 2015). The consent of others named on said patents also isn't necessary for a correction to be filed by either Apple, or, the Court. *Iowa State Univ. Research Foundation v. Honeywell Inc.* 444 F.2d 406 (4th Cir. 1971). Again, Apple could have issued a certificate of correction without the consent of those already named, and, independent of the District Court. This would have allowed plaintiff an opportunity to rehabilitate his excellent reputation which Apple destroyed, but they demurred. An impartial observer not skilled in the art could reasonably deduce there's very little (in general) Apple doesn't demur.

E. <u>Reputational Injury Sufficient Alone for §256 Correction</u>

699. A plaintiff not named as a joint inventor on several patents has standing to maintain an action to correct inventorship under § 256, despite the fact they previously assigned all rights in the patented inventions to a former employer. *Shukh v. Seagate Technology, LLC* 803 F. 3d 659 (Fed. Cir. 2015). This decision recognizes that an omitted co-inventor has an enforceable interest in correcting inventorship based on evidence that such correction would enhance the inventor's reputation. "Being considered an inventor of important subject matter is a mark of success in one's field, comparable to being an author of an important scientific paper." at **1359**. We reasoned that "pecuniary consequences may well flow from being designated as an inventor." *Id*. This is particularly true when the claimed inventor is employed or seeks to be

employed in the field of his or her claimed invention. If the claimed inventor can show that being named as an inventor on a patent would affect his employment, the alleged reputational injury likely has an economic component sufficient to demonstrate *Article III* standing.

700. Given the facts here, plaintiffs reputation will be considerably rehabilitated by a correction of ownership for sixteen patents—for two technologies he lawfully invented; which've generated industry acclaim, notoriety and profit for Apple. Overly negative public scrutiny of plaintiff in so many news stories *in re* this case is a fine example of additional reputational damage. Such a correction would mitigate this.

701. Plaintiff will realize better employment prospects in being named an inventor of said patents, after being wrongfully terminated by defendant. Such nonjoinder patents would've caused plaintiff to be awarded Apple's most prestigious award, the *Apple Innovators Award*. This award requires 5 awarded patents and would have increased plaintiff's income and stature considerably. Plaintiff was, "*a stellar and highly valued employee*" like the employee at a partner company Apple had wrongfully fired in *Popescu v. Apple Inc.* H040508 Cal.App.4th (2016).

702. Factors bearing on plaintiffs' credibility and whether his testimony has been adequately corroborated are proven beyond a reasonable doubt. *In re Reuter* at **1021** and *supra*. (1) Delay between the event and the trial have been solely the result of Apple ignoring its written promise on November 17, 2016 by a Director to investigate plaintiffs' claims for two years in **Exhibit 14**; with plaintiff not being aware of his nonjoinder previous to Apple wrongfully terminating him—which avoided issuing him the *Apple Innovators Award*. (2) Interest of corroborating witnesses will be established at trial. (3) Contradiction or impeachment certainly applies; else Apple would have issued a certificate of correction upon promptly learning of plaintiffs nonjoinder and sizable evidence in 2016. Herein Apple has and continues to argue against undisputed fact; contrary to the plurality of written evidence to the contrary. Moreover, those unskilled in law or technology would conclude ample evidence existed for Apple to return plaintiff to full-time employment; as he obviously performed a much-needed work product by the high standards demanded by the CEO and co-founder. Mr. Jobs would have promptly removed plaintiff from his employ if he'd demonstrated anything but superior conduct, and, a high quality of measurable work with exceeding attention to detail. (4) Corroboration exists in the form of a plurality of dated, written evidence; which includes lab notebooks displaying plaintiff's conception and inventorship—in one case predating Apple's patent application by a decade. (5) The corroborating witnesses' familiarity with details of alleged prior structure doesn't apply, as

the Court explained Apple has no burden to provide evidence from the misjoinder inventors. See **ECF No. 39**. Nonetheless, a compelling narrative exists showing that one (or more) employees of Apple photocopied plaintiffs' notes contained in the exhibits without his knowledge; allowing the misjoinder inventors to have anything to claim in the patent disclosures. (6) Improbability of prior use considering state of the art has been Apple's only defense thus far; arguing plaintiff's novel inventions are simply prior art—when one unskilled in the art can compare his notes with the patent applications and make the independent assertion the work product's mostly identical. Other than the case of the Passbook patent (which contains GPS proximity fence claims other Apple employee's unknown to plaintiff invented) one may argue Apple's patents are simply a facsimile of plaintiff's work product. (7) The impact of both the inventions on the industry is sizable. The longstanding problem of reliably retrieving a lost mobile device from either an honest finder or thief has been so well received by customers its spawned efforts to help law enforcement stop abduction and kidnapping, as well as prosecute those who steal mobile devices from the true owner.

703. The Find my iPhone app alone was named the #6 best iPhone application of all-time, receiving a maximum score for cultural impact. [20] It's so loved by Apple customers that thousands of donated iPhones regularly become useless every month; because well-meaning donors forgot to turn off "Find my iPhone" first. [21]

704. Redeeming virtual digital tickets has fundamentally and permanently changed the event ticketing industry. The resulting time saved from efficiency offset alone is too immense to calculate, with daily airport boarding lines around the world seeing constant usage of plaintiffs' invention; as well as being used for the most popular concerting and sporting events by the market leader in Ticketmaster. A sizable revenue stream which Mr. Jobs and plaintiff devised for selling media and ticketing online was abandoned by Apple. One could easily forecast lost ticketing revenue in excess of one billion dollars per year. The global market for live music ticketing alone (not counting sports) is predicted by grow by an average of 7% annually until 2021, with an estimated value of US$24.55 billion in 2021. Sporting ticketing is expected to raise from $49.26 billion to $62.31 billion in 2021, with movies increasing from $45.71 billion to

[20] The 100 Best iPhone Apps of All Time
https://mashable.com/2015/12/08/100-best-iphone-apps/

[21] Thousands of usable iPhones trashed by one Colorado firm because well-meaning donors forgot to turn off "Find my iPhone"
https://coloradosun.com/2019/04/17/recycled-iphones-trashed-find-my-iphone/

$60.68 billion in 2021. [22] Even if Apple took a very small commission for processing ticket transaction for service providers, as his invention also allows, the revenue potential is too enormous to ignore. One percent of the 2017 digital ticketing market of $113.44 billion is $1.13 billion and represents an example of lost revenue from Apple's negligence. The relationship between witness and alleged prior user has been sufficiently identified as employer-employee.

705. One unskilled in the art may reasonably conclude Apple followed the *Shukh* playbook; deciding that wrongfully terminating him for the false reason of "poor communication" was better than following the Constitution and awarding him the *Apple Innovators Award*, 15 patents and 1 pending patent application due. Shukh was only wrongfully omitted from the inventorship of 6 patents and 4 pending patent applications. The reputational damage plaintiff suffered herein exceeds that of Shukh—understanding they both experienced negative economic consequences, harm to their reputations as notable inventors, and, a false reputation for employment conduct he did not engage in.

[22] Concert ticket market to top $24BN by 2021
https://www.iq-mag.net/2017/02/concert-ticket-market-top-24bn-2021-technavio/#.XMOA4y_My3A

PRAYER FOR RELIEF

WHEREFORE, Darren Eastman prays for judgment and relief as follows:

A. An entry of judgment holding the defendant liable of patent nonjoinder;

B. A PTO certificate of ownership correction writ for all sixteen patents named;

C. An order to the defendant to issue corresponding patent award plaques, the *Apple Innovators Award*, and, corresponding IP cash awards due current Apple employees;

D. An award to plaintiff for *actual* and *punitive* damages;

E. A finding that this case is "exceptional" and an award to Eastman of his costs and previous attorneys' fees, as provided by 35 U.S.C. § 285;

F. Such further and other relief as the Court may deem proper and just.

Respectfully submitted,

Dated: April 29, 2020

By: /S/ Darren Eastman

DARREN EASTMAN
Pro Se

DEMAND FOR JURY TRIAL

Darren Eastman demands a jury trial on all issues so triable, pursuant to Rule 38 of the Federal Rules of Civil Procedure.

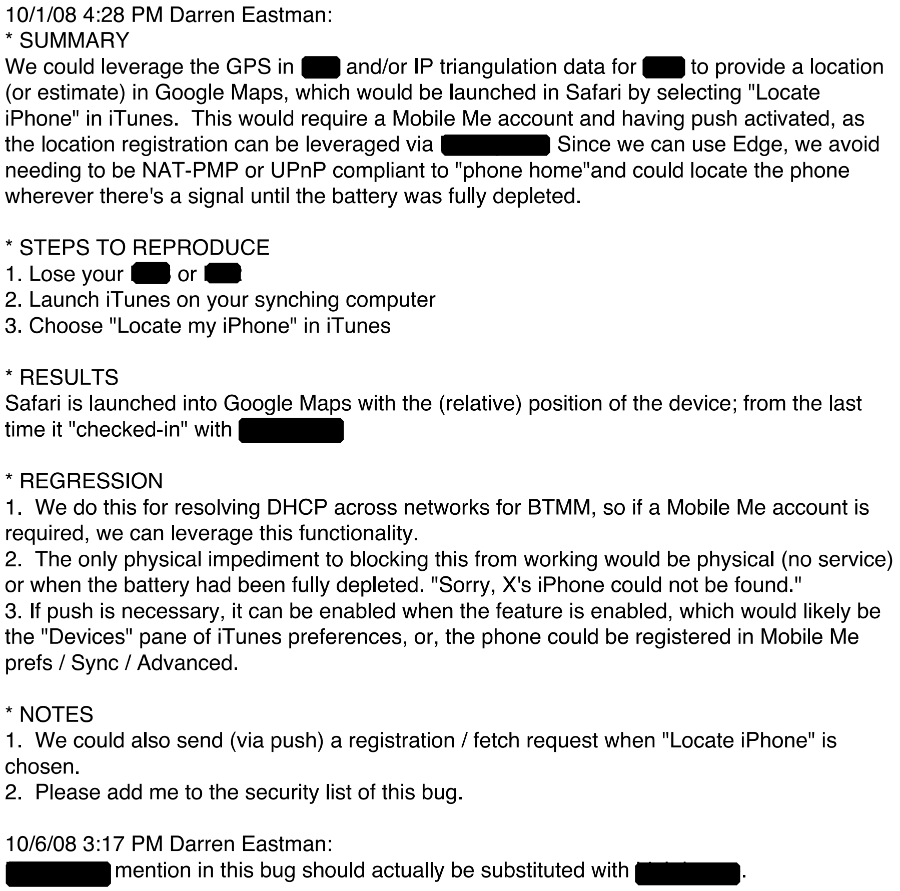
Respectfully submitted,

Dated: April 29, 2020

By: /S/ Darren Eastman

DARREN EASTMAN
Pro Se

EXHIBIT 1



10/1/08 4:28 PM Darren Eastman:
* SUMMARY
We could leverage the GPS in [redacted] and/or IP triangulation data for [redacted] to provide a location (or estimate) in Google Maps, which would be launched in Safari by selecting "Locate iPhone" in iTunes. This would require a Mobile Me account and having push activated, as the location registration can be leveraged via [redacted] Since we can use Edge, we avoid needing to be NAT-PMP or UPnP compliant to "phone home"and could locate the phone wherever there's a signal until the battery was fully depleted.

* STEPS TO REPRODUCE
1. Lose your [redacted] or [redacted]
2. Launch iTunes on your synching computer
3. Choose "Locate my iPhone" in iTunes

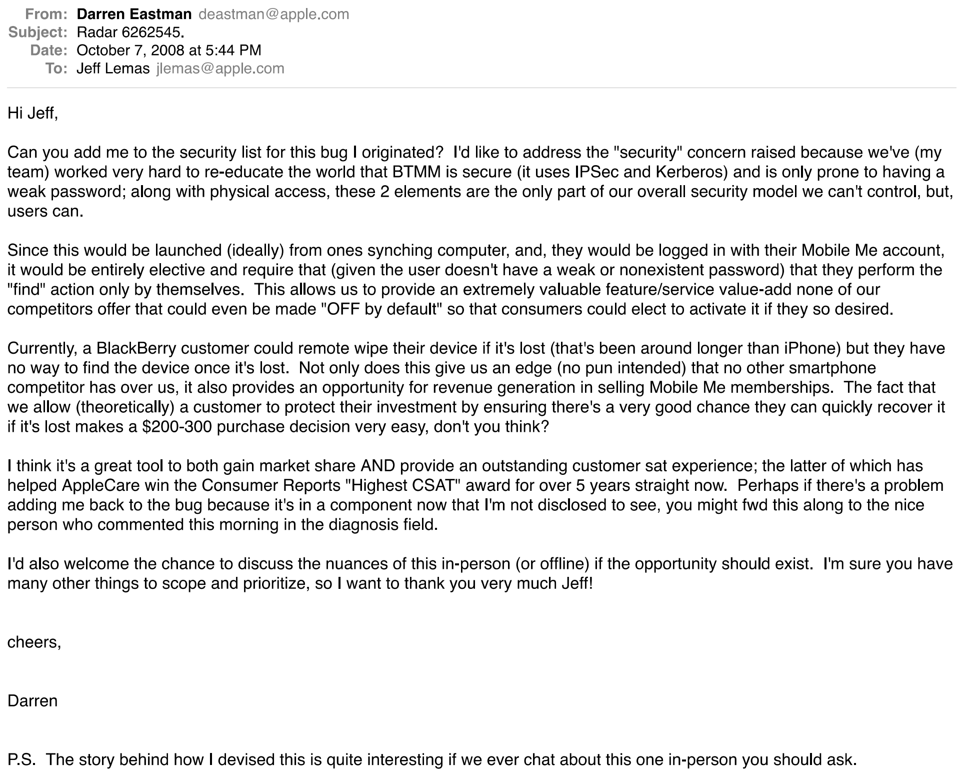
* RESULTS
Safari is launched into Google Maps with the (relative) position of the device; from the last time it "checked-in" with [redacted]

* REGRESSION
1. We do this for resolving DHCP across networks for BTMM, so if a Mobile Me account is required, we can leverage this functionality.
2. The only physical impediment to blocking this from working would be physical (no service) or when the battery had been fully depleted. "Sorry, X's iPhone could not be found."
3. If push is necessary, it can be enabled when the feature is enabled, which would likely be the "Devices" pane of iTunes preferences, or, the phone could be registered in Mobile Me prefs / Sync / Advanced.

* NOTES
1. We could also send (via push) a registration / fetch request when "Locate iPhone" is chosen.
2. Please add me to the security list of this bug.

10/6/08 3:17 PM Darren Eastman:
[redacted] mention in this bug should actually be substituted with [redacted].

EXHIBIT 2



From: **Darren Eastman** deastman@apple.com
Subject: Radar 6262545.
Date: October 7, 2008 at 5:44 PM
To: Jeff Lemas jlemas@apple.com

Hi Jeff,

Can you add me to the security list for this bug I originated? I'd like to address the "security" concern raised because we've (my team) worked very hard to re-educate the world that BTMM is secure (it uses IPSec and Kerberos) and is only prone to having a weak password; along with physical access, these 2 elements are the only part of our overall security model we can't control, but, users can.

Since this would be launched (ideally) from ones synching computer, and, they would be logged in with their Mobile Me account, it would be entirely elective and require that (given the user doesn't have a weak or nonexistent password) that they perform the "find" action only by themselves. This allows us to provide an extremely valuable feature/service value-add none of our competitors offer that could even be made "OFF by default" so that consumers could elect to activate it if they so desired.

Currently, a BlackBerry customer could remote wipe their device if it's lost (that's been around longer than iPhone) but they have no way to find the device once it's lost. Not only does this give us an edge (no pun intended) that no other smartphone competitor has over us, it also provides an opportunity for revenue generation in selling Mobile Me memberships. The fact that we allow (theoretically) a customer to protect their investment by ensuring there's a very good chance they can quickly recover it if it's lost makes a $200-300 purchase decision very easy, don't you think?

I think it's a great tool to both gain market share AND provide an outstanding customer sat experience; the latter of which has helped AppleCare win the Consumer Reports "Highest CSAT" award for over 5 years straight now. Perhaps if there's a problem adding me back to the bug because it's in a component now that I'm not disclosed to see, you might fwd this along to the nice person who commented this morning in the diagnosis field.

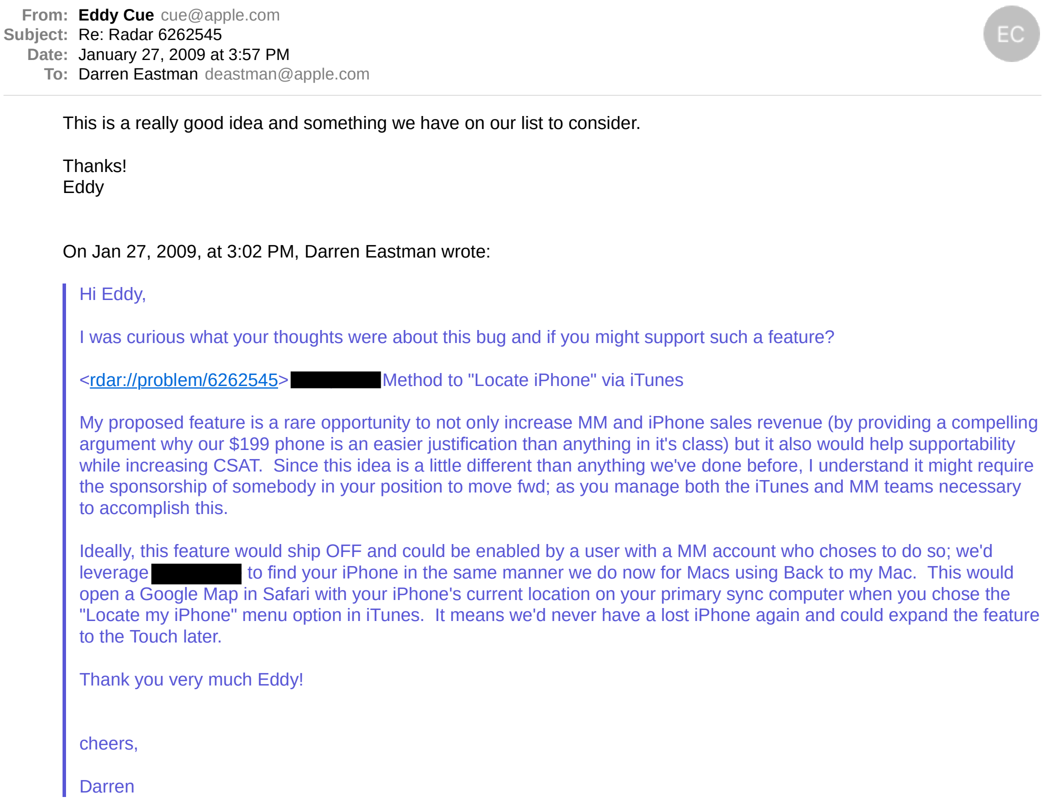
I'd also welcome the chance to discuss the nuances of this in-person (or offline) if the opportunity should exist. I'm sure you have many other things to scope and prioritize, so I want to thank you very much Jeff!

cheers,

Darren

P.S. The story behind how I devised this is quite interesting if we ever chat about this one in-person you should ask.

EXHIBIT 3



From: **Eddy Cue** cue@apple.com
Subject: Re: Radar 6262545
Date: January 27, 2009 at 3:57 PM
To: Darren Eastman deastman@apple.com

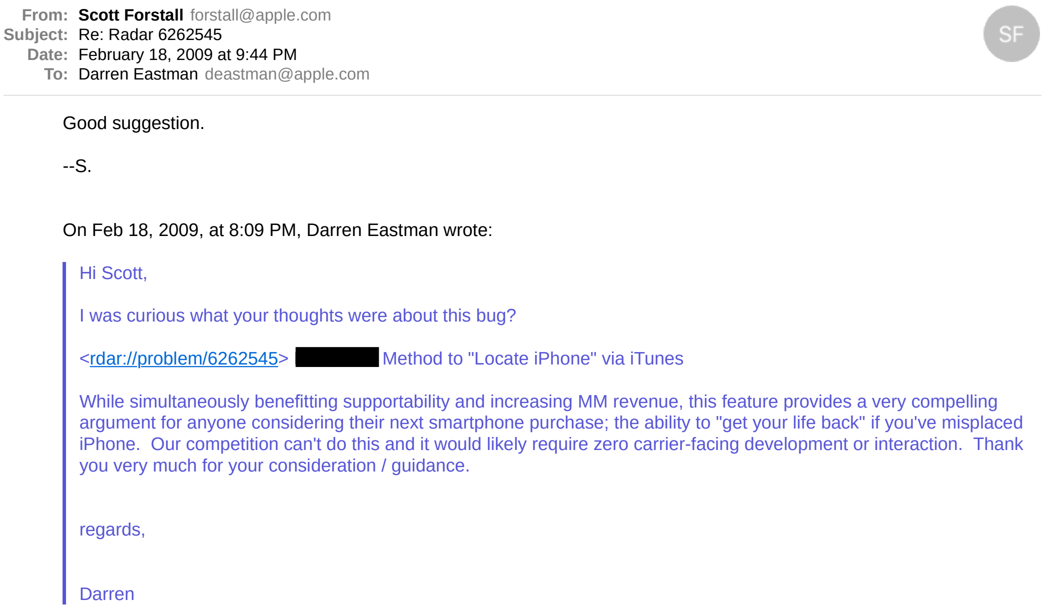
This is a really good idea and something we have on our list to consider.

Thanks!
Eddy

On Jan 27, 2009, at 3:02 PM, Darren Eastman wrote:

> Hi Eddy,
>
> I was curious what your thoughts were about this bug and if you might support such a feature?
>
> <rdar://problem/6262545>[redacted]Method to "Locate iPhone" via iTunes
>
> My proposed feature is a rare opportunity to not only increase MM and iPhone sales revenue (by providing a compelling argument why our $199 phone is an easier justification than anything in it's class) but it also would help supportability while increasing CSAT. Since this idea is a little different than anything we've done before, I understand it might require the sponsorship of somebody in your position to move fwd; as you manage both the iTunes and MM teams necessary to accomplish this.
>
> Ideally, this feature would ship OFF and could be enabled by a user with a MM account who choses to do so; we'd leverage [redacted] to find your iPhone in the same manner we do now for Macs using Back to my Mac. This would open a Google Map in Safari with your iPhone's current location on your primary sync computer when you chose the "Locate my iPhone" menu option in iTunes. It means we'd never have a lost iPhone again and could expand the feature to the Touch later.
>
> Thank you very much Eddy!
>
> cheers,
>
> Darren

EXHIBIT 4



From: **Scott Forstall** forstall@apple.com
Subject: Re: Radar 6262545
Date: February 18, 2009 at 9:44 PM
To: Darren Eastman deastman@apple.com

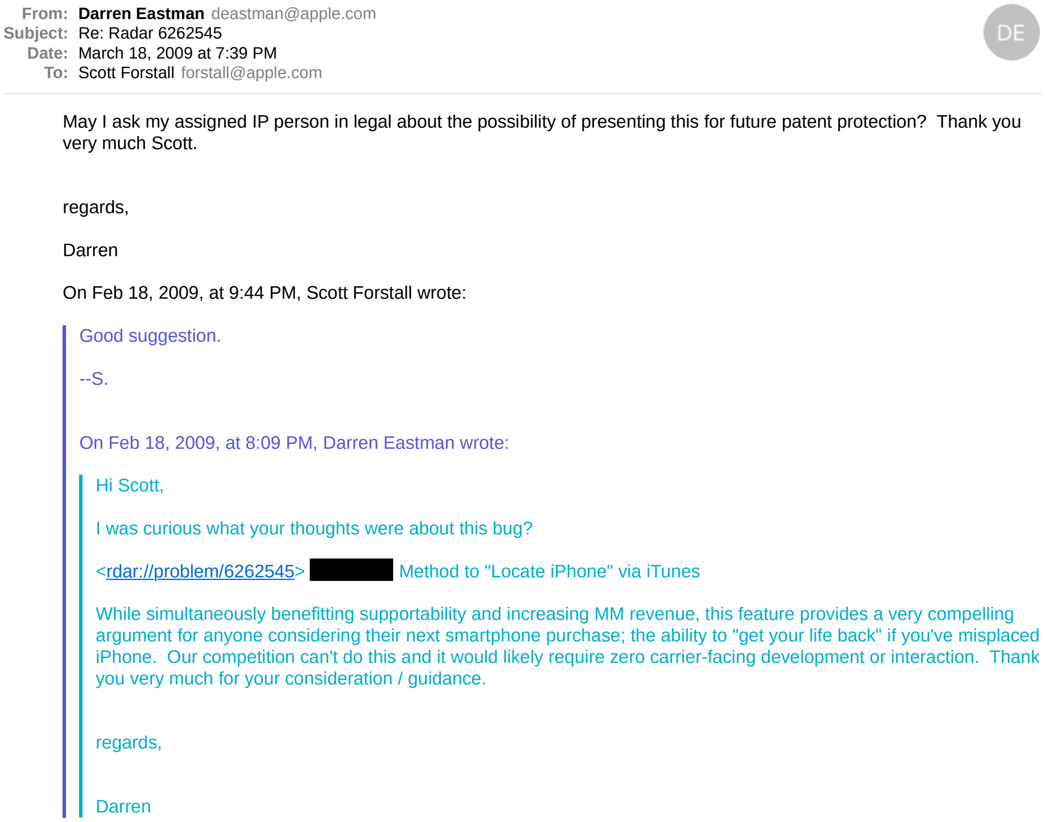
Good suggestion.

--S.

On Feb 18, 2009, at 8:09 PM, Darren Eastman wrote:

> Hi Scott,
>
> I was curious what your thoughts were about this bug?
>
> <rdar://problem/6262545> [redacted] Method to "Locate iPhone" via iTunes
>
> While simultaneously benefitting supportability and increasing MM revenue, this feature provides a very compelling argument for anyone considering their next smartphone purchase; the ability to "get your life back" if you've misplaced iPhone. Our competition can't do this and it would likely require zero carrier-facing development or interaction. Thank you very much for your consideration / guidance.
>
> regards,
>
> Darren

EXHIBIT 5



From: **Darren Eastman** deastman@apple.com
Subject: Re: Radar 6262545
Date: March 18, 2009 at 7:39 PM
To: Scott Forstall forstall@apple.com

May I ask my assigned IP person in legal about the possibility of presenting this for future patent protection? Thank you very much Scott.

regards,

Darren

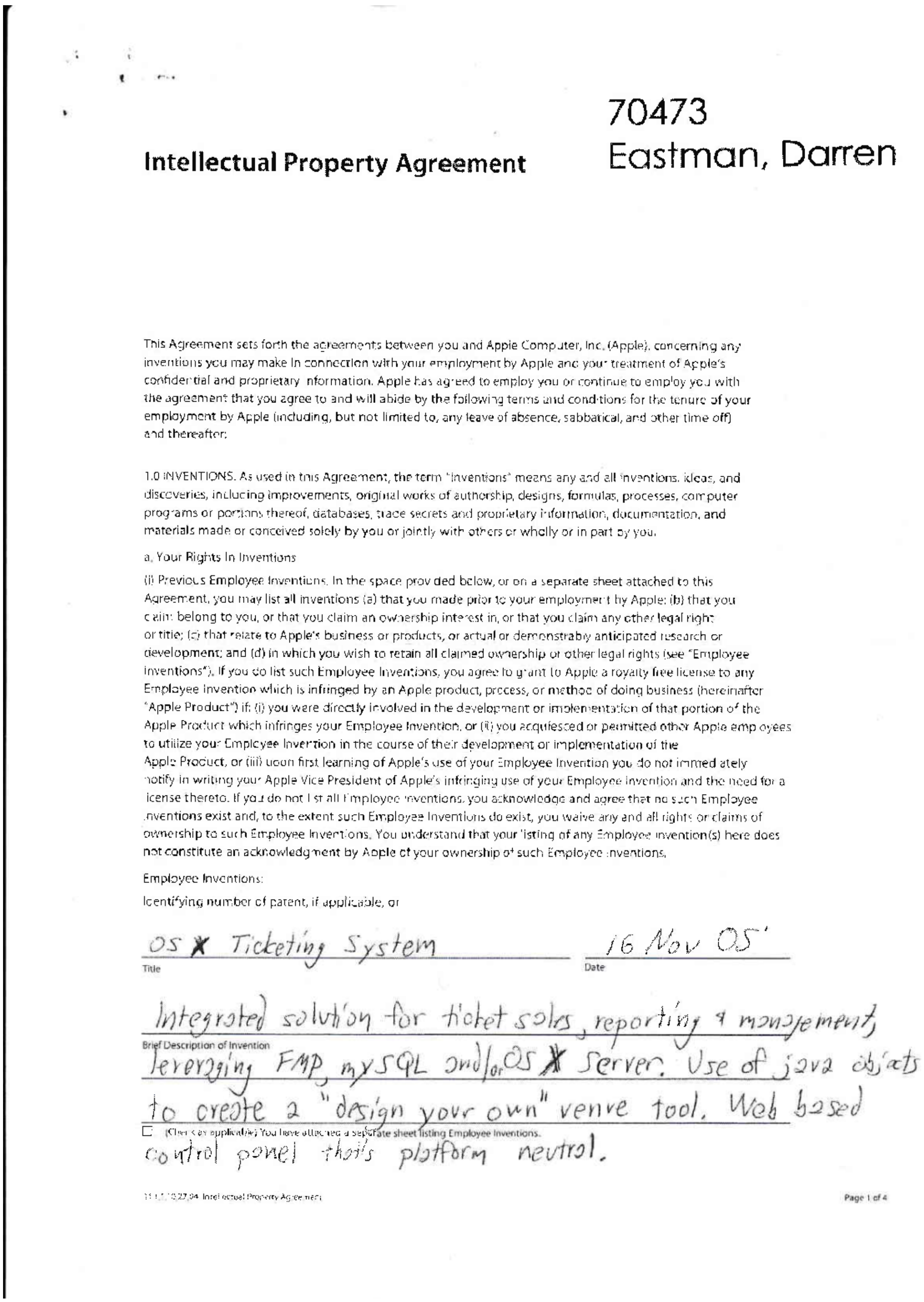
On Feb 18, 2009, at 9:44 PM, Scott Forstall wrote:

> Good suggestion.
>
> --S.
>
> On Feb 18, 2009, at 8:09 PM, Darren Eastman wrote:
>
>> Hi Scott,
>>
>> I was curious what your thoughts were about this bug?
>>
>> <rdar://problem/6262545> [redacted] Method to "Locate iPhone" via iTunes
>>
>> While simultaneously benefitting supportability and increasing MM revenue, this feature provides a very compelling argument for anyone considering their next smartphone purchase; the ability to "get your life back" if you've misplaced iPhone. Our competition can't do this and it would likely require zero carrier-facing development or interaction. Thank you very much for your consideration / guidance.
>>
>> regards,
>>
>> Darren

EXHIBIT 6

(Filed under seal, ECF No. 19)

EXHIBIT 7



70473
Eastman, Darren

Intellectual Property Agreement

This Agreement sets forth the agreements between you and Apple Computer, Inc. (Apple), concerning any inventions you may make in connection with your employment by Apple and your treatment of Apple's confidential and proprietary information. Apple has agreed to employ you or continue to employ you with the agreement that you agree to and will abide by the following terms and conditions for the tenure of your employment by Apple (including, but not limited to, any leave of absence, sabbatical, and other time off) and thereafter:

1.0 INVENTIONS. As used in this Agreement, the term "Inventions" means any and all inventions, ideas, and discoveries, including improvements, original works of authorship, designs, formulas, processes, computer programs or portions thereof, databases, trade secrets and proprietary information, documentation, and materials made or conceived solely by you or jointly with others or wholly or in part by you.

a. Your Rights In Inventions

(i) Previous Employee Inventions. In the space provided below, or on a separate sheet attached to this Agreement, you may list all inventions (a) that you made prior to your employment by Apple; (b) that you claim belong to you, or that you claim an ownership interest in, or that you claim any other legal right or title; (c) that relate to Apple's business or products, or actual or demonstrably anticipated research or development; and (d) in which you wish to retain all claimed ownership or other legal rights (see "Employee Inventions"). If you do list such Employee Inventions, you agree to grant to Apple a royalty free license to any Employee Invention which is infringed by an Apple product, process, or method of doing business (hereinafter "Apple Product") if: (i) you were directly involved in the development or implementation of that portion of the Apple Product which infringes your Employee Invention, or (ii) you acquiesced or permitted other Apple employees to utilize your Employee Invention in the course of their development or implementation of the Apple Product, or (iii) upon first learning of Apple's use of your Employee Invention you do not immediately notify in writing your Apple Vice President of Apple's infringing use of your Employee Invention and the need for a license thereto. If you do not list all Employee Inventions, you acknowledge and agree that no such Employee Inventions exist and, to the extent such Employee Inventions do exist, you waive any and all rights or claims of ownership to such Employee Inventions. You understand that your listing of any Employee Invention(s) here does not constitute an acknowledgment by Apple of your ownership of such Employee Inventions.

Employee Inventions:

Identifying number of patent, if applicable, or

OS X Ticketing System
Title

16 Nov 05'
Date

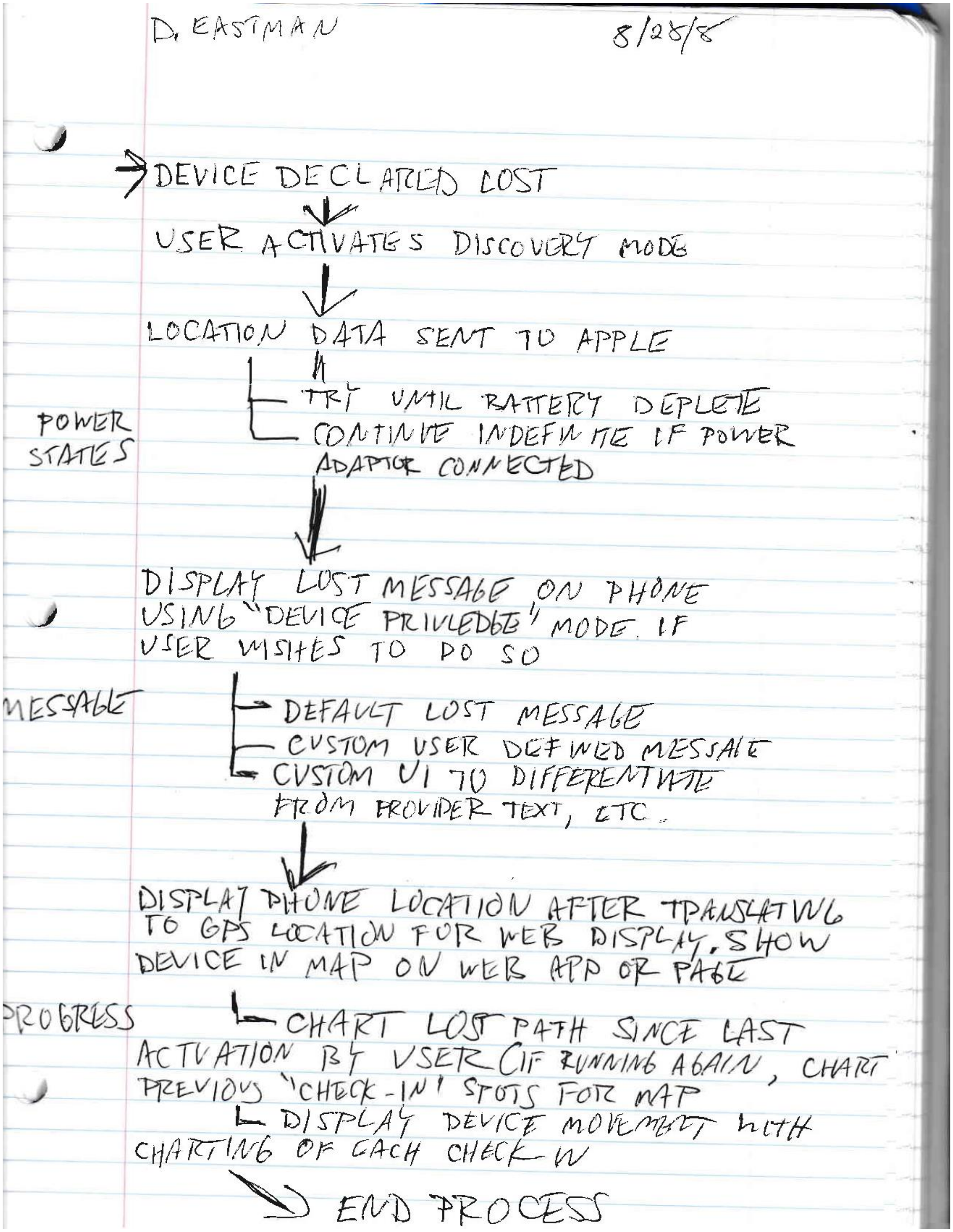
Integrated solution for ticket sales, reporting & management, leveraging FMP, mySQL and/or OS X Server. Use of java objects to create a "design your own" venue tool. Web based control panel that's platform neutral.
Brief Description of Invention

□ (Check as applicable) You have attached a separate sheet listing Employee Inventions.

11.1.1.10.27.04 Intellectual Property Agreement

Page 1 of 4

EXHIBIT 8



D. EASTMAN 8/28/8

→ DEVICE DECLARED LOST

↓

USER ACTIVATES DISCOVERY MODE

↓

LOCATION DATA SENT TO APPLE

POWER STATES

- TRY UNTIL BATTERY DEPLETE
- CONTINUE INDEFINITE IF POWER ADAPTOR CONNECTED

↓

DISPLAY LOST MESSAGE ON PHONE USING "DEVICE PRIVLEDGE" MODE. IF USER WISHES TO DO SO

MESSAGE

- DEFAULT LOST MESSAGE
- CUSTOM USER DEFINED MESSAGE
- CUSTOM UI TO DIFFERENTIATE FROM PROVIDER TEXT, ETC.

↓

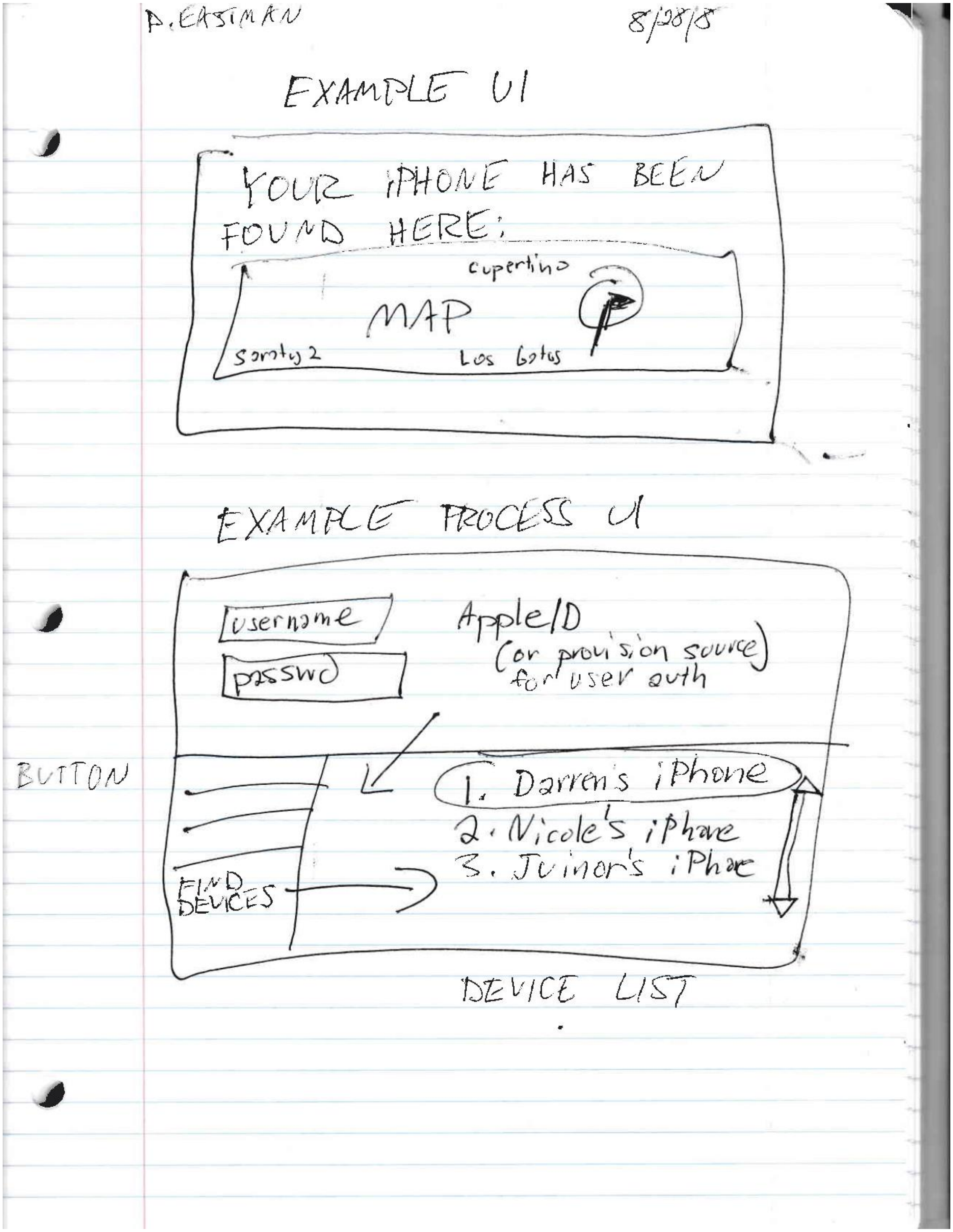
DISPLAY PHONE LOCATION AFTER TRANSLATING TO GPS LOCATION FOR WEB DISPLAY, SHOW DEVICE IN MAP ON WEB APP OR PAGE

PROGRESS

- CHART LOST PATH SINCE LAST ACTIVATION BY USER (IF RUNNING AGAIN, CHART PREVIOUS "CHECK-IN" SPOTS FOR MAP
- DISPLAY DEVICE MOVEMENT WITH CHARTING OF EACH CHECK IN

↘ END PROCESS

EXHIBIT 9



D. EASTMAN 8/28/8

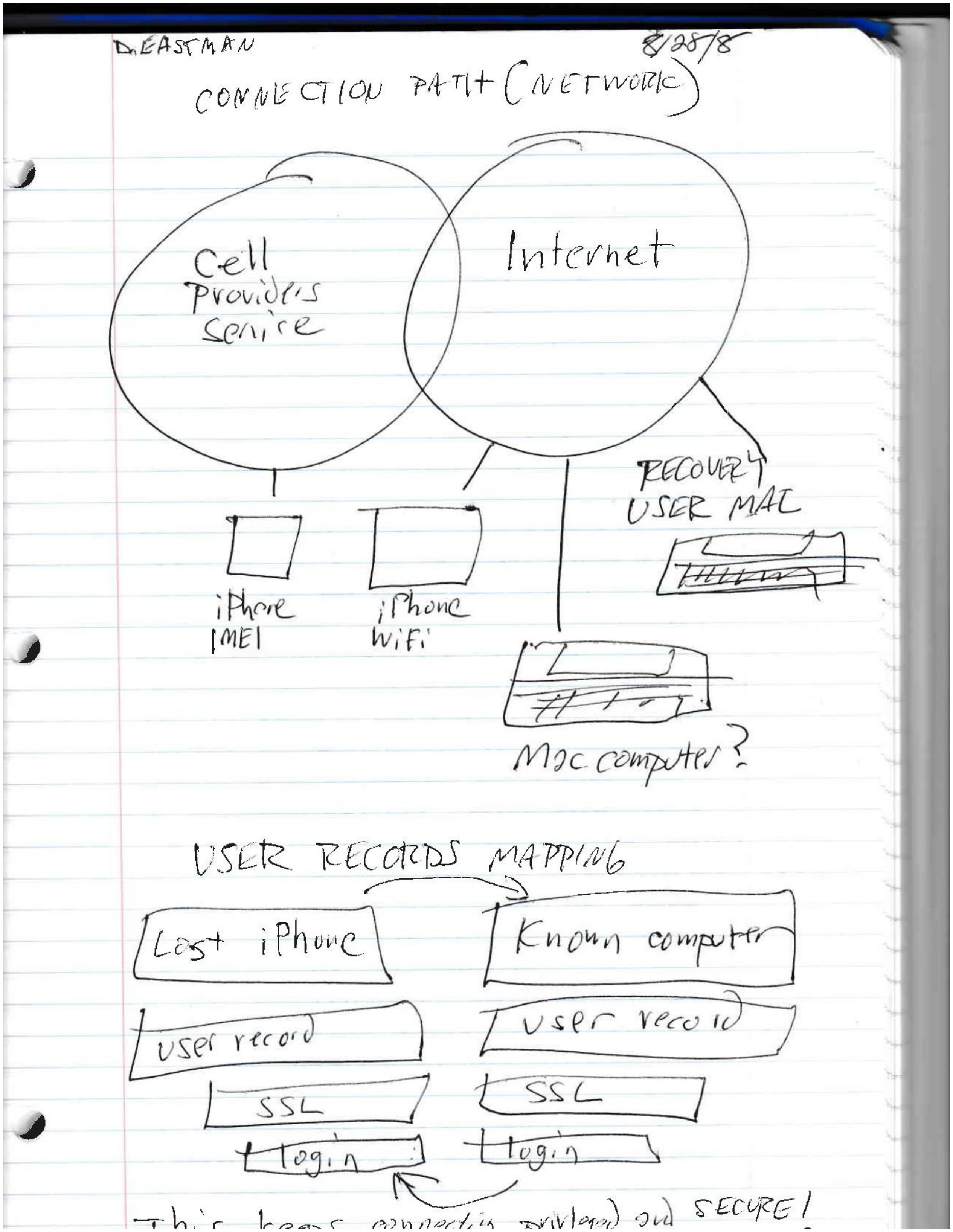
EXAMPLE UI

YOUR iPHONE HAS BEEN FOUND HERE:
cupertino
MAP
Saratoga
Los Gatos

EXAMPLE PROCESS UI

username
passwd
AppleID (or provision source for user auth
BUTTON
1. Darren's iPhone
2. Nicole's iPhone
3. Junior's iPhone
FIND DEVICES
DEVICE LIST

EXHIBIT 10



Dr. EASTMAN 8/28/18

CONNECTION PATH (NETWORK)

Cell Providers Service

Internet

iPhone IMEI

iPhone WiFi

RECOVERY USER MAC

Mac computer?

USER RECORDS MAPPING

Lost iPhone

Known computer

user record

user record

SSL

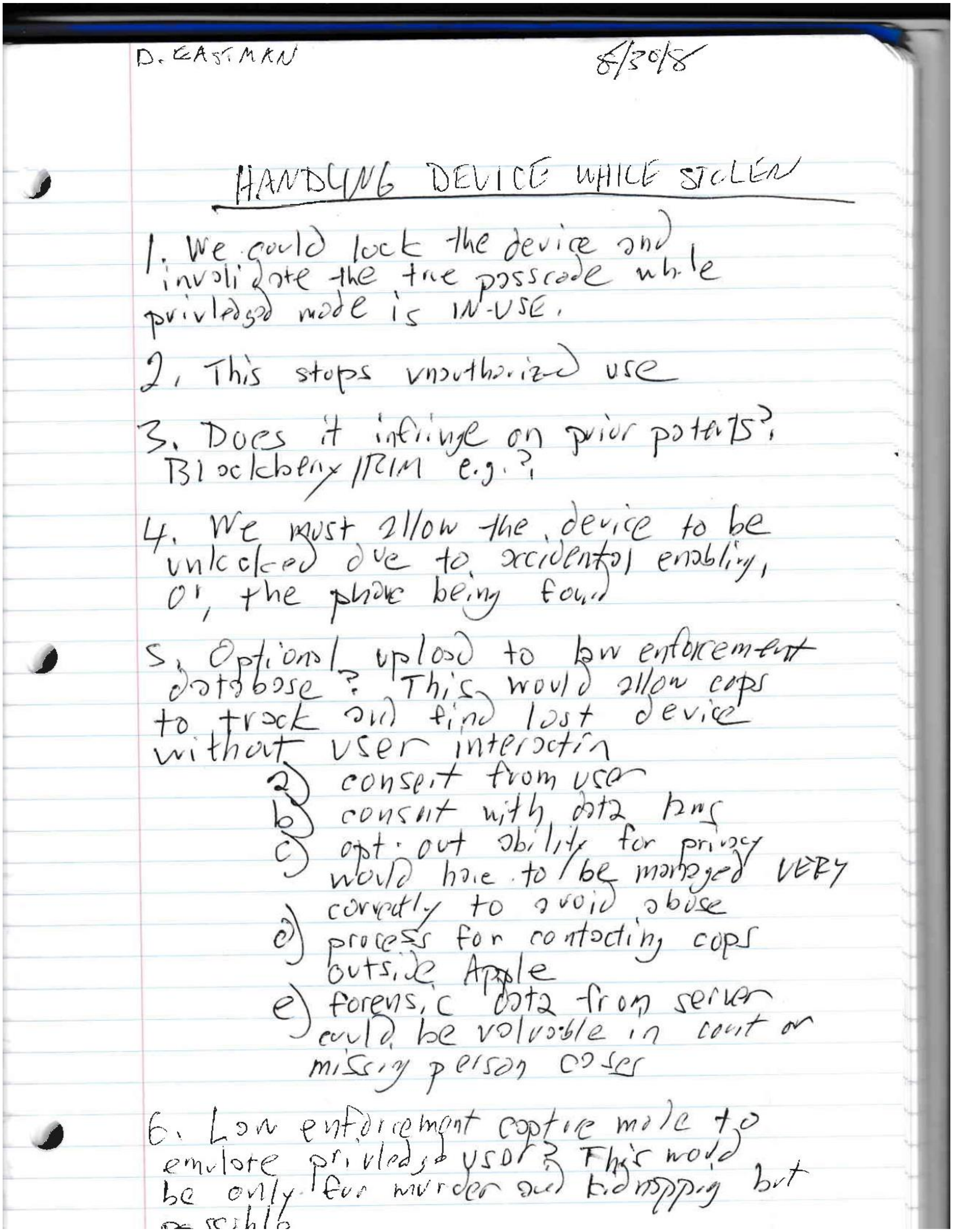
SSL

login

login

This keeps connectin privleged and SECURE!

EXHIBIT 11



D. EASTMAN 8/30/18

HANDLING DEVICE WHILE STOLEN

1. We could lock the device and invalidate the true passcode while privledged mode is IN-USE.

2. This stops unauthorized use

3. Does it infringe on prior patents? Blackberry/RIM e.g.?

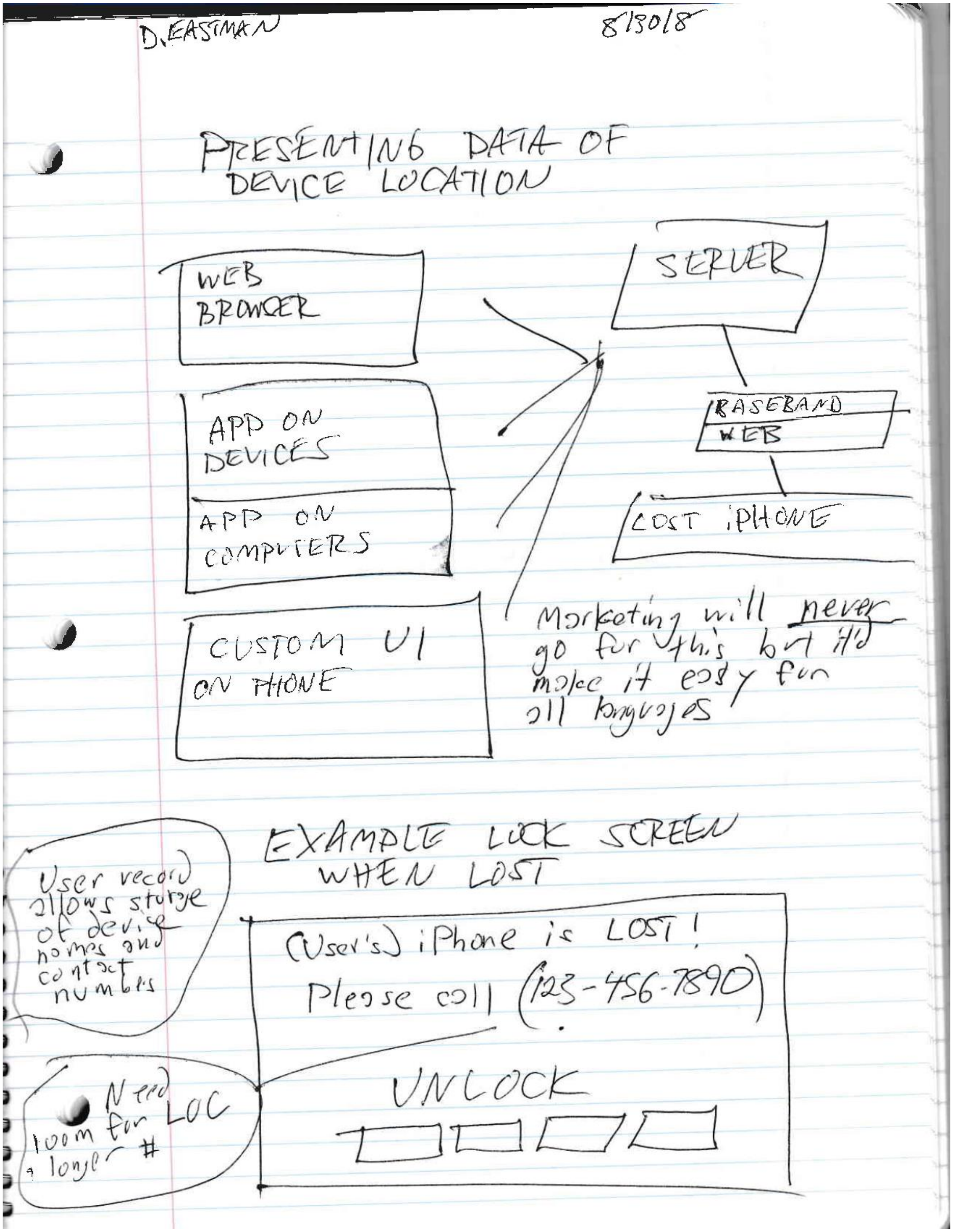
4. We must allow the device to be unlocked due to accidental enabling, or, the phone being found

5. Optional upload to law enforcement database? This would allow cops to track and find lost device without user interaction

   a) consent from user
   b) consult with data laws
   c) opt-out ability for privacy would have to be managed VERY correctly to avoid abuse
   d) process for contacting cops outside Apple
   e) forensic data from server could be valuable in court on missing person cases

6. Law enforcement capture mode to emulate privledged user? This would be only for murder and kidnapping but possible

EXHIBIT 12



D. EASTMAN

8/30/18

PRESENTING DATA OF DEVICE LOCATION

Marketing will never go for this but it'd make it easy for all languages

EXAMPLE LOCK SCREEN WHEN LOST

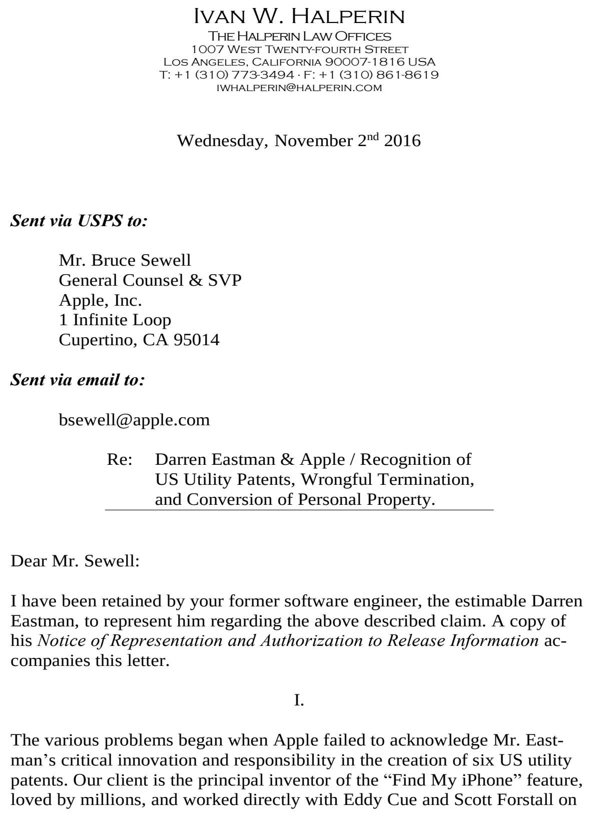
User record allows storage of device names and contact numbers

(User's) iPhone is LOST!

Please call (123-456-7890)

Need room for LOC + longer #

EXHIBIT 13



Ivan W. Halperin
The Halperin Law Offices
1007 West Twenty-fourth Street
Los Angeles, California 90007-1816 USA
T: +1 (310) 773-3494 · F: +1 (310) 861-8619
iwhalperin@halperin.com

Wednesday, November 2nd 2016

Sent via USPS to:

Mr. Bruce Sewell
General Counsel & SVP
Apple, Inc.
1 Infinite Loop
Cupertino, CA 95014

Sent via email to:

bsewell@apple.com

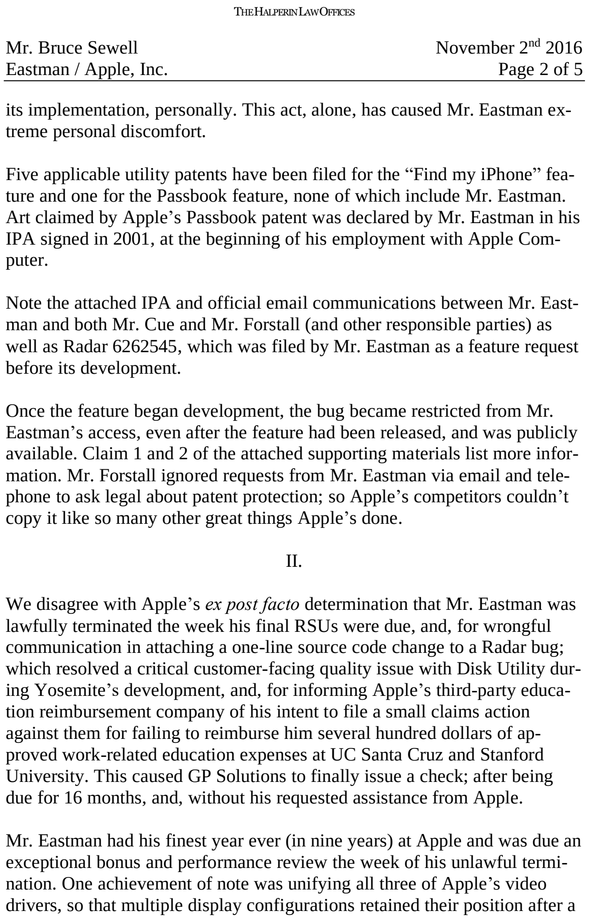
Re: Darren Eastman & Apple / Recognition of
US Utility Patents, Wrongful Termination,
and Conversion of Personal Property.

Dear Mr. Sewell:

I have been retained by your former software engineer, the estimable Darren Eastman, to represent him regarding the above described claim. A copy of his *Notice of Representation and Authorization to Release Information* accompanies this letter.

I.

The various problems began when Apple failed to acknowledge Mr. Eastman's critical innovation and responsibility in the creation of six US utility patents. Our client is the principal inventor of the "Find My iPhone" feature, loved by millions, and worked directly with Eddy Cue and Scott Forstall on



its implementation, personally. This act, alone, has caused Mr. Eastman extreme personal discomfort.

Five applicable utility patents have been filed for the “Find my iPhone” feature and one for the Passbook feature, none of which include Mr. Eastman. Art claimed by Apple’s Passbook patent was declared by Mr. Eastman in his IPA signed in 2001, at the beginning of his employment with Apple Computer.

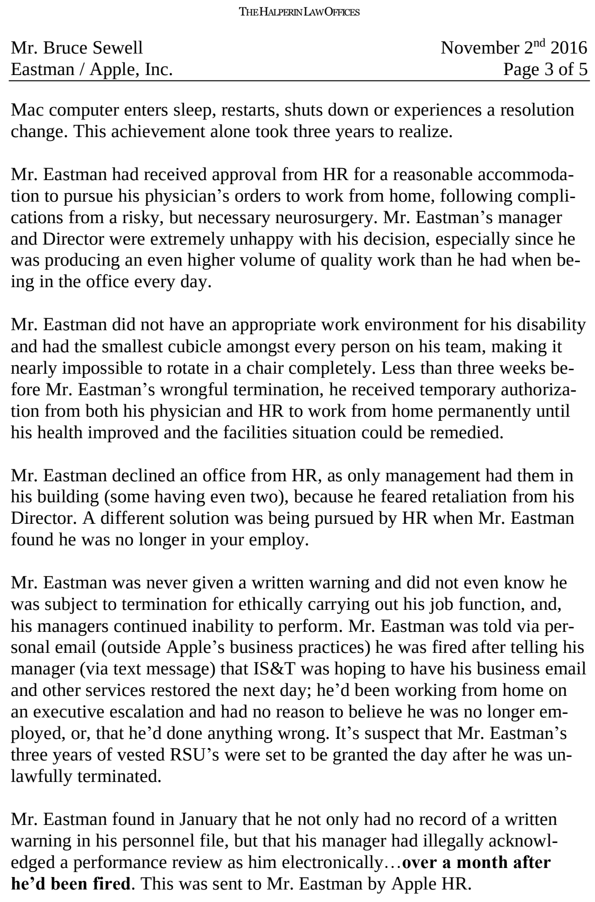
Note the attached IPA and official email communications between Mr. Eastman and both Mr. Cue and Mr. Forstall (and other responsible parties) as well as Radar 6262545, which was filed by Mr. Eastman as a feature request before its development.

Once the feature began development, the bug became restricted from Mr. Eastman’s access, even after the feature had been released, and was publicly available. Claim 1 and 2 of the attached supporting materials list more information. Mr. Forstall ignored requests from Mr. Eastman via email and telephone to ask legal about patent protection; so Apple’s competitors couldn’t copy it like so many other great things Apple’s done.

II.

We disagree with Apple’s *ex post facto* determination that Mr. Eastman was lawfully terminated the week his final RSUs were due, and, for wrongful communication in attaching a one-line source code change to a Radar bug; which resolved a critical customer-facing quality issue with Disk Utility during Yosemite’s development, and, for informing Apple’s third-party education reimbursement company of his intent to file a small claims action against them for failing to reimburse him several hundred dollars of approved work-related education expenses at UC Santa Cruz and Stanford University. This caused GP Solutions to finally issue a check; after being due for 16 months, and, without his requested assistance from Apple.

Mr. Eastman had his finest year ever (in nine years) at Apple and was due an exceptional bonus and performance review the week of his unlawful termination. One achievement of note was unifying all three of Apple’s video drivers, so that multiple display configurations retained their position after a



Mac computer enters sleep, restarts, shuts down or experiences a resolution change. This achievement alone took three years to realize.

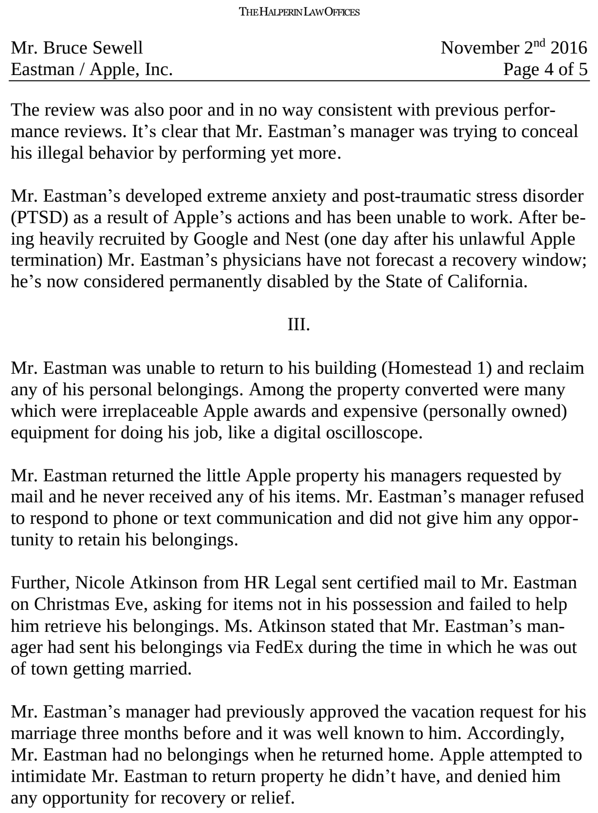
Mr. Eastman had received approval from HR for a reasonable accommodation to pursue his physician's orders to work from home, following complications from a risky, but necessary neurosurgery. Mr. Eastman's manager and Director were extremely unhappy with his decision, especially since he was producing an even higher volume of quality work than he had when being in the office every day.

Mr. Eastman did not have an appropriate work environment for his disability and had the smallest cubicle amongst every person on his team, making it nearly impossible to rotate in a chair completely. Less than three weeks before Mr. Eastman's wrongful termination, he received temporary authorization from both his physician and HR to work from home permanently until his health improved and the facilities situation could be remedied.

Mr. Eastman declined an office from HR, as only management had them in his building (some having even two), because he feared retaliation from his Director. A different solution was being pursued by HR when Mr. Eastman found he was no longer in your employ.

Mr. Eastman was never given a written warning and did not even know he was subject to termination for ethically carrying out his job function, and, his managers continued inability to perform. Mr. Eastman was told via personal email (outside Apple's business practices) he was fired after telling his manager (via text message) that IS&T was hoping to have his business email and other services restored the next day; he'd been working from home on an executive escalation and had no reason to believe he was no longer employed, or, that he'd done anything wrong. It's suspect that Mr. Eastman's three years of vested RSU's were set to be granted the day after he was unlawfully terminated.

Mr. Eastman found in January that he not only had no record of a written warning in his personnel file, but that his manager had illegally acknowledged a performance review as him electronically…**over a month after he'd been fired**. This was sent to Mr. Eastman by Apple HR.



The review was also poor and in no way consistent with previous performance reviews. It's clear that Mr. Eastman's manager was trying to conceal his illegal behavior by performing yet more.

Mr. Eastman's developed extreme anxiety and post-traumatic stress disorder (PTSD) as a result of Apple's actions and has been unable to work. After being heavily recruited by Google and Nest (one day after his unlawful Apple termination) Mr. Eastman's physicians have not forecast a recovery window; he's now considered permanently disabled by the State of California.

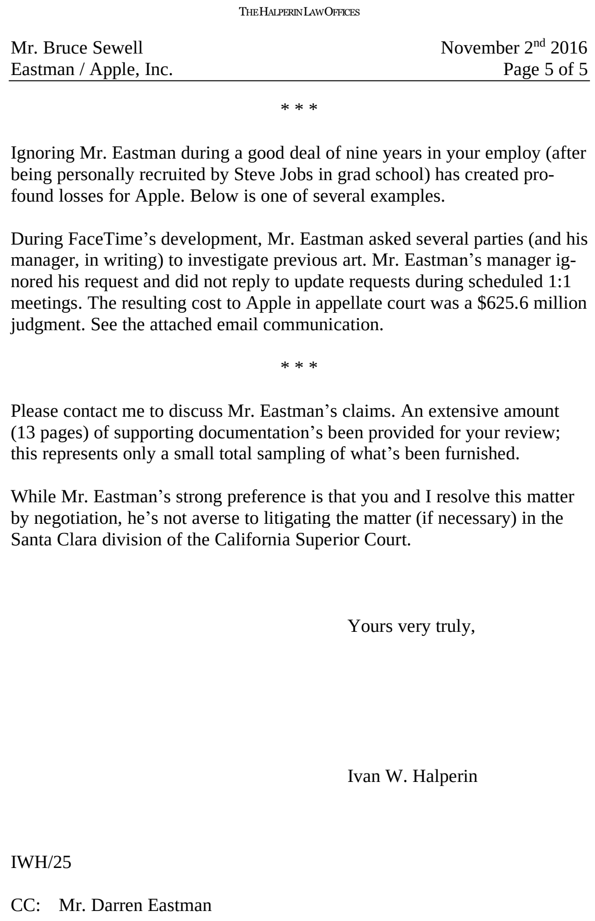
III.

Mr. Eastman was unable to return to his building (Homestead 1) and reclaim any of his personal belongings. Among the property converted were many which were irreplaceable Apple awards and expensive (personally owned) equipment for doing his job, like a digital oscilloscope.

Mr. Eastman returned the little Apple property his managers requested by mail and he never received any of his items. Mr. Eastman's manager refused to respond to phone or text communication and did not give him any opportunity to retain his belongings.

Further, Nicole Atkinson from HR Legal sent certified mail to Mr. Eastman on Christmas Eve, asking for items not in his possession and failed to help him retrieve his belongings. Ms. Atkinson stated that Mr. Eastman's manager had sent his belongings via FedEx during the time in which he was out of town getting married.

Mr. Eastman's manager had previously approved the vacation request for his marriage three months before and it was well known to him. Accordingly, Mr. Eastman had no belongings when he returned home. Apple attempted to intimidate Mr. Eastman to return property he didn't have, and denied him any opportunity for recovery or relief.



* * *

Ignoring Mr. Eastman during a good deal of nine years in your employ (after being personally recruited by Steve Jobs in grad school) has created profound losses for Apple. Below is one of several examples.

During FaceTime's development, Mr. Eastman asked several parties (and his manager, in writing) to investigate previous art. Mr. Eastman's manager ignored his request and did not reply to update requests during scheduled 1:1 meetings. The resulting cost to Apple in appellate court was a $625.6 million judgment. See the attached email communication.

* * *

Please contact me to discuss Mr. Eastman's claims. An extensive amount (13 pages) of supporting documentation's been provided for your review; this represents only a small total sampling of what's been furnished.

While Mr. Eastman's strong preference is that you and I resolve this matter by negotiation, he's not averse to litigating the matter (if necessary) in the Santa Clara division of the California Superior Court.

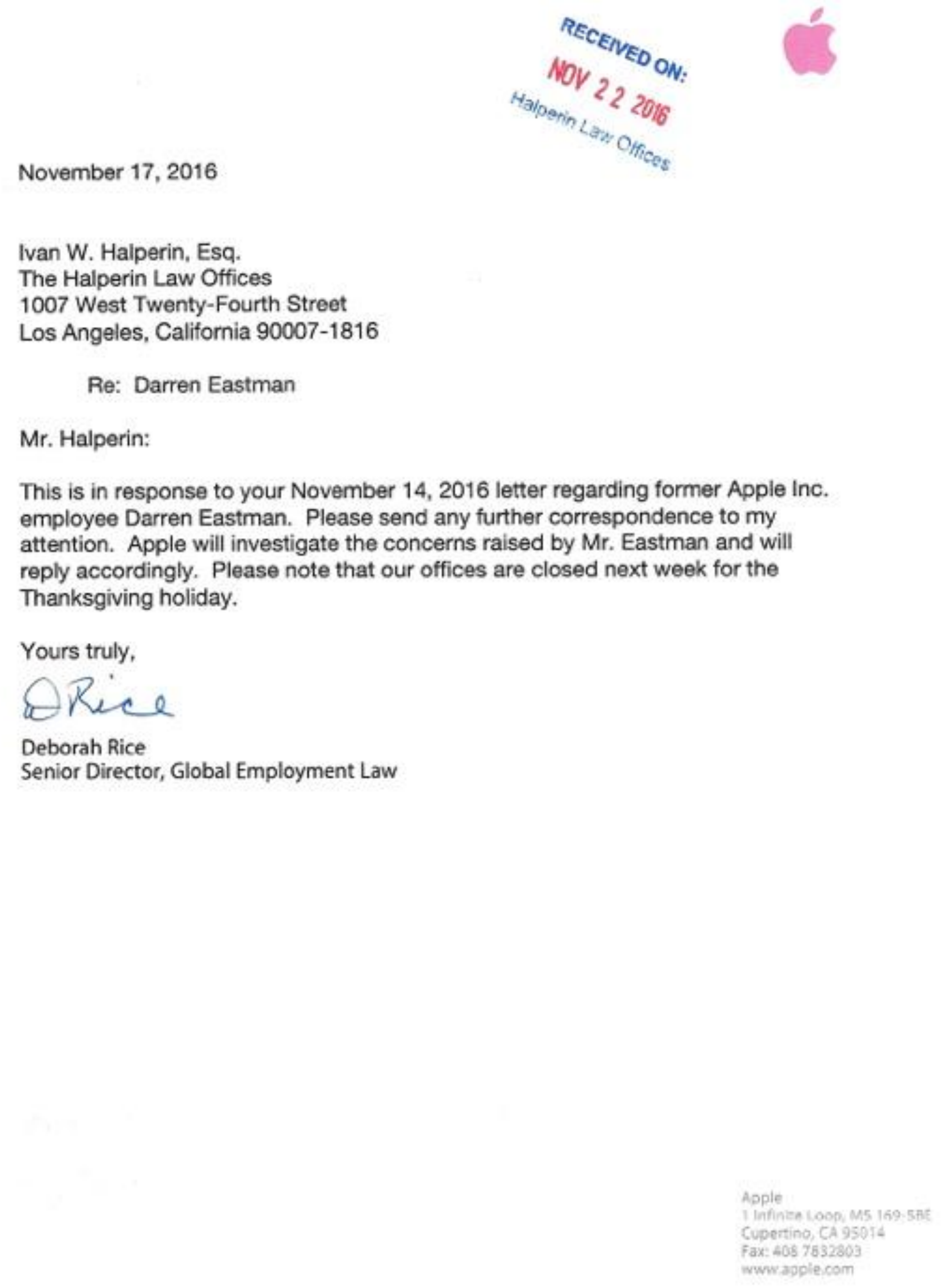
Yours very truly,

Ivan W. Halperin

IWH/25

CC: Mr. Darren Eastman

EXHIBIT 14



November 17, 2016

Ivan W. Halperin, Esq.
The Halperin Law Offices
1007 West Twenty-Fourth Street
Los Angeles, California 90007-1816

Re: Darren Eastman

Mr. Halperin:

This is in response to your November 14, 2016 letter regarding former Apple Inc. employee Darren Eastman. Please send any further correspondence to my attention. Apple will investigate the concerns raised by Mr. Eastman and will reply accordingly. Please note that our offices are closed next week for the Thanksgiving holiday.

Yours truly,

D Rice

Deborah Rice
Senior Director, Global Employment Law

Apple
1 Infinite Loop, MS 169-5BE
Cupertino, CA 95014
Fax: 408 7832803
www.apple.com

EXHIBIT 15

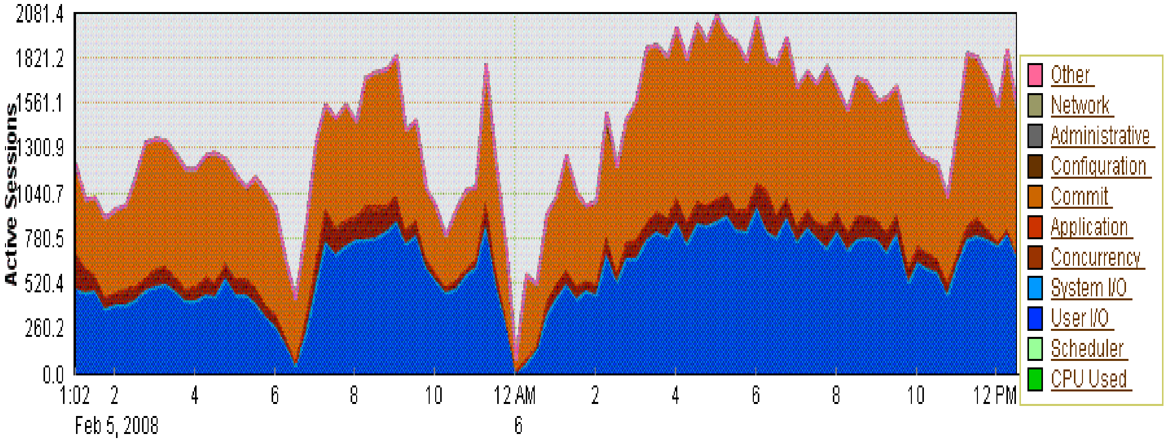
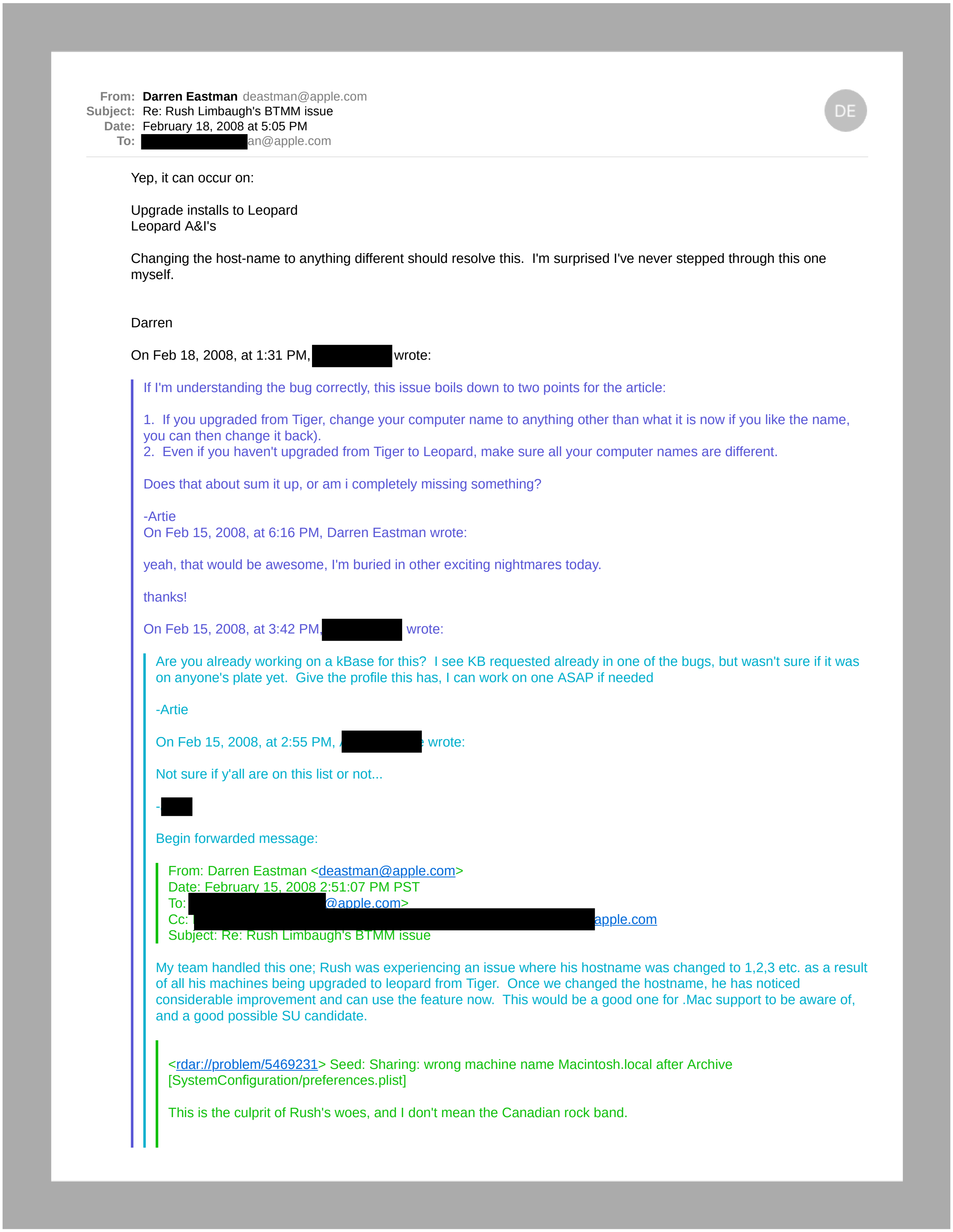

2081.4
1821.2
1561.1
1300.9
1040.7
780.5
520.4
260.2
0.0
Active Sessions
1:02
2
4
6
8
10
12 AM
2
4
6
8
10
12 PM
Feb 5, 2008
6
Other
Network
Administrative
Configuration
Commit
Application
Concurrency
System I/O
User I/O
Scheduler
CPU Used

EXHIBIT 16



From: **Darren Eastman** deastman@apple.com
Subject: Re: Rush Limbaugh's BTMM issue
Date: February 18, 2008 at 5:05 PM
To: [redacted]an@apple.com

Yep, it can occur on:

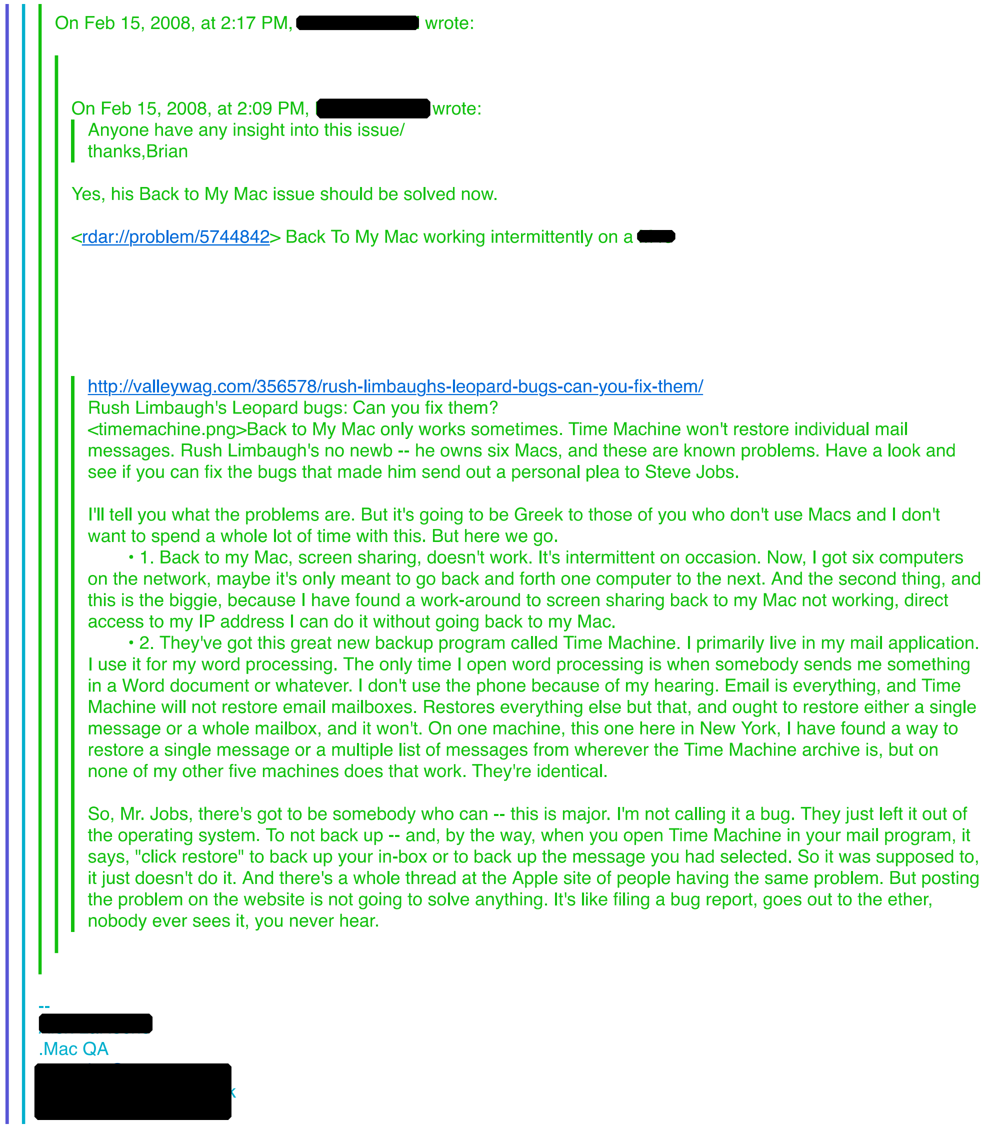
Upgrade installs to Leopard
Leopard A&I's

Changing the host-name to anything different should resolve this. I'm surprised I've never stepped through this one myself.

Darren

On Feb 18, 2008, at 1:31 PM, [redacted] wrote:

> If I'm understanding the bug correctly, this issue boils down to two points for the article:
>
> 1. If you upgraded from Tiger, change your computer name to anything other than what it is now if you like the name, you can then change it back).
> 2. Even if you haven't upgraded from Tiger to Leopard, make sure all your computer names are different.
>
> Does that about sum it up, or am i completely missing something?
>
> -Artie
> On Feb 15, 2008, at 6:16 PM, Darren Eastman wrote:
>
> yeah, that would be awesome, I'm buried in other exciting nightmares today.
>
> thanks!
>
> On Feb 15, 2008, at 3:42 PM,[redacted] wrote:
>
>> Are you already working on a kBase for this? I see KB requested already in one of the bugs, but wasn't sure if it was on anyone's plate yet. Give the profile this has, I can work on one ASAP if needed
>>
>> -Artie
>>
>> On Feb 15, 2008, at 2:55 PM, [redacted] wrote:
>>
>> Not sure if y'all are on this list or not...
>>
>> -[redacted]
>>
>> Begin forwarded message:
>>
>>> From: Darren Eastman <deastman@apple.com>
>>> Date: February 15, 2008 2:51:07 PM PST
>>> To: [redacted]@apple.com>
>>> Cc: [redacted]apple.com
>>> Subject: Re: Rush Limbaugh's BTMM issue
>>
>> My team handled this one; Rush was experiencing an issue where his hostname was changed to 1,2,3 etc. as a result of all his machines being upgraded to leopard from Tiger. Once we changed the hostname, he has noticed considerable improvement and can use the feature now. This would be a good one for .Mac support to be aware of, and a good possible SU candidate.
>>
>>> <rdar://problem/5469231> Seed: Sharing: wrong machine name Macintosh.local after Archive [SystemConfiguration/preferences.plist]
>>>
>>> This is the culprit of Rush's woes, and I don't mean the Canadian rock band.



> > > On Feb 15, 2008, at 2:17 PM, [redacted] wrote:
> > >
> > > > On Feb 15, 2008, at 2:09 PM, [redacted] wrote:
> > > >
> > > > > Anyone have any insight into this issue/
> > > > > thanks,Brian
> > > >
> > > > Yes, his Back to My Mac issue should be solved now.
> > > >
> > > > <rdar://problem/5744842> Back To My Mac working intermittently on a [redacted]
> > > >
> > > > > http://valleywag.com/356578/rush-limbaughs-leopard-bugs-can-you-fix-them/
> > > > > Rush Limbaugh's Leopard bugs: Can you fix them?
> > > > > <timemachine.png>Back to My Mac only works sometimes. Time Machine won't restore individual mail messages. Rush Limbaugh's no newb -- he owns six Macs, and these are known problems. Have a look and see if you can fix the bugs that made him send out a personal plea to Steve Jobs.
> > > > >
> > > > > I'll tell you what the problems are. But it's going to be Greek to those of you who don't use Macs and I don't want to spend a whole lot of time with this. But here we go.
> > > > > • 1. Back to my Mac, screen sharing, doesn't work. It's intermittent on occasion. Now, I got six computers on the network, maybe it's only meant to go back and forth one computer to the next. And the second thing, and this is the biggie, because I have found a work-around to screen sharing back to my Mac not working, direct access to my IP address I can do it without going back to my Mac.
> > > > > • 2. They've got this great new backup program called Time Machine. I primarily live in my mail application. I use it for my word processing. The only time I open word processing is when somebody sends me something in a Word document or whatever. I don't use the phone because of my hearing. Email is everything, and Time Machine will not restore email mailboxes. Restores everything else but that, and ought to restore either a single message or a whole mailbox, and it won't. On one machine, this one here in New York, I have found a way to restore a single message or a multiple list of messages from wherever the Time Machine archive is, but on none of my other five machines does that work. They're identical.
> > > > >
> > > > > So, Mr. Jobs, there's got to be somebody who can -- this is major. I'm not calling it a bug. They just left it out of the operating system. To not back up -- and, by the way, when you open Time Machine in your mail program, it says, "click restore" to back up your in-box or to back up the message you had selected. So it was supposed to, it just doesn't do it. And there's a whole thread at the Apple site of people having the same problem. But posting the problem on the website is not going to solve anything. It's like filing a bug report, goes out to the ether, nobody ever sees it, you never hear.
> >
> > --
> > [redacted]
> > .Mac QA
> > [redacted]

EXHIBIT 17

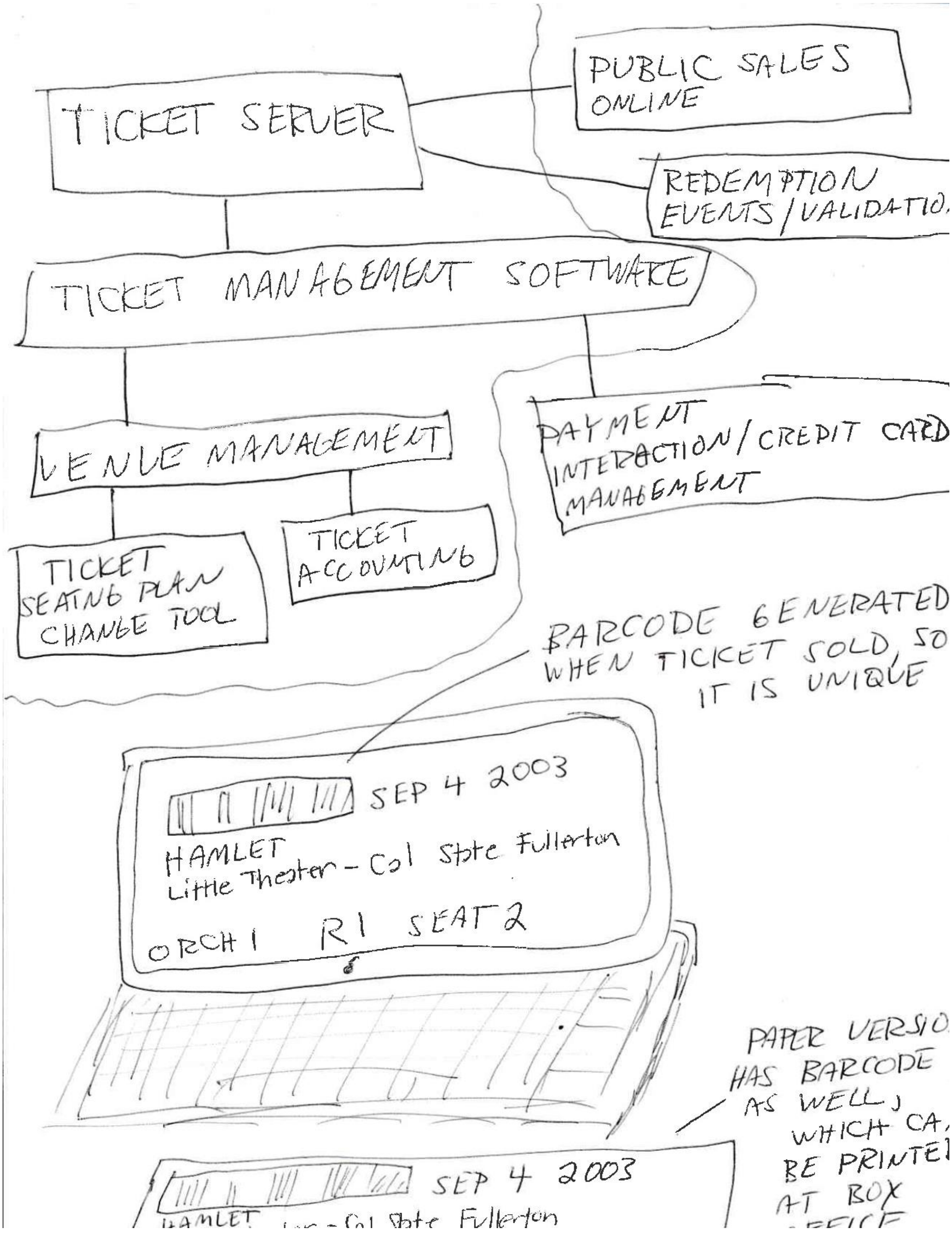
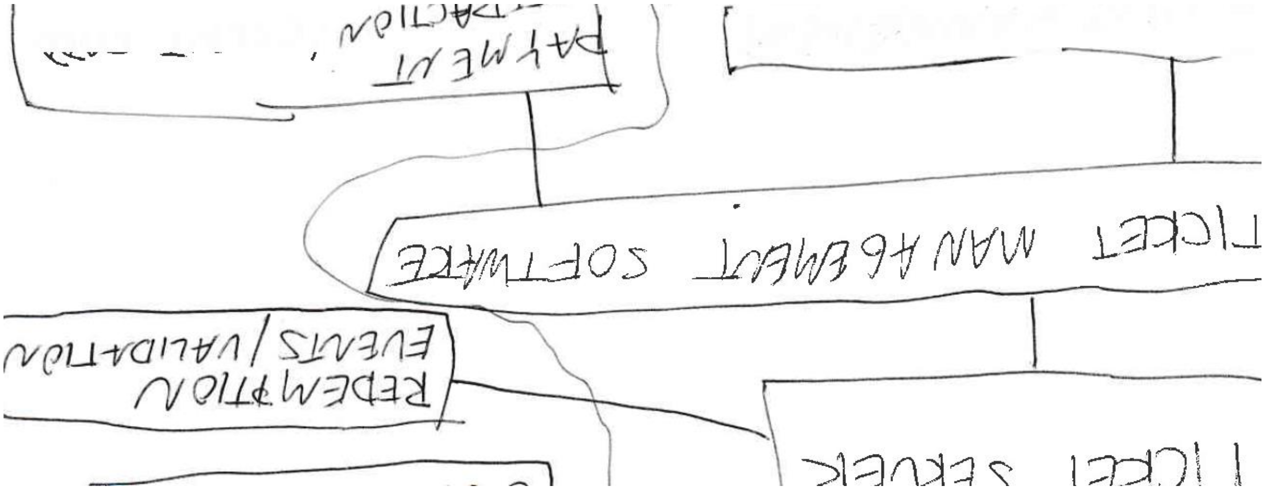

TICKET SERVER
PUBLIC SALES ONLINE
REDEMPTION EVENTS / VALIDATIO
TICKET MANAGEMENT SOFTWARE
VENUE MANAGEMENT
PAYMENT INTERACTION / CREDIT CARD MANAGEMENT
TICKET SEATING PLAN CHANGE TOOL
TICKET ACCOUNTING
BARCODE GENERATED WHEN TICKET SOLD, SO IT IS UNIQUE
SEP 4 2003
HAMLET
Little Theater - Cal State Fullerton
ORCH 1 R1 SEAT 2
PAPER VERSIO HAS BARCODE AS WELL, WHICH CA BE PRINTE AT BOX OFFICE
SEP 4 2003
HAMLET
ter - Cal State Fullerton

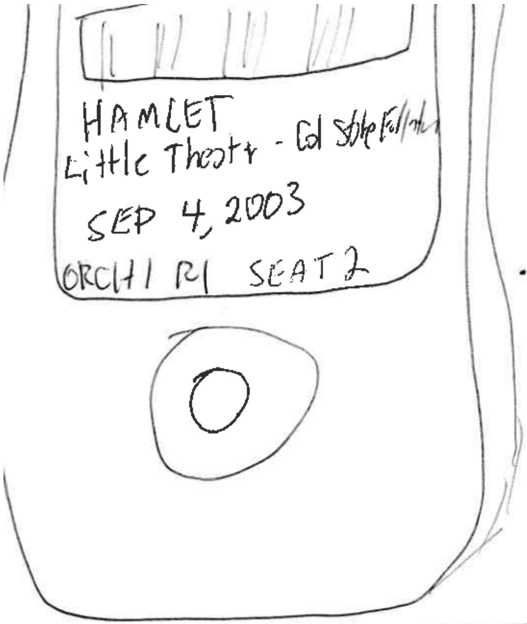
EXHIBIT 18
(Obscured portion of EXHIBIT 17)



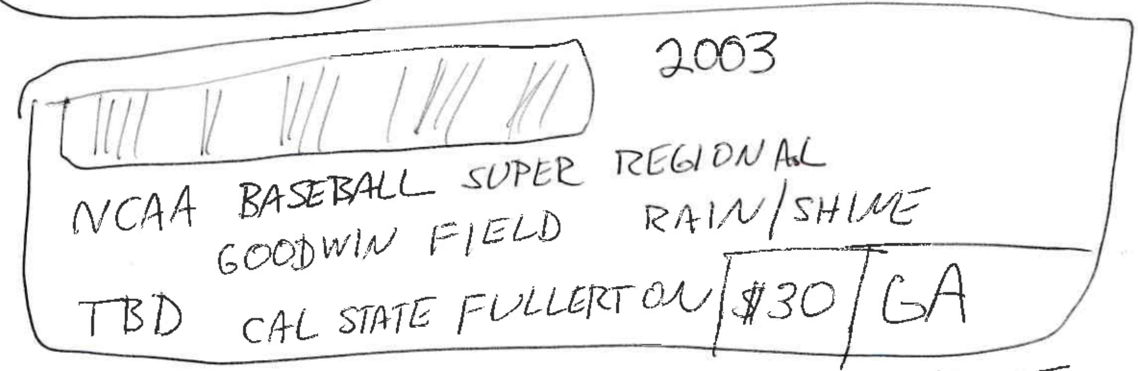
TICKET SERVER
REDEMPTION EVENTS/VALIDATION
TICKET MANAGEMENT SOFTWARE
PAYMENT

EXHIBIT 19

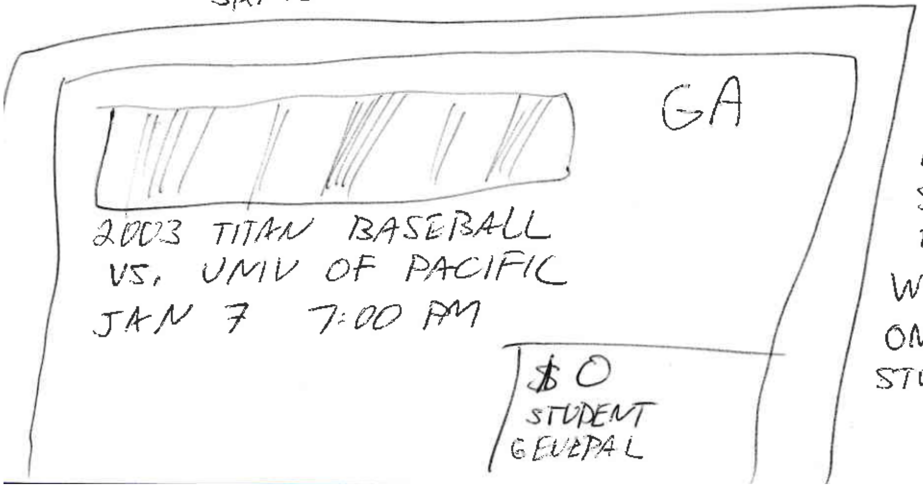
VAT
THE TURNKEY SOLUTION
TO AIRPORTS AND
EDUCATION



PERSONAL
DIGITAL
ASSISTANT
· EXAMPLE



PAPER EXAMPLE WITH UNIQUE BARCODE
SOLD AT SPORTS BOX OFFICE, BUT USING
SAME SERVER AS PERFORMING ARTS.



PDA SPORTS
EXAMPLE
FOR FREE
STUDENT
TICKET THAT
WAS ISSUED
ONLINE FOR
STUDENT VERIFICATION

EXHIBIT 20

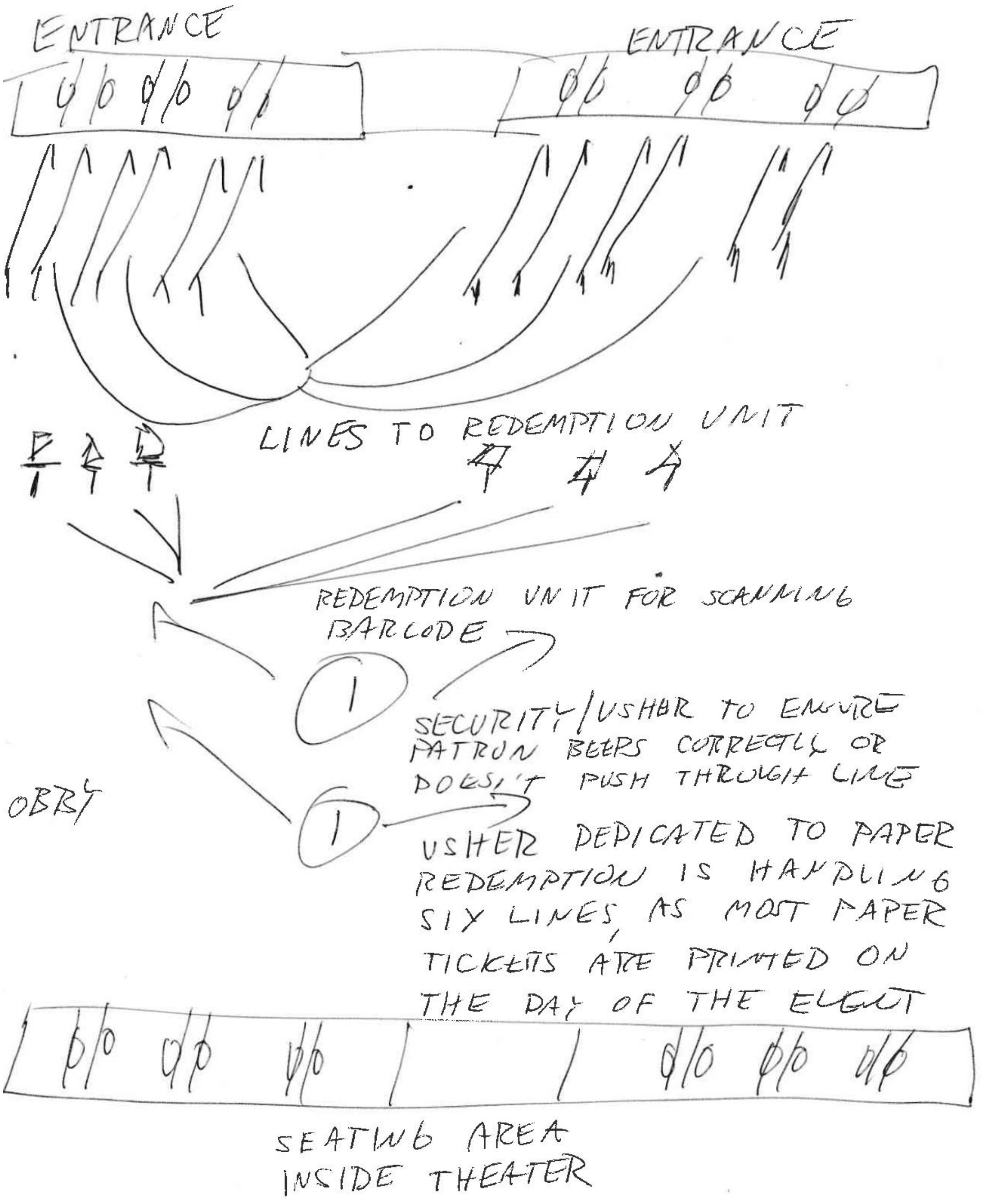

ENTRANCE
ENTRANCE
LINES TO REDEMPTION UNIT
REDEMPTION UNIT FOR SCANNING BARCODE
1
SECURITY/USHER TO ENSURE PATRON BEEPS CORRECTLY OR DOESN'T PUSH THROUGH LINE
OBBY
1
USHER DEDICATED TO PAPER REDEMPTION IS HANDLING SIX LINES, AS MOST PAPER TICKETS ARE PRINTED ON THE DAY OF THE EVENT
SEATING AREA INSIDE THEATER

EXHIBIT 21

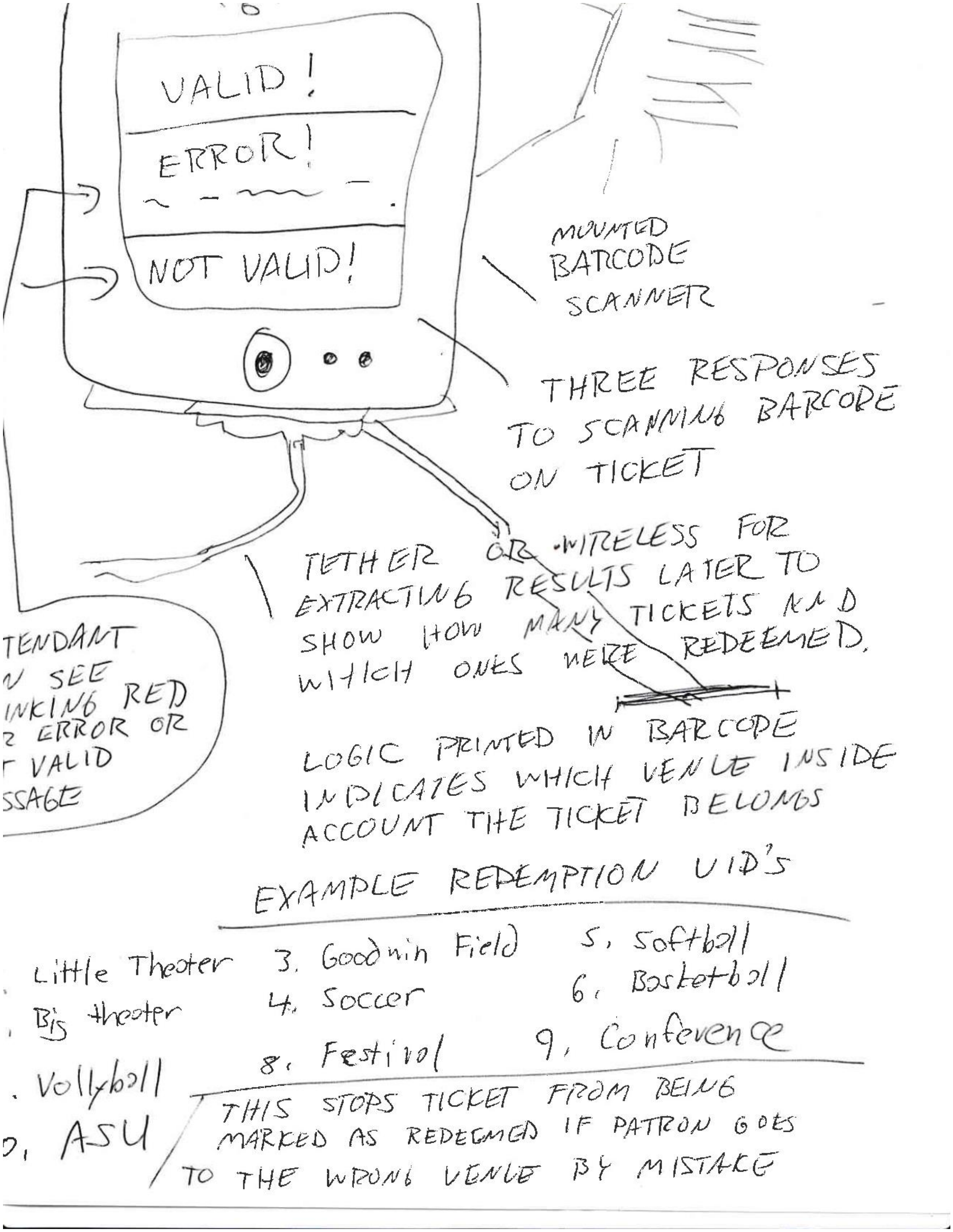
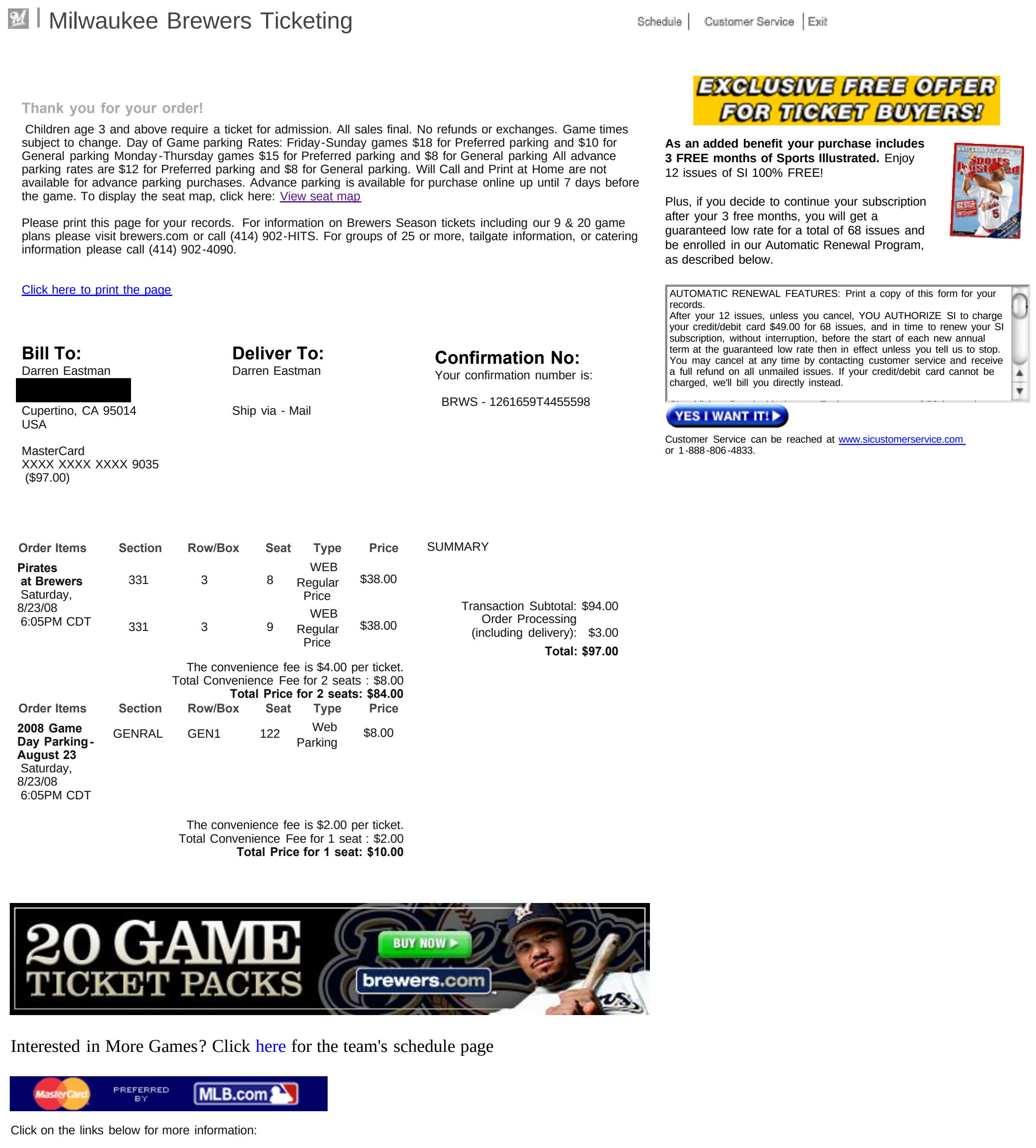

VALID!
ERROR!
NOT VALID!
MOUNTED BARCODE SCANNER
THREE RESPONSES TO SCANNING BARCODE ON TICKET
TETHER OR WIRELESS FOR EXTRACTING RESULTS LATER TO SHOW HOW MANY TICKETS AND WHICH ONES WERE REDEEMED.
TENDANT N SEE INKING RED 2 ERROR OR T VALID SSAGE
LOGIC PRINTED IN BARCODE INDICATES WHICH VENUE INSIDE ACCOUNT THE TICKET BELONGS
EXAMPLE REDEMPTION UID'S
Little Theater
Big theater
3. Goodwin Field
4. Soccer
5. softball
6. Basketball
Vollyball
8. Festival
9. Conference
0, ASU
THIS STOPS TICKET FROM BEING MARKED AS REDEEMED IF PATRON GOES TO THE WRONG VENUE BY MISTAKE

EXHIBIT 22



Milwaukee Brewers Ticketing

Schedule | Customer Service | Exit

Thank you for your order!

Children age 3 and above require a ticket for admission. All sales final. No refunds or exchanges. Game times subject to change. Day of Game parking Rates: Friday-Sunday games $18 for Preferred parking and $10 for General parking Monday-Thursday games $15 for Preferred parking and $8 for General parking All advance parking rates are $12 for Preferred parking and $8 for General parking. Will Call and Print at Home are not available for advance parking purchases. Advance parking is available for purchase online up until 7 days before the game. To display the seat map, click here: View seat map

Please print this page for your records. For information on Brewers Season tickets including our 9 & 20 game plans please visit brewers.com or call (414) 902-HITS. For groups of 25 or more, tailgate information, or catering information please call (414) 902-4090.

Click here to print the page

Bill To:
Darren Eastman
[redacted]
Cupertino, CA 95014
USA

MasterCard
XXXX XXXX XXXX 9035
($97.00)

Deliver To:
Darren Eastman

Ship via - Mail

Confirmation No:
Your confirmation number is:

BRWS - 1261659T4455598

Order Items	Section	Row/Box	Seat	Type	Price
Pirates at Brewers Saturday, 8/23/08 6:05PM CDT	331	3	8	WEB Regular Price	$38.00
	331	3	9	WEB Regular Price	$38.00

The convenience fee is $4.00 per ticket.
Total Convenience Fee for 2 seats : $8.00
Total Price for 2 seats: $84.00

Order Items	Section	Row/Box	Seat	Type	Price
2008 Game Day Parking-August 23 Saturday, 8/23/08 6:05PM CDT	GENRAL	GEN1	122	Web Parking	$8.00

The convenience fee is $2.00 per ticket.
Total Convenience Fee for 1 seat : $2.00
Total Price for 1 seat: $10.00

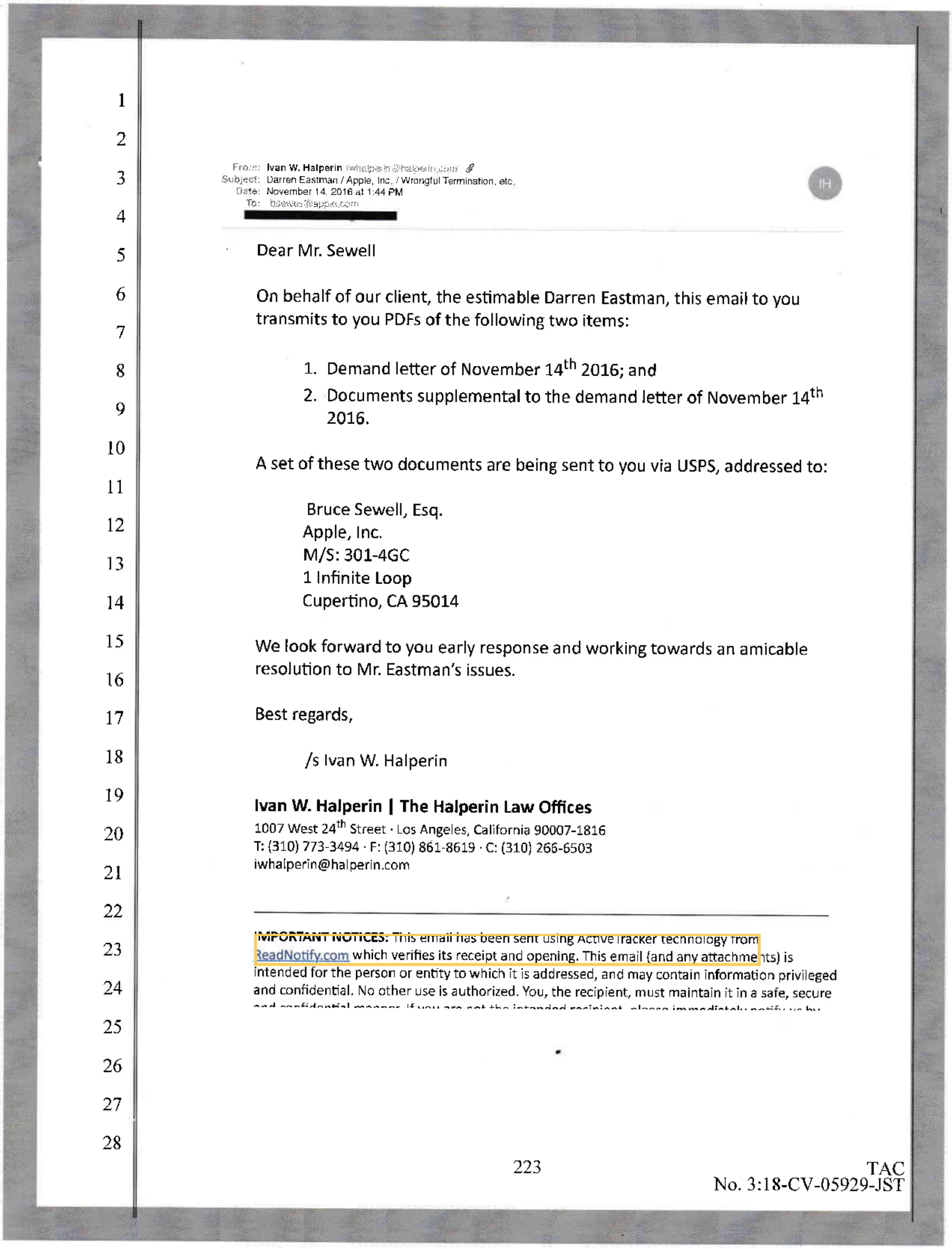
SUMMARY

Transaction Subtotal:	$94.00
Order Processing (including delivery):	$3.00
Total:	**$97.00**

Interested in More Games? Click here for the team's schedule page

Click on the links below for more information:

EXHIBIT 23



From: Ivan W. Halperin iwhalperin@halperin.com
Subject: Darren Eastman / Apple, Inc. / Wrongful Termination, etc.
Date: November 14, 2016 at 1:44 PM
To: bsewell@apple.com
[redacted]

Dear Mr. Sewell

On behalf of our client, the estimable Darren Eastman, this email to you transmits to you PDFs of the following two items:

1. Demand letter of November 14th 2016; and
2. Documents supplemental to the demand letter of November 14th 2016.

A set of these two documents are being sent to you via USPS, addressed to:

Bruce Sewell, Esq.
Apple, Inc.
M/S: 301-4GC
1 Infinite Loop
Cupertino, CA 95014

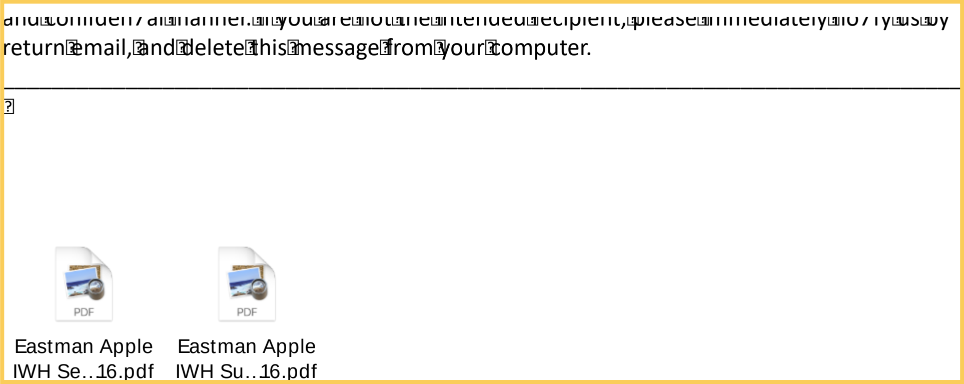
We look forward to you early response and working towards an amicable resolution to Mr. Eastman's issues.

Best regards,

/s Ivan W. Halperin

Ivan W. Halperin | The Halperin Law Offices
1007 West 24th Street · Los Angeles, California 90007-1816
T: (310) 773-3494 · F: (310) 861-8619 · C: (310) 266-6503
iwhalperin@halperin.com

IMPORTANT NOTICES: This email has been sent using Active Tracker technology from ReadNotify.com which verifies its receipt and opening. This email (and any attachments) is intended for the person or entity to which it is addressed, and may contain information privileged and confidential. No other use is authorized. You, the recipient, must maintain it in a safe, secure and confidential manner. If you are not the intended recipient, please immediately notify us by

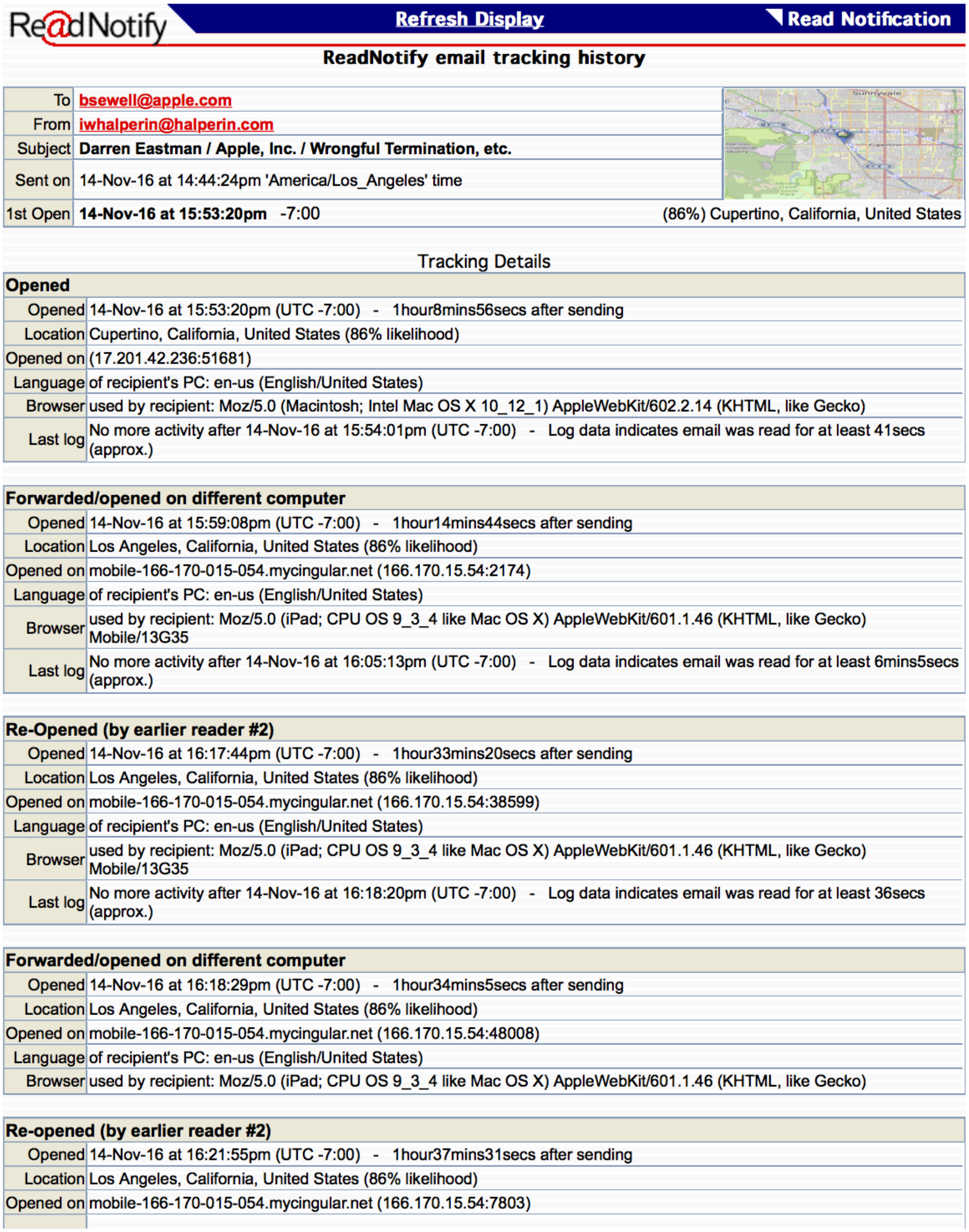


and confiden7al manner. If you are not the intended recipient, please immediately no7fy us by return email, and delete this message from your computer.

Eastman Apple IWH Se...16.pdf

Eastman Apple IWH Su...16.pdf

EXHIBIT 24



ReadNotify email tracking history

To	bsewell@apple.com
From	iwhalperin@halperin.com
Subject	**Darren Eastman / Apple, Inc. / Wrongful Termination, etc.**
Sent on	14-Nov-16 at 14:44:24pm 'America/Los_Angeles' time
1st Open	**14-Nov-16 at 15:53:20pm** -7:00 (86%) Cupertino, California, United States

Tracking Details

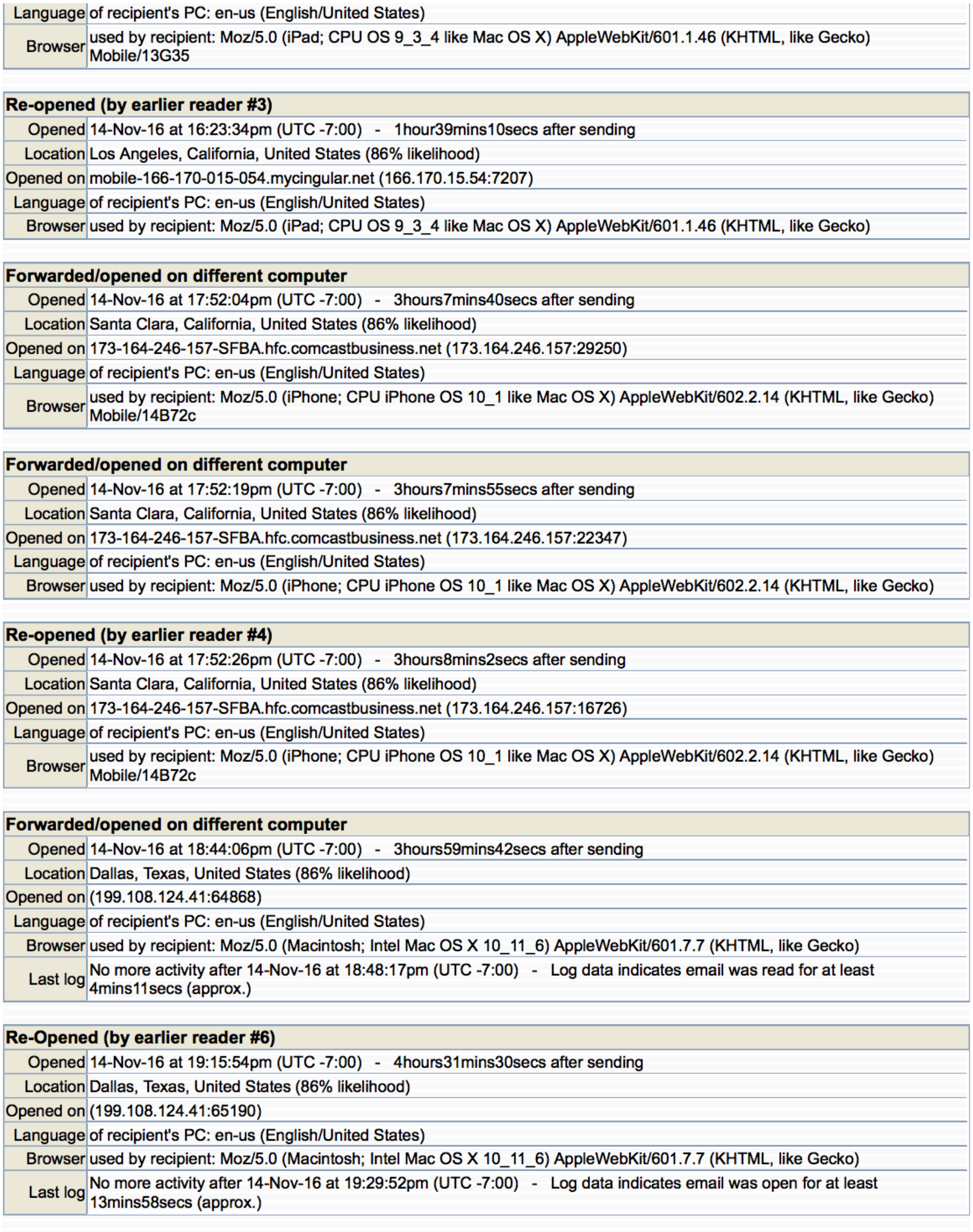
Opened	
Opened	14-Nov-16 at 15:53:20pm (UTC -7:00) - 1hour8mins56secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.201.42.236:51681)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 14-Nov-16 at 15:54:01pm (UTC -7:00) - Log data indicates email was read for at least 41secs (approx.)

Forwarded/opened on different computer	
Opened	14-Nov-16 at 15:59:08pm (UTC -7:00) - 1hour14mins44secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-015-054.mycingular.net (166.170.15.54:2174)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35
Last log	No more activity after 14-Nov-16 at 16:05:13pm (UTC -7:00) - Log data indicates email was read for at least 6mins5secs (approx.)

Re-Opened (by earlier reader #2)	
Opened	14-Nov-16 at 16:17:44pm (UTC -7:00) - 1hour33mins20secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-015-054.mycingular.net (166.170.15.54:38599)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35
Last log	No more activity after 14-Nov-16 at 16:18:20pm (UTC -7:00) - Log data indicates email was read for at least 36secs (approx.)

Forwarded/opened on different computer	
Opened	14-Nov-16 at 16:18:29pm (UTC -7:00) - 1hour34mins5secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-015-054.mycingular.net (166.170.15.54:48008)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)

Re-opened (by earlier reader #2)	
Opened	14-Nov-16 at 16:21:55pm (UTC -7:00) - 1hour37mins31secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-015-054.mycingular.net (166.170.15.54:7803)



Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-opened (by earlier reader #3)	
Opened	14-Nov-16 at 16:23:34pm (UTC -7:00) - 1hour39mins10secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-015-054.mycingular.net (166.170.15.54:7207)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)

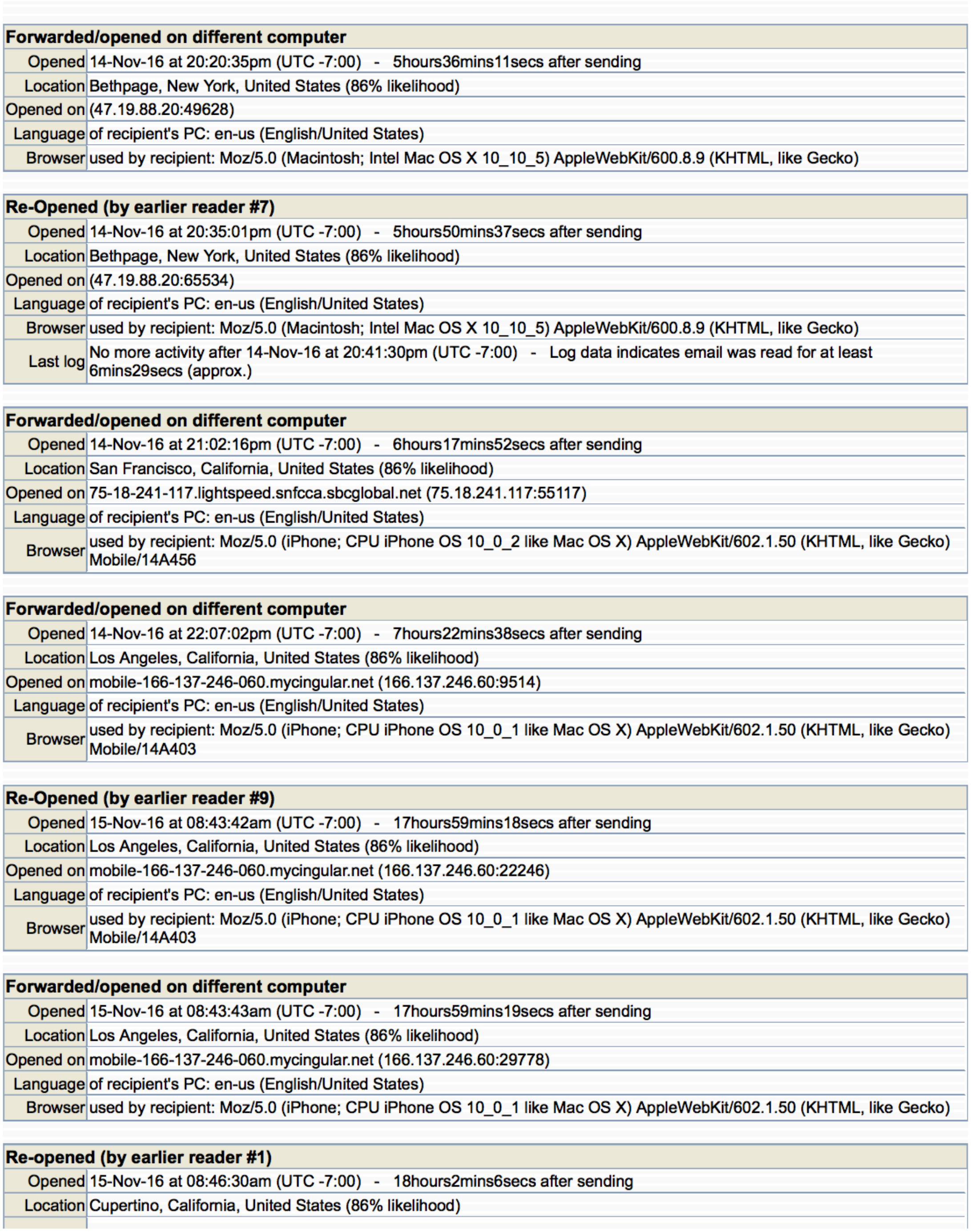
Forwarded/opened on different computer	
Opened	14-Nov-16 at 17:52:04pm (UTC -7:00) - 3hours7mins40secs after sending
Location	Santa Clara, California, United States (86% likelihood)
Opened on	173-164-246-157-SFBA.hfc.comcastbusiness.net (173.164.246.157:29250)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B72c

Forwarded/opened on different computer	
Opened	14-Nov-16 at 17:52:19pm (UTC -7:00) - 3hours7mins55secs after sending
Location	Santa Clara, California, United States (86% likelihood)
Opened on	173-164-246-157-SFBA.hfc.comcastbusiness.net (173.164.246.157:22347)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #4)	
Opened	14-Nov-16 at 17:52:26pm (UTC -7:00) - 3hours8mins2secs after sending
Location	Santa Clara, California, United States (86% likelihood)
Opened on	173-164-246-157-SFBA.hfc.comcastbusiness.net (173.164.246.157:16726)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B72c

Forwarded/opened on different computer	
Opened	14-Nov-16 at 18:44:06pm (UTC -7:00) - 3hours59mins42secs after sending
Location	Dallas, Texas, United States (86% likelihood)
Opened on	(199.108.124.41:64868)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/601.7.7 (KHTML, like Gecko)
Last log	No more activity after 14-Nov-16 at 18:48:17pm (UTC -7:00) - Log data indicates email was read for at least 4mins11secs (approx.)

Re-Opened (by earlier reader #6)	
Opened	14-Nov-16 at 19:15:54pm (UTC -7:00) - 4hours31mins30secs after sending
Location	Dallas, Texas, United States (86% likelihood)
Opened on	(199.108.124.41:65190)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/601.7.7 (KHTML, like Gecko)
Last log	No more activity after 14-Nov-16 at 19:29:52pm (UTC -7:00) - Log data indicates email was open for at least 13mins58secs (approx.)



Forwarded/opened on different computer	
Opened	14-Nov-16 at 20:20:35pm (UTC -7:00) - 5hours36mins11secs after sending
Location	Bethpage, New York, United States (86% likelihood)
Opened on	(47.19.88.20:49628)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/600.8.9 (KHTML, like Gecko)

Re-Opened (by earlier reader #7)	
Opened	14-Nov-16 at 20:35:01pm (UTC -7:00) - 5hours50mins37secs after sending
Location	Bethpage, New York, United States (86% likelihood)
Opened on	(47.19.88.20:65534)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_10_5) AppleWebKit/600.8.9 (KHTML, like Gecko)
Last log	No more activity after 14-Nov-16 at 20:41:30pm (UTC -7:00) - Log data indicates email was read for at least 6mins29secs (approx.)

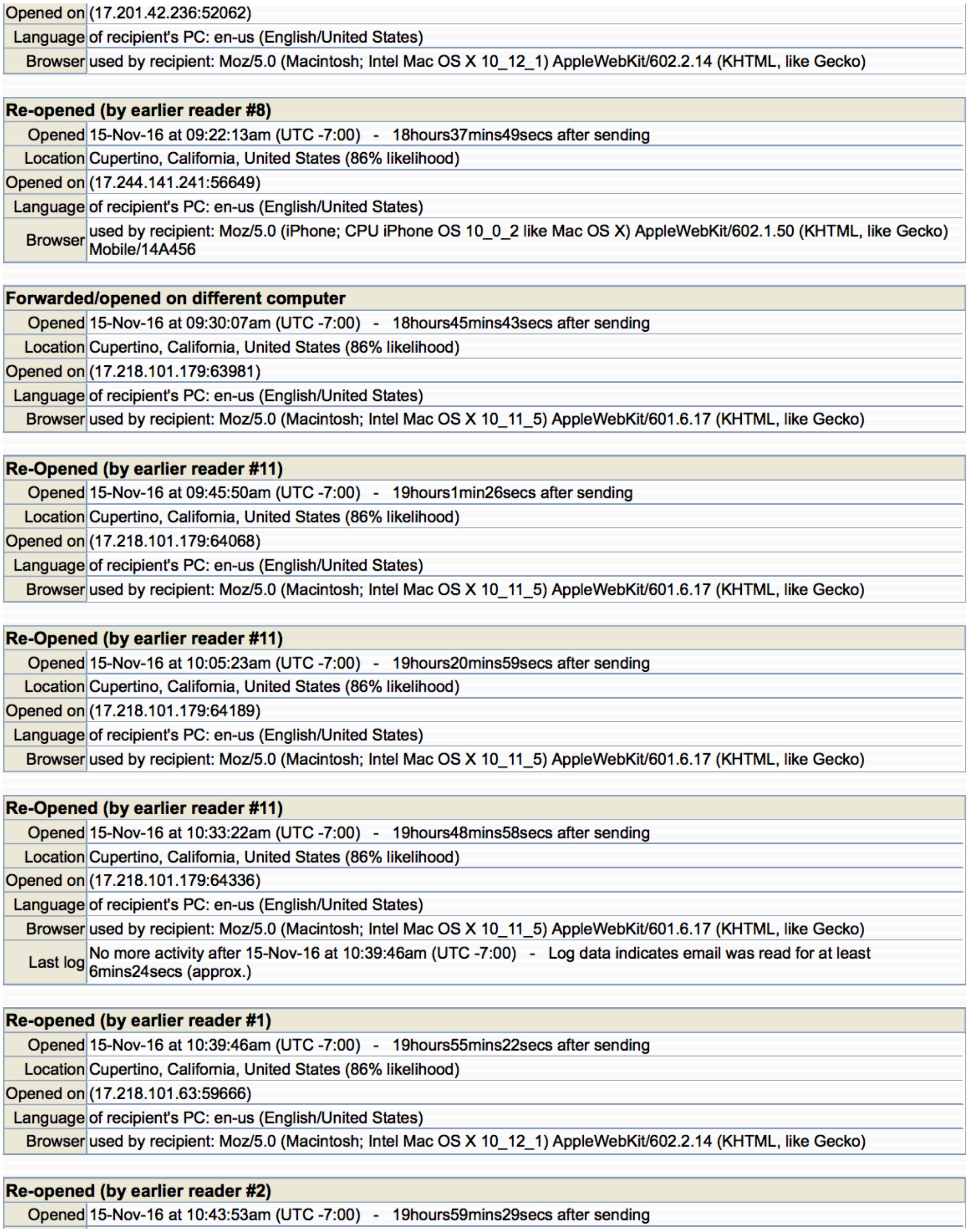
Forwarded/opened on different computer	
Opened	14-Nov-16 at 21:02:16pm (UTC -7:00) - 6hours17mins52secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	75-18-241-117.lightspeed.snfcca.sbcglobal.net (75.18.241.117:55117)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_0_2 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko) Mobile/14A456

Forwarded/opened on different computer	
Opened	14-Nov-16 at 22:07:02pm (UTC -7:00) - 7hours22mins38secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-137-246-060.mycingular.net (166.137.246.60:9514)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_0_1 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko) Mobile/14A403

Re-Opened (by earlier reader #9)	
Opened	15-Nov-16 at 08:43:42am (UTC -7:00) - 17hours59mins18secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-137-246-060.mycingular.net (166.137.246.60:22246)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_0_1 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko) Mobile/14A403

Forwarded/opened on different computer	
Opened	15-Nov-16 at 08:43:43am (UTC -7:00) - 17hours59mins19secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-137-246-060.mycingular.net (166.137.246.60:29778)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_0_1 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko)

Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 08:46:30am (UTC -7:00) - 18hours2mins6secs after sending
Location	Cupertino, California, United States (86% likelihood)



Opened on	(17.201.42.236:52062)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #8)	
Opened	15-Nov-16 at 09:22:13am (UTC -7:00) - 18hours37mins49secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.244.141.241:56649)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_0_2 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko) Mobile/14A456

Forwarded/opened on different computer	
Opened	15-Nov-16 at 09:30:07am (UTC -7:00) - 18hours45mins43secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.179:63981)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko)

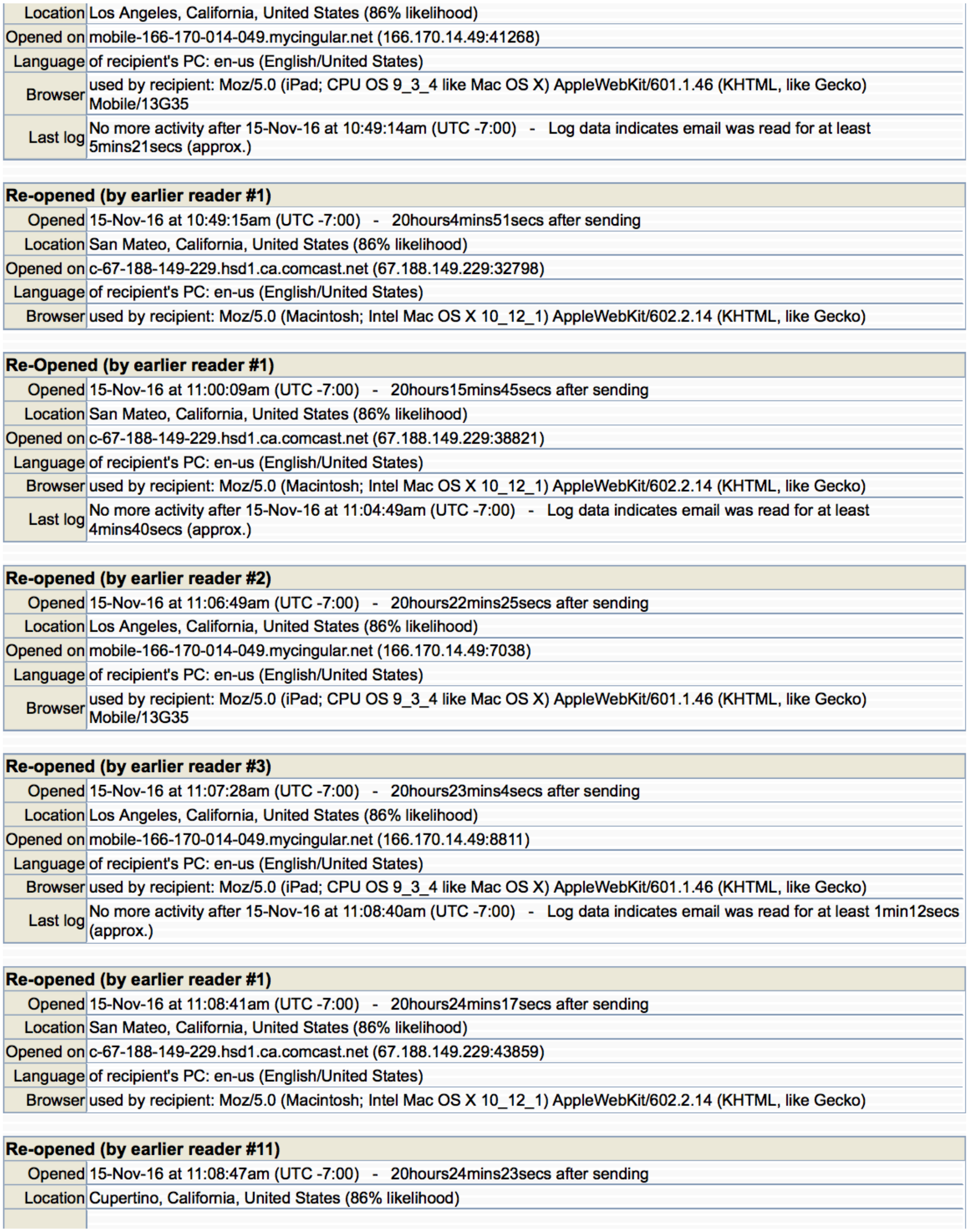
Re-Opened (by earlier reader #11)	
Opened	15-Nov-16 at 09:45:50am (UTC -7:00) - 19hours1min26secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.179:64068)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko)

Re-Opened (by earlier reader #11)	
Opened	15-Nov-16 at 10:05:23am (UTC -7:00) - 19hours20mins59secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.179:64189)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko)

Re-Opened (by earlier reader #11)	
Opened	15-Nov-16 at 10:33:22am (UTC -7:00) - 19hours48mins58secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.179:64336)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 10:39:46am (UTC -7:00) - Log data indicates email was read for at least 6mins24secs (approx.)

Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 10:39:46am (UTC -7:00) - 19hours55mins22secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.63:59666)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #2)	
Opened	15-Nov-16 at 10:43:53am (UTC -7:00) - 19hours59mins29secs after sending



Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:41268)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35
Last log	No more activity after 15-Nov-16 at 10:49:14am (UTC -7:00) - Log data indicates email was read for at least 5mins21secs (approx.)

Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 10:49:15am (UTC -7:00) - 20hours4mins51secs after sending
Location	San Mateo, California, United States (86% likelihood)
Opened on	c-67-188-149-229.hsd1.ca.comcast.net (67.188.149.229:32798)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

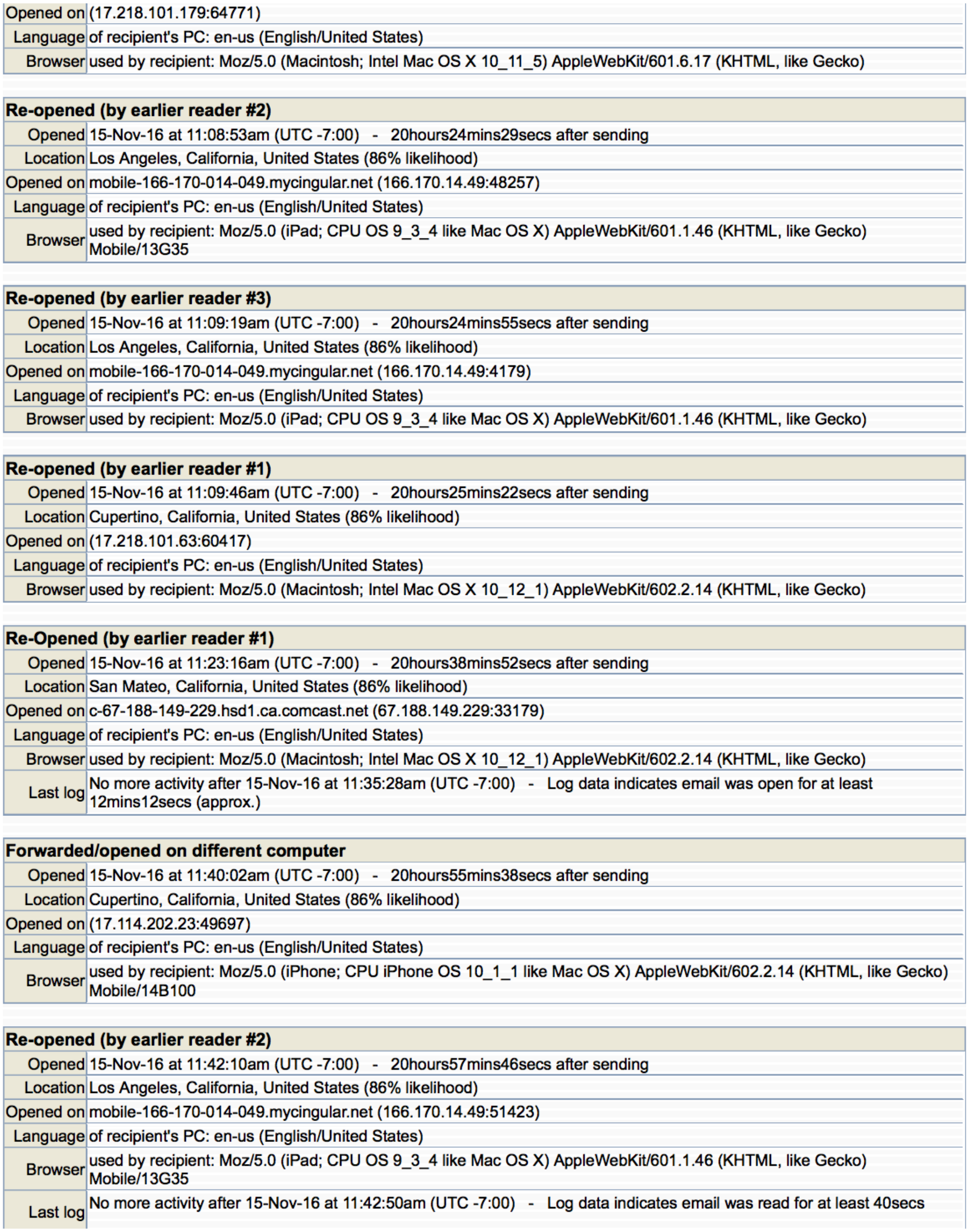
Re-Opened (by earlier reader #1)	
Opened	15-Nov-16 at 11:00:09am (UTC -7:00) - 20hours15mins45secs after sending
Location	San Mateo, California, United States (86% likelihood)
Opened on	c-67-188-149-229.hsd1.ca.comcast.net (67.188.149.229:38821)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 11:04:49am (UTC -7:00) - Log data indicates email was read for at least 4mins40secs (approx.)

Re-opened (by earlier reader #2)	
Opened	15-Nov-16 at 11:06:49am (UTC -7:00) - 20hours22mins25secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:7038)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-opened (by earlier reader #3)	
Opened	15-Nov-16 at 11:07:28am (UTC -7:00) - 20hours23mins4secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:8811)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 11:08:40am (UTC -7:00) - Log data indicates email was read for at least 1min12secs (approx.)

Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 11:08:41am (UTC -7:00) - 20hours24mins17secs after sending
Location	San Mateo, California, United States (86% likelihood)
Opened on	c-67-188-149-229.hsd1.ca.comcast.net (67.188.149.229:43859)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #11)	
Opened	15-Nov-16 at 11:08:47am (UTC -7:00) - 20hours24mins23secs after sending
Location	Cupertino, California, United States (86% likelihood)



Opened on	(17.218.101.179:64771)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_5) AppleWebKit/601.6.17 (KHTML, like Gecko)

Re-opened (by earlier reader #2)	
Opened	15-Nov-16 at 11:08:53am (UTC -7:00) - 20hours24mins29secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:48257)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

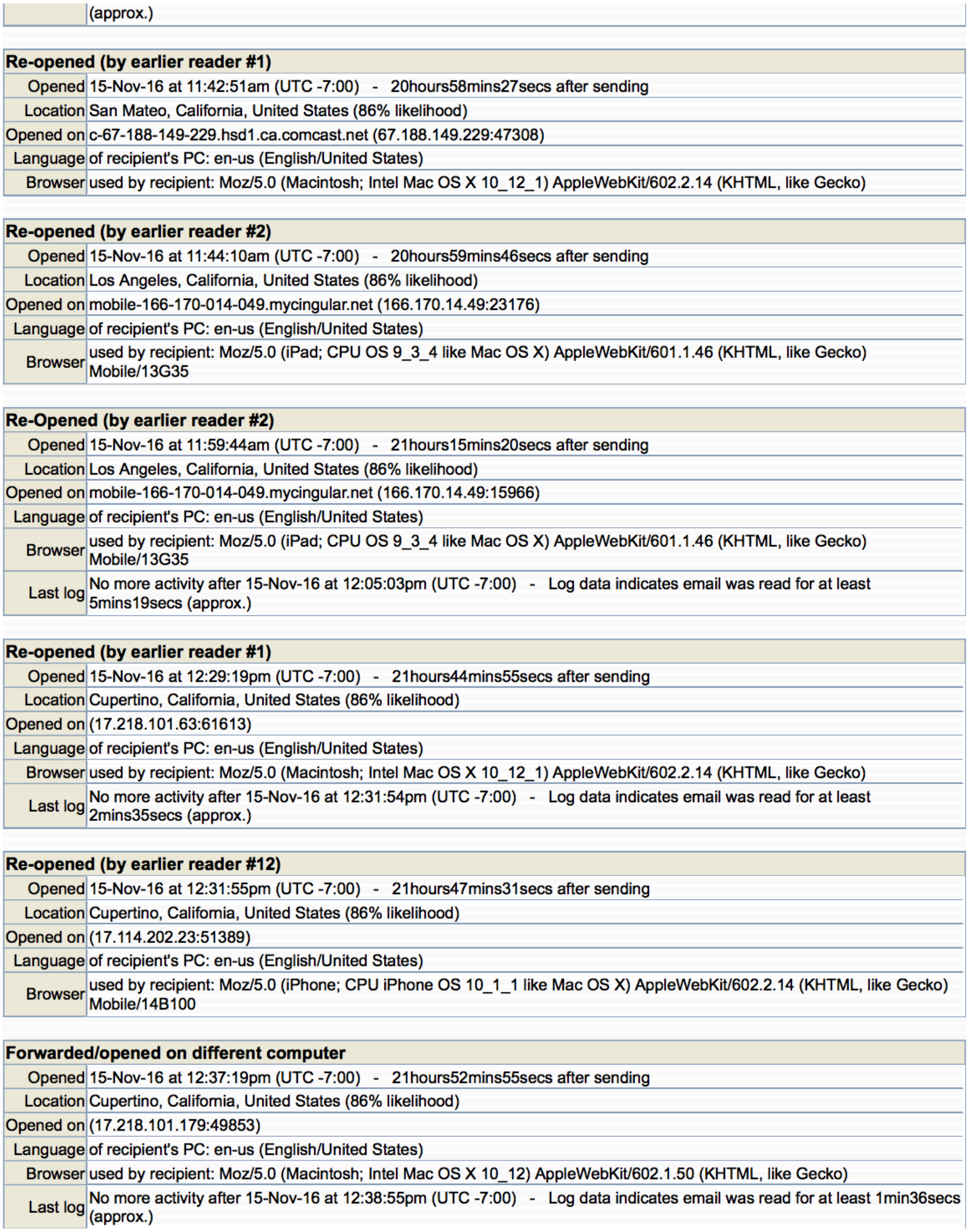
Re-opened (by earlier reader #3)	
Opened	15-Nov-16 at 11:09:19am (UTC -7:00) - 20hours24mins55secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:4179)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)

Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 11:09:46am (UTC -7:00) - 20hours25mins22secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.63:60417)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-Opened (by earlier reader #1)	
Opened	15-Nov-16 at 11:23:16am (UTC -7:00) - 20hours38mins52secs after sending
Location	San Mateo, California, United States (86% likelihood)
Opened on	c-67-188-149-229.hsd1.ca.comcast.net (67.188.149.229:33179)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 11:35:28am (UTC -7:00) - Log data indicates email was open for at least 12mins12secs (approx.)

Forwarded/opened on different computer	
Opened	15-Nov-16 at 11:40:02am (UTC -7:00) - 20hours55mins38secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.114.202.23:49697)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Re-opened (by earlier reader #2)	
Opened	15-Nov-16 at 11:42:10am (UTC -7:00) - 20hours57mins46secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:51423)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35
Last log	No more activity after 15-Nov-16 at 11:42:50am (UTC -7:00) - Log data indicates email was read for at least 40secs



	(approx.)

Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 11:42:51am (UTC -7:00) - 20hours58mins27secs after sending
Location	San Mateo, California, United States (86% likelihood)
Opened on	c-67-188-149-229.hsd1.ca.comcast.net (67.188.149.229:47308)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

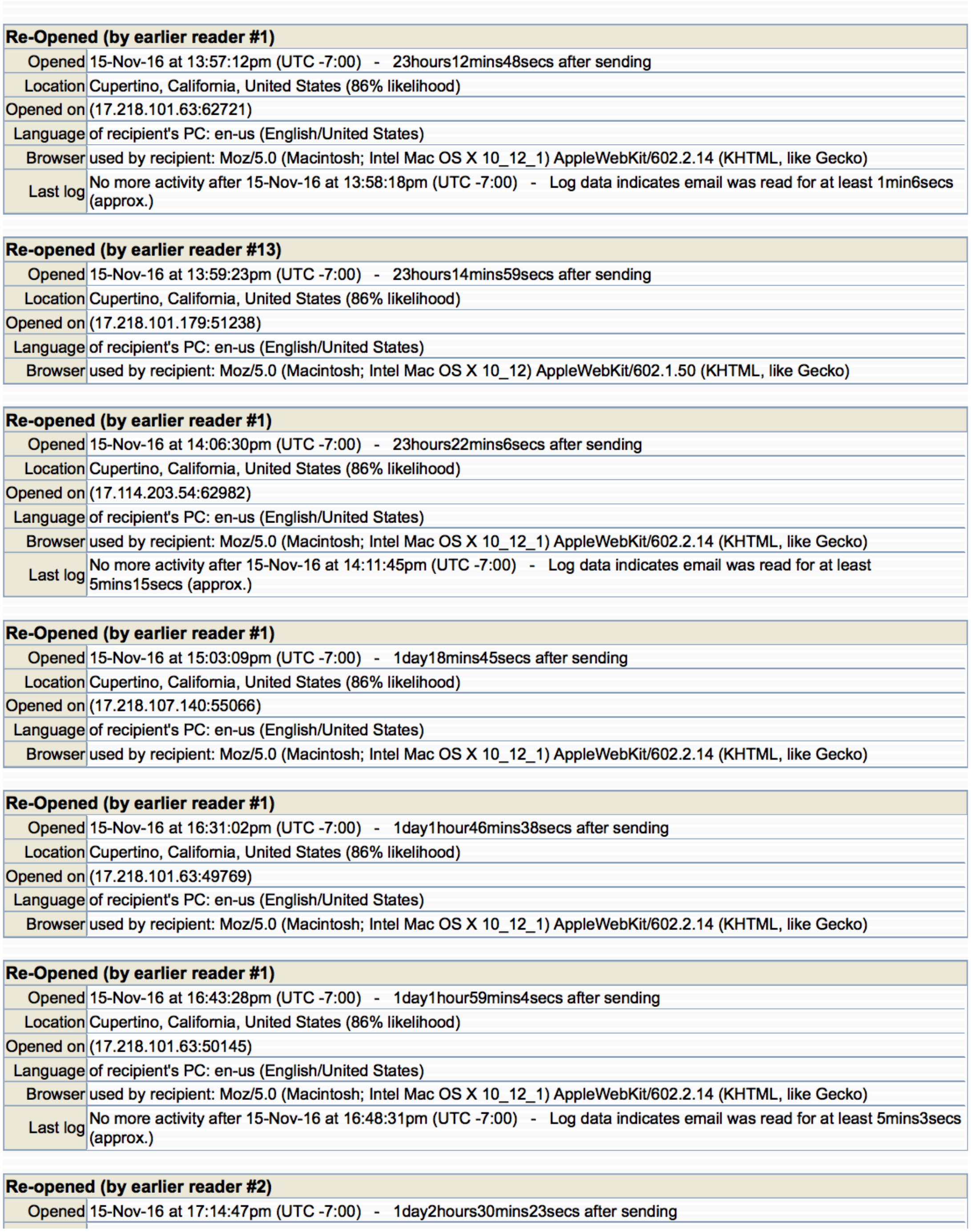
Re-opened (by earlier reader #2)	
Opened	15-Nov-16 at 11:44:10am (UTC -7:00) - 20hours59mins46secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:23176)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-Opened (by earlier reader #2)	
Opened	15-Nov-16 at 11:59:44am (UTC -7:00) - 21hours15mins20secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:15966)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35
Last log	No more activity after 15-Nov-16 at 12:05:03pm (UTC -7:00) - Log data indicates email was read for at least 5mins19secs (approx.)

Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 12:29:19pm (UTC -7:00) - 21hours44mins55secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.63:61613)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 12:31:54pm (UTC -7:00) - Log data indicates email was read for at least 2mins35secs (approx.)

Re-opened (by earlier reader #12)	
Opened	15-Nov-16 at 12:31:55pm (UTC -7:00) - 21hours47mins31secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.114.202.23:51389)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Forwarded/opened on different computer	
Opened	15-Nov-16 at 12:37:19pm (UTC -7:00) - 21hours52mins55secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.179:49853)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12) AppleWebKit/602.1.50 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 12:38:55pm (UTC -7:00) - Log data indicates email was read for at least 1min36secs (approx.)



Re-Opened (by earlier reader #1)	
Opened	15-Nov-16 at 13:57:12pm (UTC -7:00) - 23hours12mins48secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.63:62721)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 13:58:18pm (UTC -7:00) - Log data indicates email was read for at least 1min6secs (approx.)

Re-opened (by earlier reader #13)	
Opened	15-Nov-16 at 13:59:23pm (UTC -7:00) - 23hours14mins59secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.179:51238)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12) AppleWebKit/602.1.50 (KHTML, like Gecko)

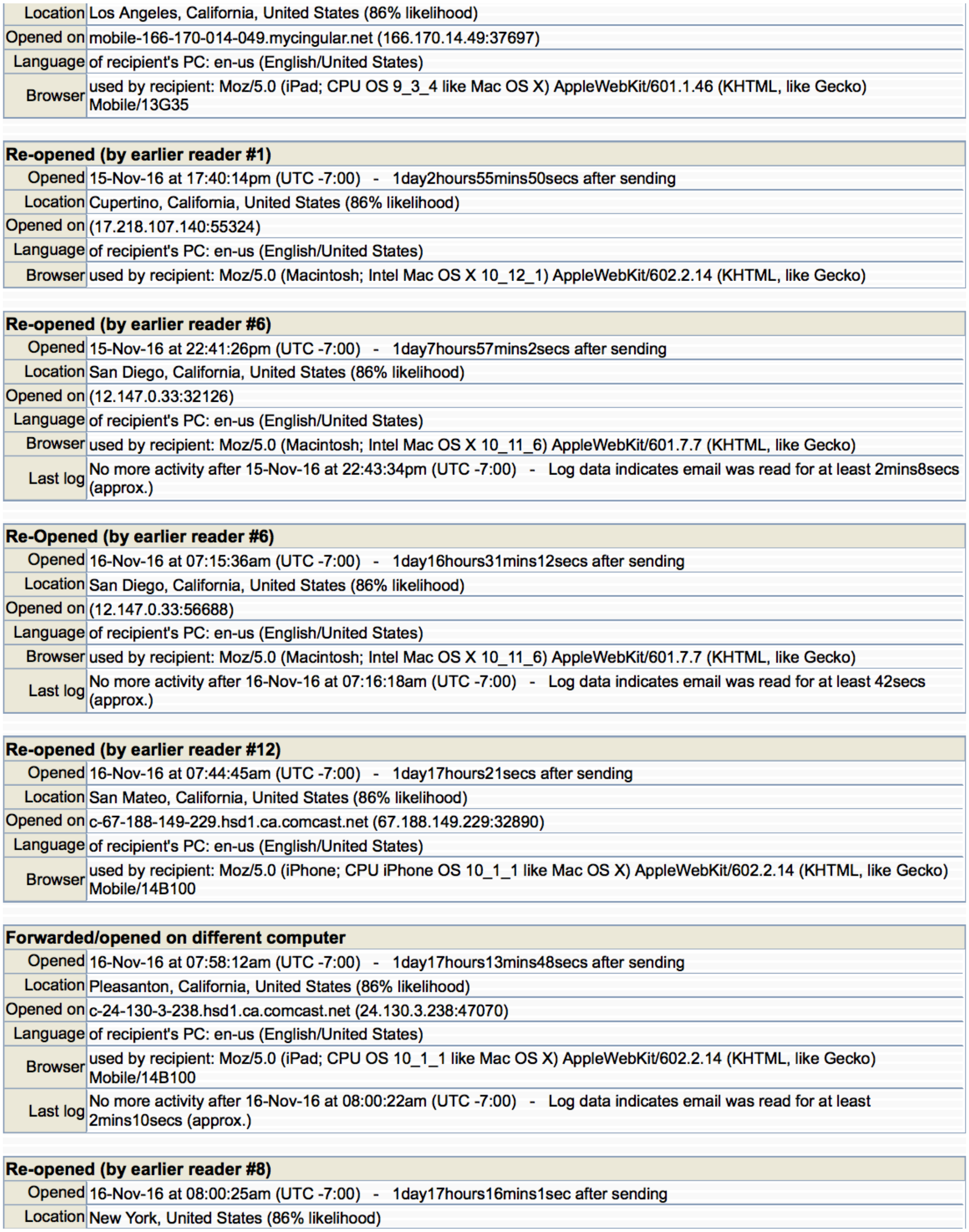
Re-opened (by earlier reader #1)	
Opened	15-Nov-16 at 14:06:30pm (UTC -7:00) - 23hours22mins6secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.114.203.54:62982)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 14:11:45pm (UTC -7:00) - Log data indicates email was read for at least 5mins15secs (approx.)

Re-Opened (by earlier reader #1)	
Opened	15-Nov-16 at 15:03:09pm (UTC -7:00) - 1day18mins45secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.107.140:55066)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-Opened (by earlier reader #1)	
Opened	15-Nov-16 at 16:31:02pm (UTC -7:00) - 1day1hour46mins38secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.63:49769)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-Opened (by earlier reader #1)	
Opened	15-Nov-16 at 16:43:28pm (UTC -7:00) - 1day1hour59mins4secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.63:50145)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 16:48:31pm (UTC -7:00) - Log data indicates email was read for at least 5mins3secs (approx.)

Re-opened (by earlier reader #2)	
Opened	15-Nov-16 at 17:14:47pm (UTC -7:00) - 1day2hours30mins23secs after sending



Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:37697)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-opened (by earlier reader #1)

Opened	15-Nov-16 at 17:40:14pm (UTC -7:00) - 1day2hours55mins50secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.107.140:55324)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #6)

Opened	15-Nov-16 at 22:41:26pm (UTC -7:00) - 1day7hours57mins2secs after sending
Location	San Diego, California, United States (86% likelihood)
Opened on	(12.147.0.33:32126)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/601.7.7 (KHTML, like Gecko)
Last log	No more activity after 15-Nov-16 at 22:43:34pm (UTC -7:00) - Log data indicates email was read for at least 2mins8secs (approx.)

Re-Opened (by earlier reader #6)

Opened	16-Nov-16 at 07:15:36am (UTC -7:00) - 1day16hours31mins12secs after sending
Location	San Diego, California, United States (86% likelihood)
Opened on	(12.147.0.33:56688)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/601.7.7 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 07:16:18am (UTC -7:00) - Log data indicates email was read for at least 42secs (approx.)

Re-opened (by earlier reader #12)

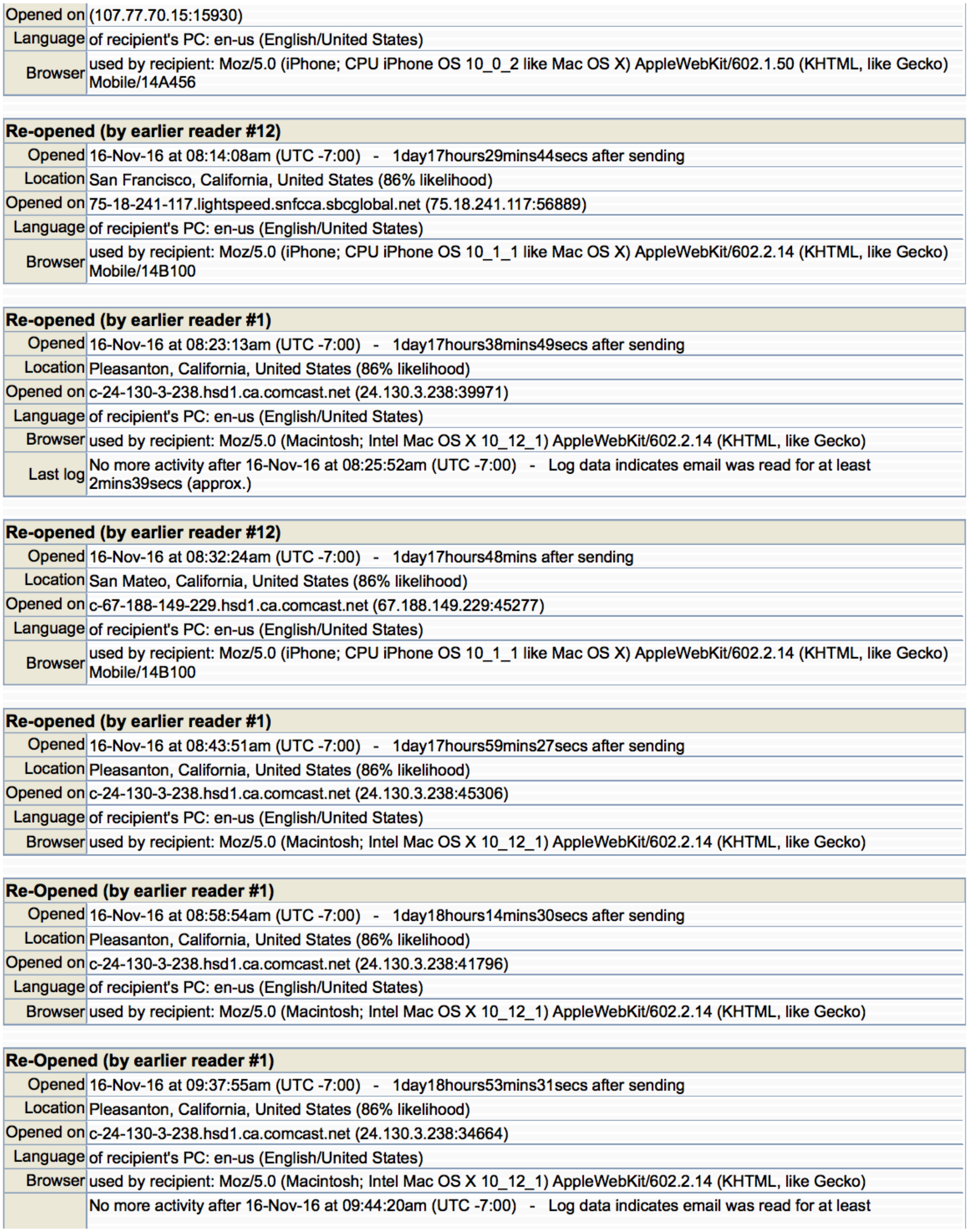
Opened	16-Nov-16 at 07:44:45am (UTC -7:00) - 1day17hours21secs after sending
Location	San Mateo, California, United States (86% likelihood)
Opened on	c-67-188-149-229.hsd1.ca.comcast.net (67.188.149.229:32890)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Forwarded/opened on different computer

Opened	16-Nov-16 at 07:58:12am (UTC -7:00) - 1day17hours13mins48secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:47070)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100
Last log	No more activity after 16-Nov-16 at 08:00:22am (UTC -7:00) - Log data indicates email was read for at least 2mins10secs (approx.)

Re-opened (by earlier reader #8)

Opened	16-Nov-16 at 08:00:25am (UTC -7:00) - 1day17hours16mins1sec after sending
Location	New York, United States (86% likelihood)



Opened on	(107.77.70.15:15930)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_0_2 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko) Mobile/14A456

Re-opened (by earlier reader #12)	
Opened	16-Nov-16 at 08:14:08am (UTC -7:00) - 1day17hours29mins44secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	75-18-241-117.lightspeed.snfcca.sbcglobal.net (75.18.241.117:56889)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

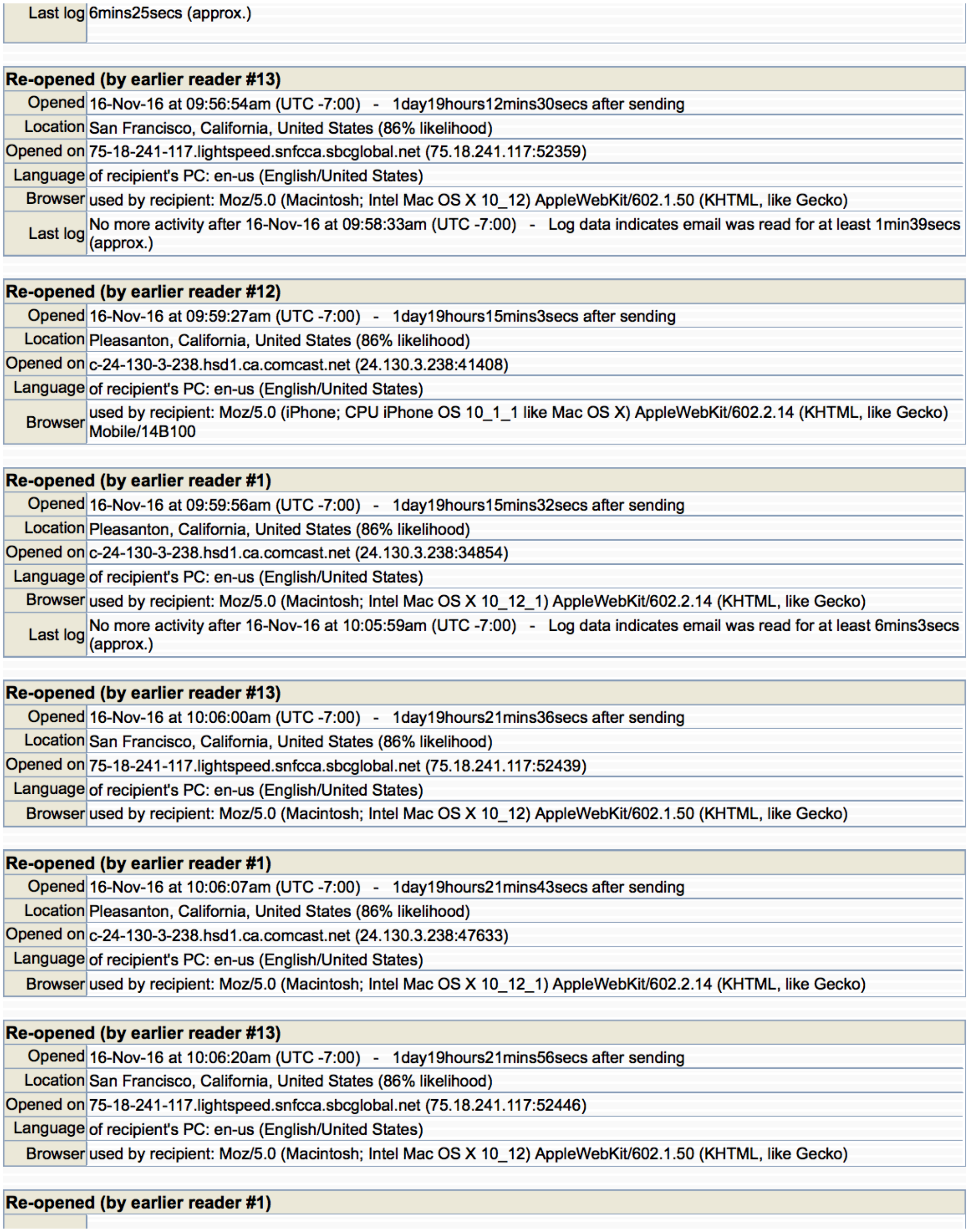
Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 08:23:13am (UTC -7:00) - 1day17hours38mins49secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:39971)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 08:25:52am (UTC -7:00) - Log data indicates email was read for at least 2mins39secs (approx.)

Re-opened (by earlier reader #12)	
Opened	16-Nov-16 at 08:32:24am (UTC -7:00) - 1day17hours48mins after sending
Location	San Mateo, California, United States (86% likelihood)
Opened on	c-67-188-149-229.hsd1.ca.comcast.net (67.188.149.229:45277)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 08:43:51am (UTC -7:00) - 1day17hours59mins27secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:45306)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-Opened (by earlier reader #1)	
Opened	16-Nov-16 at 08:58:54am (UTC -7:00) - 1day18hours14mins30secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:41796)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-Opened (by earlier reader #1)	
Opened	16-Nov-16 at 09:37:55am (UTC -7:00) - 1day18hours53mins31secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:34664)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
	No more activity after 16-Nov-16 at 09:44:20am (UTC -7:00) - Log data indicates email was read for at least



Last log	6mins25secs (approx.)

Re-opened (by earlier reader #13)	
Opened	16-Nov-16 at 09:56:54am (UTC -7:00) - 1day19hours12mins30secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	75-18-241-117.lightspeed.snfcca.sbcglobal.net (75.18.241.117:52359)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12) AppleWebKit/602.1.50 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 09:58:33am (UTC -7:00) - Log data indicates email was read for at least 1min39secs (approx.)

Re-opened (by earlier reader #12)	
Opened	16-Nov-16 at 09:59:27am (UTC -7:00) - 1day19hours15mins3secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:41408)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

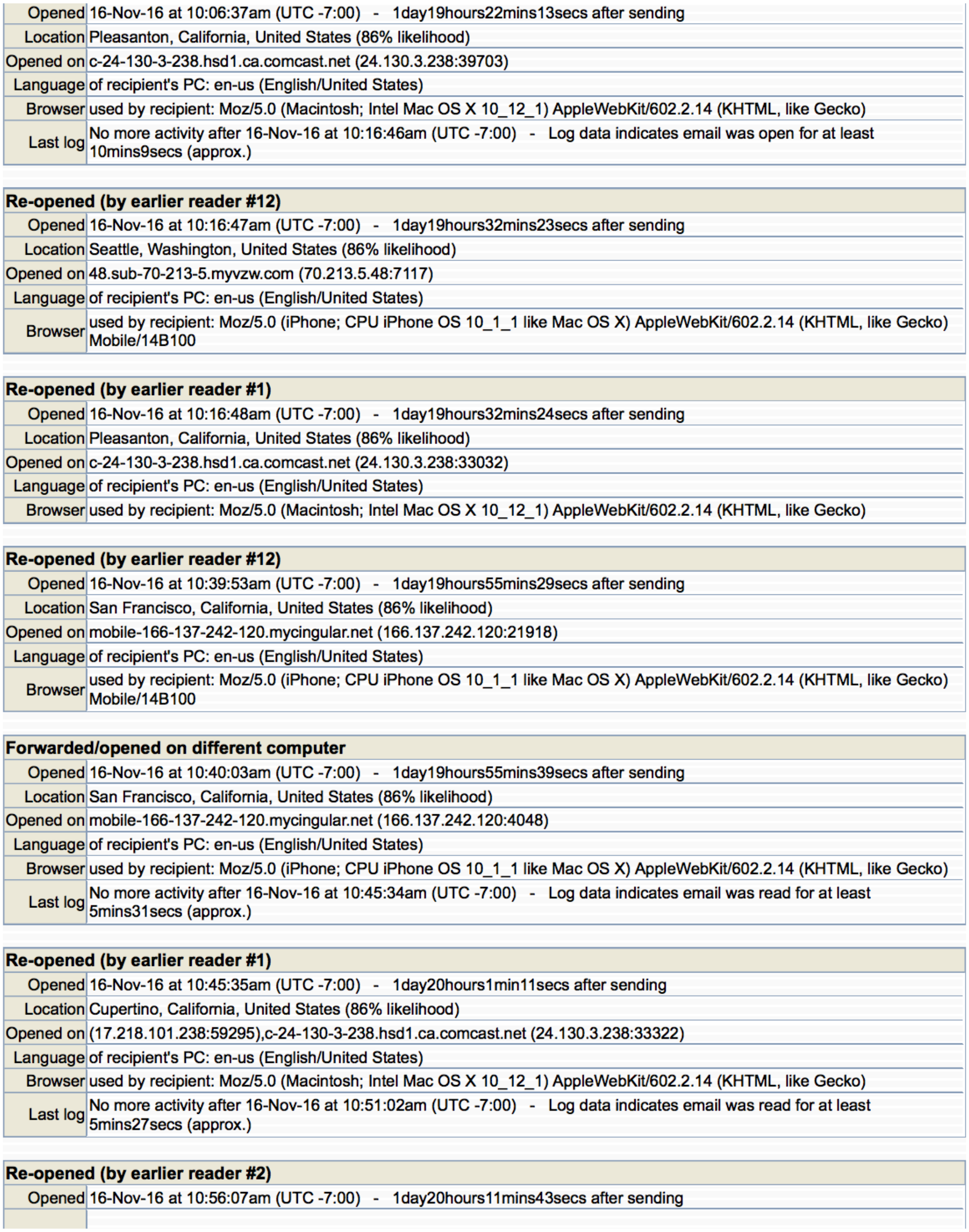
Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 09:59:56am (UTC -7:00) - 1day19hours15mins32secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:34854)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 10:05:59am (UTC -7:00) - Log data indicates email was read for at least 6mins3secs (approx.)

Re-opened (by earlier reader #13)	
Opened	16-Nov-16 at 10:06:00am (UTC -7:00) - 1day19hours21mins36secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	75-18-241-117.lightspeed.snfcca.sbcglobal.net (75.18.241.117:52439)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12) AppleWebKit/602.1.50 (KHTML, like Gecko)

Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 10:06:07am (UTC -7:00) - 1day19hours21mins43secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:47633)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #13)	
Opened	16-Nov-16 at 10:06:20am (UTC -7:00) - 1day19hours21mins56secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	75-18-241-117.lightspeed.snfcca.sbcglobal.net (75.18.241.117:52446)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12) AppleWebKit/602.1.50 (KHTML, like Gecko)

Re-opened (by earlier reader #1)	



Opened	16-Nov-16 at 10:06:37am (UTC -7:00) - 1day19hours22mins13secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:39703)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 10:16:46am (UTC -7:00) - Log data indicates email was open for at least 10mins9secs (approx.)

Re-opened (by earlier reader #12)	
Opened	16-Nov-16 at 10:16:47am (UTC -7:00) - 1day19hours32mins23secs after sending
Location	Seattle, Washington, United States (86% likelihood)
Opened on	48.sub-70-213-5.myvzw.com (70.213.5.48:7117)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

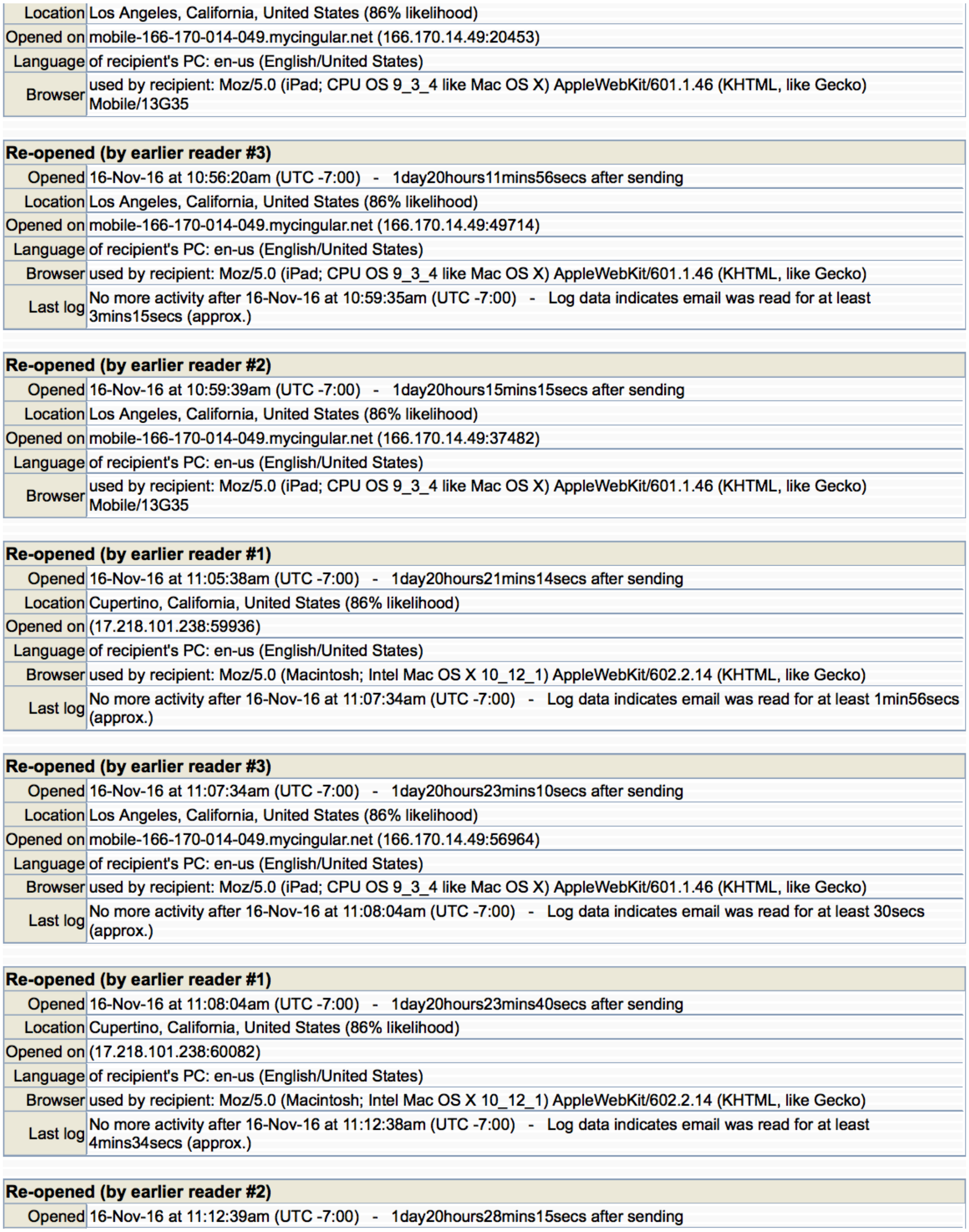
Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 10:16:48am (UTC -7:00) - 1day19hours32mins24secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:33032)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #12)	
Opened	16-Nov-16 at 10:39:53am (UTC -7:00) - 1day19hours55mins29secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	mobile-166-137-242-120.mycingular.net (166.137.242.120:21918)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Forwarded/opened on different computer	
Opened	16-Nov-16 at 10:40:03am (UTC -7:00) - 1day19hours55mins39secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	mobile-166-137-242-120.mycingular.net (166.137.242.120:4048)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 10:45:34am (UTC -7:00) - Log data indicates email was read for at least 5mins31secs (approx.)

Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 10:45:35am (UTC -7:00) - 1day20hours1min11secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.238:59295),c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:33322)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 10:51:02am (UTC -7:00) - Log data indicates email was read for at least 5mins27secs (approx.)

Re-opened (by earlier reader #2)	
Opened	16-Nov-16 at 10:56:07am (UTC -7:00) - 1day20hours11mins43secs after sending



Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:20453)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-opened (by earlier reader #3)	
Opened	16-Nov-16 at 10:56:20am (UTC -7:00) - 1day20hours11mins56secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:49714)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 10:59:35am (UTC -7:00) - Log data indicates email was read for at least 3mins15secs (approx.)

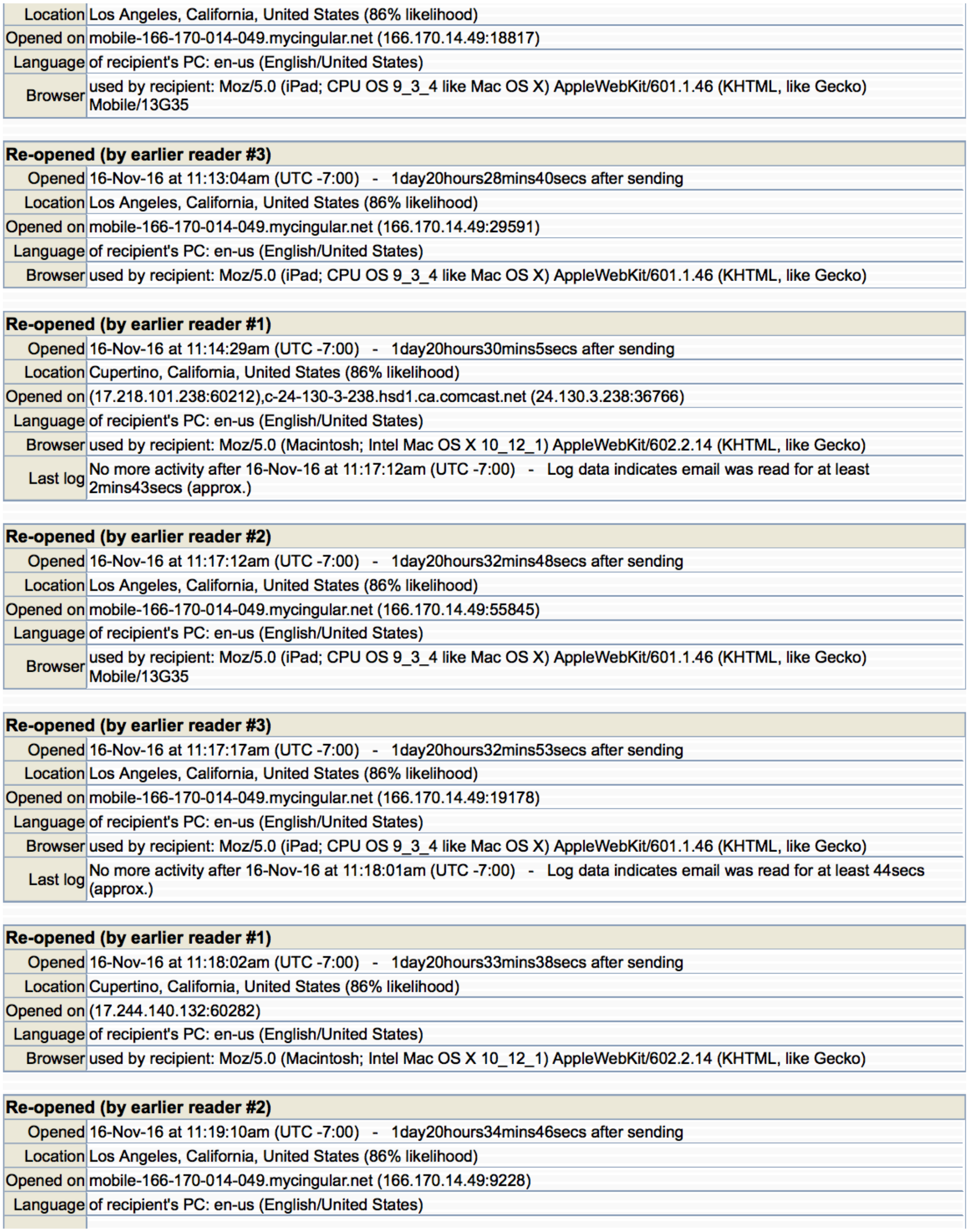
Re-opened (by earlier reader #2)	
Opened	16-Nov-16 at 10:59:39am (UTC -7:00) - 1day20hours15mins15secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:37482)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 11:05:38am (UTC -7:00) - 1day20hours21mins14secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.238:59936)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 11:07:34am (UTC -7:00) - Log data indicates email was read for at least 1min56secs (approx.)

Re-opened (by earlier reader #3)	
Opened	16-Nov-16 at 11:07:34am (UTC -7:00) - 1day20hours23mins10secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:56964)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 11:08:04am (UTC -7:00) - Log data indicates email was read for at least 30secs (approx.)

Re-opened (by earlier reader #1)	
Opened	16-Nov-16 at 11:08:04am (UTC -7:00) - 1day20hours23mins40secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.238:60082)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 11:12:38am (UTC -7:00) - Log data indicates email was read for at least 4mins34secs (approx.)

Re-opened (by earlier reader #2)	
Opened	16-Nov-16 at 11:12:39am (UTC -7:00) - 1day20hours28mins15secs after sending



Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:18817)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-opened (by earlier reader #3)

Opened	16-Nov-16 at 11:13:04am (UTC -7:00) - 1day20hours28mins40secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:29591)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)

Re-opened (by earlier reader #1)

Opened	16-Nov-16 at 11:14:29am (UTC -7:00) - 1day20hours30mins5secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.238:60212),c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:36766)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 11:17:12am (UTC -7:00) - Log data indicates email was read for at least 2mins43secs (approx.)

Re-opened (by earlier reader #2)

Opened	16-Nov-16 at 11:17:12am (UTC -7:00) - 1day20hours32mins48secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:55845)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35

Re-opened (by earlier reader #3)

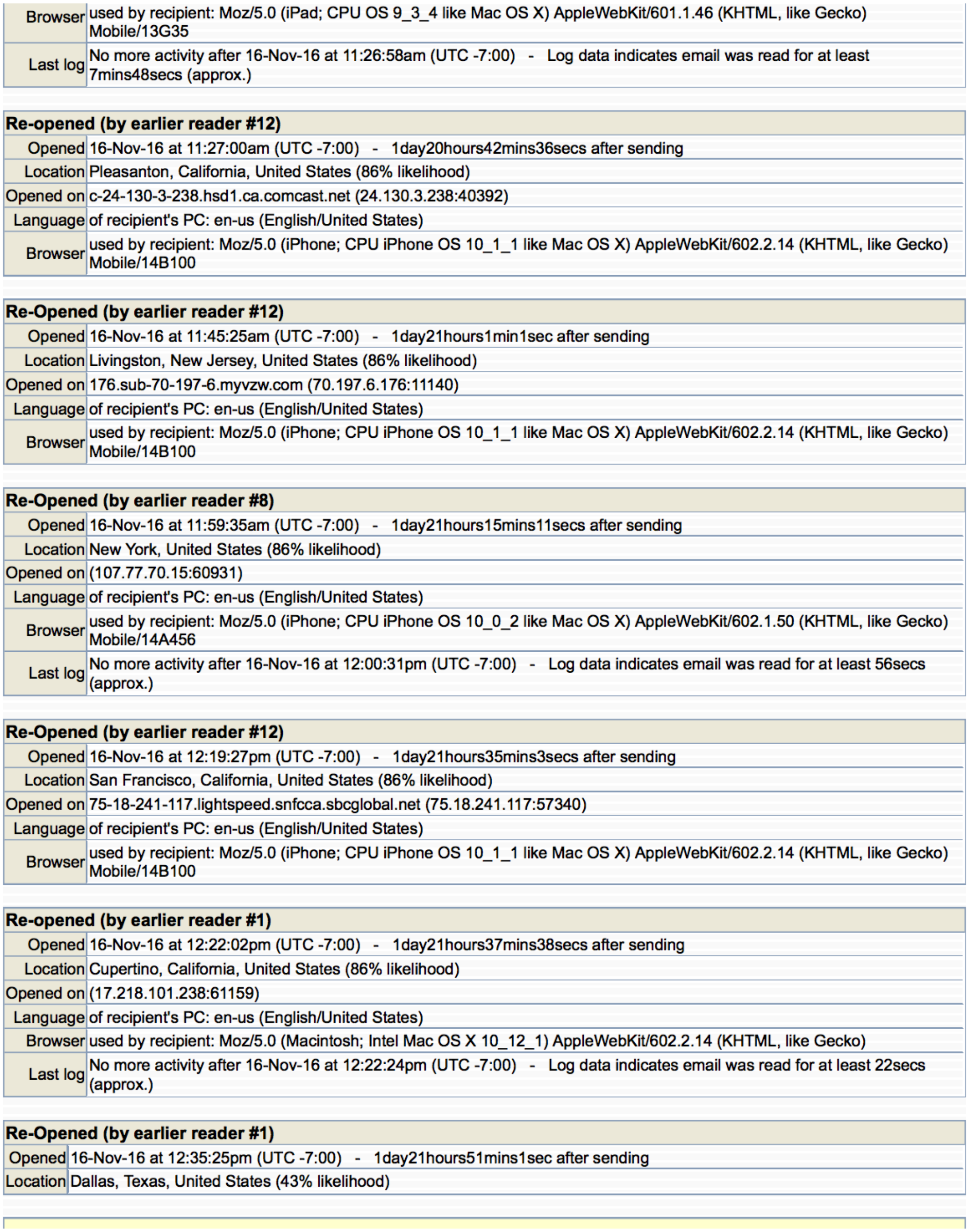
Opened	16-Nov-16 at 11:17:17am (UTC -7:00) - 1day20hours32mins53secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:19178)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 11:18:01am (UTC -7:00) - Log data indicates email was read for at least 44secs (approx.)

Re-opened (by earlier reader #1)

Opened	16-Nov-16 at 11:18:02am (UTC -7:00) - 1day20hours33mins38secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.244.140.132:60282)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)

Re-opened (by earlier reader #2)

Opened	16-Nov-16 at 11:19:10am (UTC -7:00) - 1day20hours34mins46secs after sending
Location	Los Angeles, California, United States (86% likelihood)
Opened on	mobile-166-170-014-049.mycingular.net (166.170.14.49:9228)
Language	of recipient's PC: en-us (English/United States)



Browser	used by recipient: Moz/5.0 (iPad; CPU OS 9_3_4 like Mac OS X) AppleWebKit/601.1.46 (KHTML, like Gecko) Mobile/13G35
Last log	No more activity after 16-Nov-16 at 11:26:58am (UTC -7:00) - Log data indicates email was read for at least 7mins48secs (approx.)

Re-opened (by earlier reader #12)

Opened	16-Nov-16 at 11:27:00am (UTC -7:00) - 1day20hours42mins36secs after sending
Location	Pleasanton, California, United States (86% likelihood)
Opened on	c-24-130-3-238.hsd1.ca.comcast.net (24.130.3.238:40392)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Re-Opened (by earlier reader #12)

Opened	16-Nov-16 at 11:45:25am (UTC -7:00) - 1day21hours1min1sec after sending
Location	Livingston, New Jersey, United States (86% likelihood)
Opened on	176.sub-70-197-6.myvzw.com (70.197.6.176:11140)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Re-Opened (by earlier reader #8)

Opened	16-Nov-16 at 11:59:35am (UTC -7:00) - 1day21hours15mins11secs after sending
Location	New York, United States (86% likelihood)
Opened on	(107.77.70.15:60931)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_0_2 like Mac OS X) AppleWebKit/602.1.50 (KHTML, like Gecko) Mobile/14A456
Last log	No more activity after 16-Nov-16 at 12:00:31pm (UTC -7:00) - Log data indicates email was read for at least 56secs (approx.)

Re-Opened (by earlier reader #12)

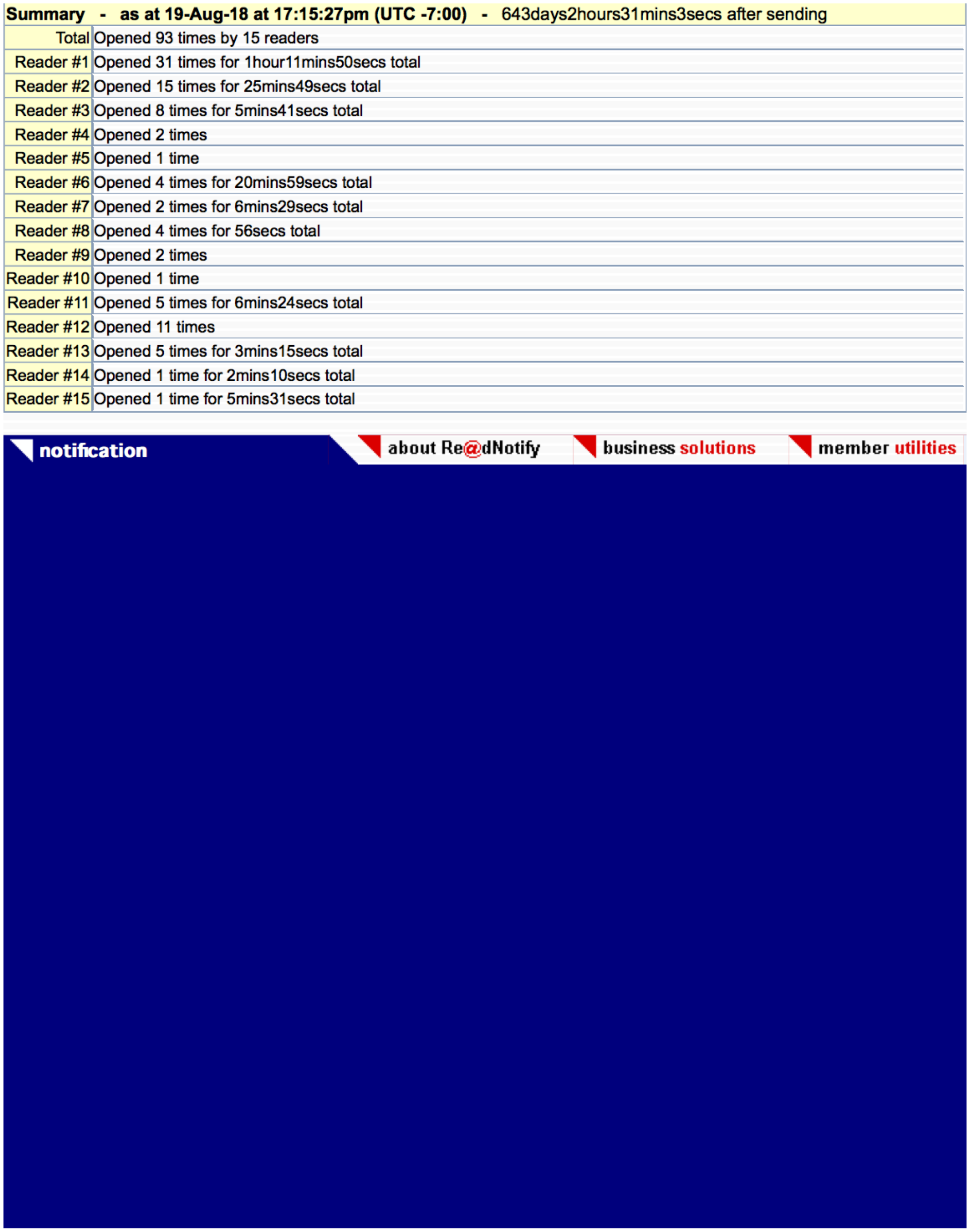
Opened	16-Nov-16 at 12:19:27pm (UTC -7:00) - 1day21hours35mins3secs after sending
Location	San Francisco, California, United States (86% likelihood)
Opened on	75-18-241-117.lightspeed.snfcca.sbcglobal.net (75.18.241.117:57340)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (iPhone; CPU iPhone OS 10_1_1 like Mac OS X) AppleWebKit/602.2.14 (KHTML, like Gecko) Mobile/14B100

Re-opened (by earlier reader #1)

Opened	16-Nov-16 at 12:22:02pm (UTC -7:00) - 1day21hours37mins38secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17.218.101.238:61159)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 16-Nov-16 at 12:22:24pm (UTC -7:00) - Log data indicates email was read for at least 22secs (approx.)

Re-Opened (by earlier reader #1)

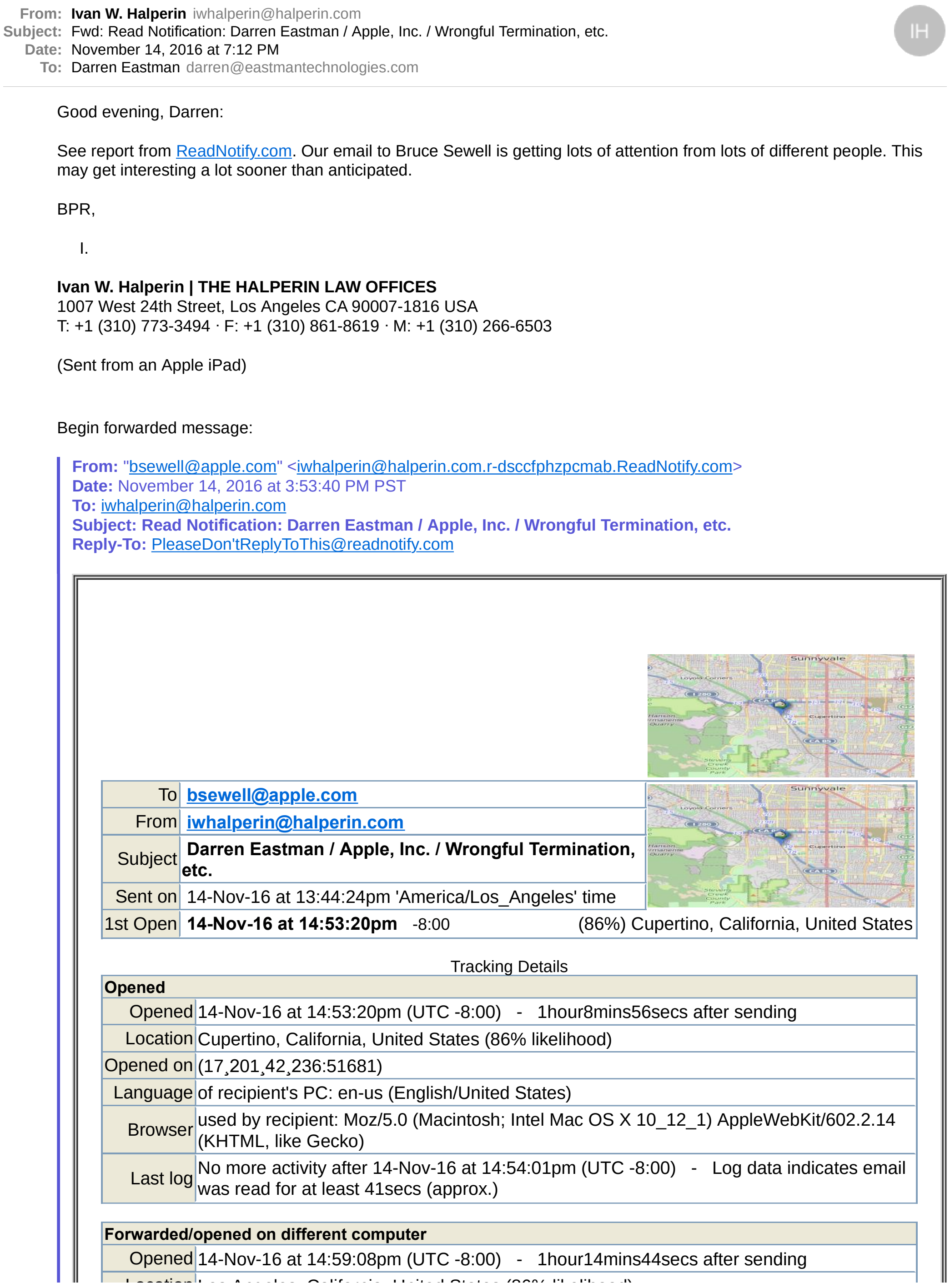
Opened	16-Nov-16 at 12:35:25pm (UTC -7:00) - 1day21hours51mins1sec after sending
Location	Dallas, Texas, United States (43% likelihood)



Summary - as at 19-Aug-18 at 17:15:27pm (UTC -7:00) - 643days2hours31mins3secs after sending

Total	Opened 93 times by 15 readers
Reader #1	Opened 31 times for 1hour11mins50secs total
Reader #2	Opened 15 times for 25mins49secs total
Reader #3	Opened 8 times for 5mins41secs total
Reader #4	Opened 2 times
Reader #5	Opened 1 time
Reader #6	Opened 4 times for 20mins59secs total
Reader #7	Opened 2 times for 6mins29secs total
Reader #8	Opened 4 times for 56secs total
Reader #9	Opened 2 times
Reader #10	Opened 1 time
Reader #11	Opened 5 times for 6mins24secs total
Reader #12	Opened 11 times
Reader #13	Opened 5 times for 3mins15secs total
Reader #14	Opened 1 time for 2mins10secs total
Reader #15	Opened 1 time for 5mins31secs total

EXHIBIT 25



From: **Ivan W. Halperin** iwhalperin@halperin.com
Subject: Fwd: Read Notification: Darren Eastman / Apple, Inc. / Wrongful Termination, etc.
Date: November 14, 2016 at 7:12 PM
To: Darren Eastman darren@eastmantechnologies.com

Good evening, Darren:

See report from ReadNotify.com. Our email to Bruce Sewell is getting lots of attention from lots of different people. This may get interesting a lot sooner than anticipated.

BPR,

I.

Ivan W. Halperin | THE HALPERIN LAW OFFICES
1007 West 24th Street, Los Angeles CA 90007-1816 USA
T: +1 (310) 773-3494 · F: +1 (310) 861-8619 · M: +1 (310) 266-6503

(Sent from an Apple iPad)

Begin forwarded message:

> **From:** "bsewell@apple.com" <iwhalperin@halperin.com.r-dsccfphzpcmab.ReadNotify.com>
> **Date:** November 14, 2016 at 3:53:40 PM PST
> **To:** iwhalperin@halperin.com
> **Subject: Read Notification: Darren Eastman / Apple, Inc. / Wrongful Termination, etc.**
> **Reply-To:** PleaseDon'tReplyToThis@readnotify.com

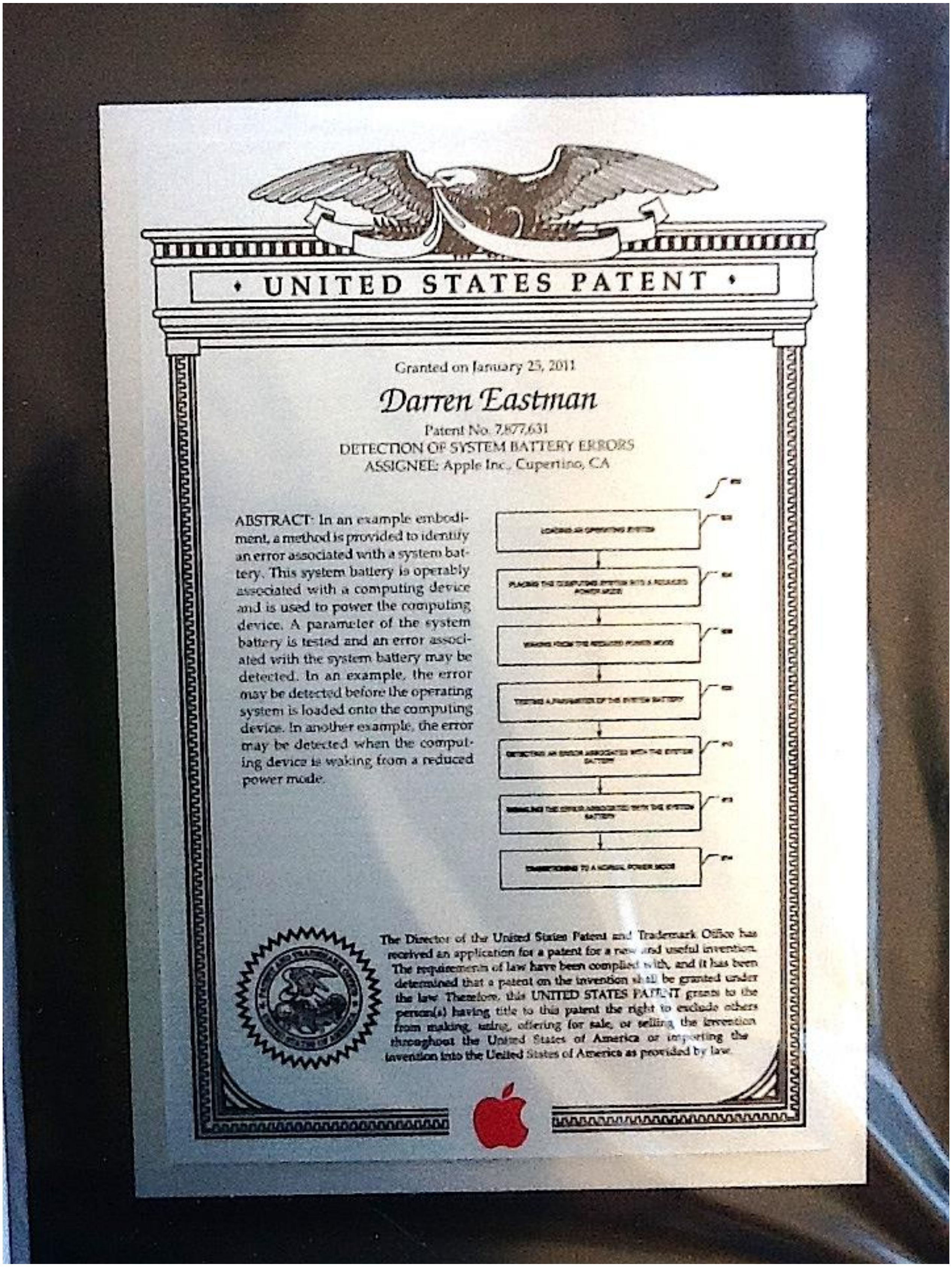
To	**bsewell@apple.com**
From	**iwhalperin@halperin.com**
Subject	**Darren Eastman / Apple, Inc. / Wrongful Termination, etc.**
Sent on	14-Nov-16 at 13:44:24pm 'America/Los_Angeles' time
1st Open	**14-Nov-16 at 14:53:20pm** -8:00 (86%) Cupertino, California, United States

Tracking Details

Opened	
Opened	14-Nov-16 at 14:53:20pm (UTC -8:00) - 1hour8mins56secs after sending
Location	Cupertino, California, United States (86% likelihood)
Opened on	(17¸201¸42¸236:51681)
Language	of recipient's PC: en-us (English/United States)
Browser	used by recipient: Moz/5.0 (Macintosh; Intel Mac OS X 10_12_1) AppleWebKit/602.2.14 (KHTML, like Gecko)
Last log	No more activity after 14-Nov-16 at 14:54:01pm (UTC -8:00) - Log data indicates email was read for at least 41secs (approx.)

Forwarded/opened on different computer	
Opened	14-Nov-16 at 14:59:08pm (UTC -8:00) - 1hour14mins44secs after sending
Location	Los Angeles, California, United States (86% likelihood)

EXHIBIT 26



UNITED STATES PATENT

Granted on January 25, 2011

Darren Eastman

Patent No. 7,877,631
DETECTION OF SYSTEM BATTERY ERRORS
ASSIGNEE: Apple Inc., Cupertino, CA

ABSTRACT: In an example embodiment, a method is provided to identify an error associated with a system battery. This system battery is operably associated with a computing device and is used to power the computing device. A parameter of the system battery is tested and an error associated with the system battery may be detected. In an example, the error may be detected before the operating system is loaded onto the computing device. In another example, the error may be detected when the computing device is waking from a reduced power mode.

LOADING AN OPERATING SYSTEM
PLACING THE COMPUTING SYSTEM INTO A REDUCED POWER MODE
WAKING FROM THE REDUCED POWER MODE
TESTING A PARAMETER OF THE SYSTEM BATTERY
DETECTING AN ERROR ASSOCIATED WITH THE SYSTEM BATTERY
SIGNALING THE ERROR ASSOCIATED WITH THE SYSTEM BATTERY
TRANSITIONING TO A NORMAL POWER MODE

The Director of the United States Patent and Trademark Office has received an application for a patent for a new and useful invention. The requirements of law have been complied with, and it has been determined that a patent on the invention shall be granted under the law. Therefore, this UNITED STATES PATENT grants to the person(s) having title to this patent the right to exclude others from making, using, offering for sale, or selling the invention throughout the United States of America or importing the invention into the United States of America as provided by law.