IN THE UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF TEXAS
DALLAS DIVISION

SEVEN NETWORKS, LLC **Plaintiff,** **v.** **MOTOROLA MOBILITY LLC** **Defendant.**	**Civil Action No. 3:21-cv-1036** **JURY TRIAL REQUESTED**

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff SEVEN Networks, LLC ("SEVEN") files this complaint for patent infringement against Defendant Motorola Mobility LLC ("Motorola") for infringement of U.S. Patents Nos. 9,661,103 ("the '103 patent"), 10,063,486 ("the '486 patent"), 9,602,457 ("the '457 patent"), 10,154,432 ("the '432 patent"), 10,178,199 ("the '199 patent"), 10,299,161 ("the '161 patent"), 10,499,339 ("the '339 patent"), 10,595,228 ("the '228 patent"), and 9,516,127 ("the '127 patent") (collectively "the Patents-in-Suit"), pursuant to 35 U.S.C. § 271.

I. PARTIES

1. Plaintiff SEVEN is a company formed under the laws of Delaware with its principal place of business at 2660 East End Boulevard South, Marshall, Texas 75672.

2. Defendant Motorola is a Delaware corporation with its headquarters and principal place of business at 222 West Merchandise Mart Plaza, Suite 1800, Chicago, Illinois 60654. On information and belief, Motorola has at least one regular and established place of business in the Northern District of Texas, including at 13301 Park Vista Blvd., Fort Worth, TX, 76177. Motorola has confirmed in court filings that it owns and operates an office in Fort Worth, TX

that, at least as of March 29, 2021, employed no less than 13 employees. Motorola has further stated in court filings that its Texas employees are in the Fort Worth area.

II. JURISDICTION AND VENUE

3. This Court has subject matter jurisdiction over this case under 28 U.S.C. §§ 1331, 1332, 1338, and 1367.

4. The amount in controversy exceeds $75,000.

5. Venue is proper in this district pursuant to 28 U.S.C. §§ 1391 and 1400(b). Motorola has committed acts of infringement, both direct and indirect, in this district with respect to each asserted patent and has a regular and established place of business in this judicial district.

6. This Court has personal jurisdiction over Motorola. Motorola has continuous and systematic business contacts with the state of Texas. Motorola, directly or through subsidiaries or intermediaries (including distributors, retailers, and others), conducts its business extensively throughout Texas, by shipping, distributing, making, using, offering for sale, selling, and advertising (including the provision of interactive web pages) their products and services in the State of Texas and the Northern District of Texas. Motorola, directly and through subsidiaries or intermediaries (including distributors, retailers, and others), has purposefully and voluntarily placed infringing products and services into this district and into the stream of commerce with the intention and expectation that they will be purchased and used by consumers in this district. Motorola has offered and sold and continues to offer and sell these infringing products and services in this district. Motorola and its customers also commit additional acts of direct infringement in this district with respect to each asserted patent through their infringing use of the accused devices, including when Motorola and its customers put the accused devices into

service and receive a benefit, and Motorola is liable for these additional acts of direct infringement and indirect infringement in this district.

III. BACKGROUND

7. For more than 20 years, SEVEN has researched and developed innovative software solutions for mobile devices. SEVEN's software has been installed on over 130 million smartphones.

8. SEVEN's technologies have also been deployed by top mobile operators worldwide. In 2005, the Wall Street Journal reported that SEVEN had "licensing agreements with 45 wireless carriers to provide [] wireless e-mail service, and its software will be included on phone models made by five major handset manufacturers, such as Nokia Corp. and Motorola Inc." Mark Heinzl, Seven Networks Plans To Buy Finnish Maker Of E-Mail Software, The Wall Street Journal, wsj.com/articles/SB111316873210802927. Another Wall Street Journal article reported that as of 2005, SEVEN had "well over 500,000 users." Sarmad Ali, Wireless Email's New Fans, The Wall Street Journal, wsj.com/articles/SB112069201280578957.

9. SEVEN has been recognized in the industry for its innovative technologies and products. SEVEN's awards and recognition include:

- 2002 Red Herring 100 Companies that Will Shape the Future;
- 2002 Frost and Sullivan Market Engineering Award for Best Mobile Office Application;
- 2002 World Economic Forum Award;
- 2002 MIT's Top Innovators of the Year Award;
- 2002 Unstrung Top 25 Private Companies List;
- 2002 Technologic Partners Investors' Choice Award;
- 2002 Red Herring Top 50 Private Companies;
- 2003 Mobile Focus Best Stuff Mobile Innovation Award Finalist;
- 2003 Networks Magazine Products of the Year;

- 2003 Entrepreneurial Company of the Year Award;
- 2004 Fierce 15 Top Wireless Companies of 2004;
- 2004 Frost and Sullivan Mobile Office Innovative Service of the Year Award;
- 2004 Frost and Sullivan Mobile Communications CEO of the Year;
- 2005 Always 100 Top Innovators;
- 2005 San Francisco Business Times Top 50 Fastest Growing Private Company;
- 2008 GSMA Winner for Best Messaging Product;
- 2010 Global Merit Award; 2011 Mobile Innovation Award;
- 2011 Finalist for Best Enabling Technology for LTE;
- 2011 Frost & Sullivan New Product Innovation Award;
- 2011 OnMobile Top 100;
- 2011 GSMA Best Technology Breakthrough Award at Mobile World Congress;
- 2013 Best Free Android Apps, Best LTE Traffic Management Shortlist by TechRadar;
- 2013 OnMobile Companies to Watch List; 2013 Mobile Merit Award.

10. The inventions of the Patents-in-Suit address technological problems and provide technological solutions that were not well-understood, routine, or conventional at the time of the invention. A person of ordinary skill in the art reading the Patents-in-Suit and their claims would understand that the patents' disclosures and claims are drawn to solving specific, technical problems. Moreover, a person of ordinary skill in the art would understand that the claimed subject matter represents advancements in the technical fields of the Patents-in-Suit. The claims do not preempt all techniques for or approaches to accomplishing the same or a similar end to what they recite. For example, the claims do not preempt the use of the techniques taught in the prior art cited on the face of the Patents-in-Suit. The large volume of prior art cited on the faces of the Patents-in-Suit, none of which, as the Examiners found, discloses or render obvious the

claimed inventions further shows that the claims are not well-understood, routine, or conventional.

IV. THE PATENTS-IN-SUIT

The '103 Patent

11. On May 23, 2017, the United States Patent and Trademark Office issued U.S. Patent No. 9,661,103 ("the '103 patent"), entitled "Mobile Device Having Improved Polling Characteristics for Background Applications." SEVEN is the assignee of all rights, title, and interest in and to the '103 patent, including the right to recover damages for present, past, and future infringement. The '103 patent is valid and enforceable.

12. The'103 patent addresses technological problems related to mobile networks that are designed "for high-throughput of large amounts of data," not for "mobile applications that require frequent, low-throughput request of small amounts of data." '103, Col. 2:46-49. The patent states that "[e]xisting networks also do not take into account different types of mobile traffic and priorities of the different types of traffic[.]" '103, Col. 2:49-51.

13. The claims of the '103 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the '103 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

14. A person of ordinary skill in the art reading the '103 patent and its claims would understand that the patent's disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would

understand that the claims' subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

15. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the '103 patent are directed to specific improvements in the field. Accordingly, each claim of the '103 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The '486 Patent

16. On August 28, 2018, the United States Patent and Trademark Office issued U.S. Patent No. 10,063,486 ("the '486 patent"), entitled "Offloading Application Traffic to a Shared Communication Channel for Signal Optimization in a Wireless Network for Traffic Utilizing Proprietary and Non-Proprietary Protocols." SEVEN is the assignee of all rights, title, and interest in and to the '486 patent, including the right to recover damages for present, past, and future infringement. The '486 patent is valid and enforceable.

17. The'486 patent recites technological problems related to the "increasing amount of mobile traffic" moving to vendor-specific proprietary protocols. '486, Col. 1:42-43. "More and more of the application traffic that causes signaling now includes significant contribution from proprietary protocols on top of traffic utilizing standardized protocols such as HTTP/HTTPS." '486, Col. 1:46-50. The '486 patent provides a solution that conserves network and battery usage.

18. The claims of the '486 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it

on a computer. Instead, the claims of the '486 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

19. A person of ordinary skill in the art reading the '486 patent and its claims would understand that the patent's disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would understand that the claims' subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

20. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the '486 patent are directed to specific improvements in the field. Accordingly, each claim of the '486 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The '457 Patent

21. On March 21, 2017, the United States Patent and Trademark Office issued U.S. Patent No. 9,602,457 ("the '457 patent"), entitled "Mobile Device Having Power Save Feature for Establishing Communications." SEVEN is the assignee of all rights, title, and interest in and to the '457 patent, including the right to recover damages for present, past, and future infringement. The '457 patent is valid and enforceable.

22. The ’457 patent addresses technological problems related to the utilization of resources in a mobile device. Because existing solutions resulted in high power consumption, a need existed for a solution that would send application data requests without such a large impact on power consumption. ’457, Col. 1:26-2:7; 8:64-9:19; p. 160-165 of 60/403,249. The claims of the ’457 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the ’457 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

23. A person of ordinary skill in the art reading the ’457 patent and its claims would understand that the patent’s disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would understand that the claims’ subject matter presents advancements in the field. The claims do not preempt all types of improving power management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

24. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the ’457 patent are directed to specific improvements in the field. Accordingly, each claim of the ’457 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The ’432 Patent

25. On December 11, 2018, the United States Patent and Trademark Office issued U.S. Patent No. 10,154,432 (“the ’432 patent”), entitled “Mobile Application Traffic

Optimization." SEVEN is the assignee of all rights, title, and interest in and to the '432 patent, including the right to recover damages for present, past, and future infringement. The '432 patent is valid and enforceable.

26. The inventions of the '432 patent solve technological problems related to mobile device battery drain. The '432 patent explains that applications such as "push email, instant messaging, visual voicemail and voice and video telephony" "typically require an always-on IP connection and frequent transmit of small bits of data." '432, Col. 31:53–59. The "rapid increase of popularity" of these applications exaggerated the problem of battery drain due to the mobile device radio being frequently engaged. '432, Col. 31:60–32:3. Prior art solutions to this problem "typically assume lack of coordination between the user, the application and the network, forcing the network to guess what the application might be doing, and [the] application to act independently of whether [the] user actually is available for taking action on any network initiated activity." '432, Col. 32:4–17. The patent teaches that "[f]requently sending bursts of data in a wireless network also result in high battery consumption due to the constant need of powering/re-powering the radio module." '432, Col. 2:35-43.

27. The claims of the '432 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the '432 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

28. A person of ordinary skill in the art reading the ’432 patent and its claims would understand that the patent’s disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would understand that the claims’ subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

29. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the ’432 patent are directed to specific improvements in the relevant field. Accordingly, each claim of the ’432 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The ’199 Patent

30. On January 8, 2019 the United States Patent and Trademark Office issued U.S. Patent No. 10,178,199 (“the ’199 patent”), entitled “Intelligent Alarm Manipulator and Resource Tracker.” SEVEN is the assignee of all rights, title, and interest in and to the ’199 patent, including the right to recover damages for present, past, and future infringement. The ’199 patent is valid and enforceable.

31. The inventions of the ’199 patent solve technological problems related to resource usage on a mobile device. Existing solutions had high utilization of network resources, power/battery resources, CPU resources, and memory resources. ’199, Col. 17:7-19. Because existing solutions had high power consumption and consumption of other resources, a need existed for a solution that would give users control over background tasks and improve power

and resource consumption. The specification of the '199 patent provides such a solution. '199, Col. 4:5-9.

32. The claims of the '199 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the '199 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

33. A person of ordinary skill in the art reading the '199 patent and its claims would understand that the patent's disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would understand that the claims' subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

34. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the '199 patent are directed to specific improvements in the field. Accordingly, each claim of the '199 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The '161 Patent

35. On May 21, 2019, the United States Patent and Trademark Office issued U.S. Patent No. 10,299,161 ("the '161 patent"), entitled "Predictive Fetching of Background Data

Request in Resource Conserving Manner." SEVEN is the assignee of all rights, title, and interest in and to the '161, including the right to recover damages for present, past, and future infringement. The '161 patent is valid and enforceable.

36. The '161 patent addresses technological problems related to "high battery consumption due to the constant need of powering/re-powering the radio module." '161, Col. 2:49-51. The'161 patent addresses mobile applications which frequently send and receive data in mobile networks that are "designed and optimized for high-throughput of large amounts of data, not for mobile applications that require frequent, but low-throughput and/or small amounts of data." '161, Col. 2:14-17; see also Col. 2:34-51.

37. The claims of the '161 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the '161 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

38. A person of ordinary skill in the art reading the '161 patent and its claims would understand that the patent's disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would understand that the claims' subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

39. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the '161 patent are directed to specific improvements in the field. Accordingly, each claim of the '161 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The '339 Patent

40. On December 3, 2019, the United States Patent and Trademark Office issued U.S. Patent No. 10,499,339 ("the '339 patent"), entitled "Optimizing Mobile Network Traffic Coordination Across Multiple Applications Running on a Mobile Device." SEVEN is the assignee of all rights, title, and interest in and to the '339 patent, including the right to recover damages for present, past, and future infringement. The '339 patent is valid and enforceable.

41. The'339 patent addresses technological problems related to "high battery consumption due to the constant need of powering/re-powering the radio module." '339, Col. 2:59-61. It addresses technological problems relating to mobile applications which frequently send and receive data in mobile networks that are designed "for high-throughput of large amounts of data, not for applications that require frequent, but low-throughput and/or small amounts of data." '339, Col. 2:25-28; see also '339, Col. 2:36-61.

42. The claims of the '339 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the '339 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

43. A person of ordinary skill in the art reading the ’339 patent and its claims would understand that the patent’s disclosure and claims are drawn to solving a specific, technical problem arising in the field. Moreover, a person of ordinary skill in the art would understand that the claims’ subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

44. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the ’339 patent are directed to specific improvements in the field. Accordingly, each claim of the ’339 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The ’228 Patent

45. On March 17, 2020, the United States Patent and Trademark Office issued U.S. Patent No. 10,595,228 (“the ’228 patent”), entitled “Mobile Device Configured for Operating in a Power Save Mode and a Traffic Optimization Mode and Related Method.” SEVEN is the assignee of all rights, title, and interest in and to the ’228 patent, including the right to recover damages for present, past, and future infringement. The ’228 patent is valid and enforceable.

46. The inventions of the ’228 patent solve technological problems related to mobile device resource consumption. The ’228 patent explains that “the increased availability of free apps only makes network congestion worse with constant signaling from the application to the application stores and/or advertiser websites.” ’228, Col. 1:41-44. Network operators and carriers “lack a mechanism of monitoring usage, in particular application usage and its impact on network resources.” ’228, Col. 1:48-50. Many of the performance enhancing solutions and

standards, such as "3.5G, LTE, 4G, and WiMAX, [were] focused on providing increased bandwidth," though "the standard does not address battery life very well." '228, Col. 7:5-11.

47. The claims of the '228 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the '228 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques suffered drawbacks solved by the new and novel invention.

48. A person of ordinary skill in the art reading the '228 patent and its claims would understand that the patent's disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would understand that the claims' subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

49. In light of the foregoing, a person of ordinary skill in the art would understand that the claims of the '228 patent are directed to specific improvements in the field. Accordingly, each claim of the '228 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

The '127 Patent

50. On Dec. 6, 2016, the United States Patent and Trademark Office issued U.S. Patent No. 9,516,127 ("the '127 patent"), entitled "Intelligent Alarm Manipulator and Resource

Tracker." Plaintiff is the assignee of all rights, title, and interest in and to the '127 patent, including the right to recover damages for present, past, and future infringement. The '127 patent is valid and enforceable.

51. The inventions of the '127 patent resolve technological problems related to the utilization of resources at a mobile device. Existing solutions had high utilization of network resources, power/battery resources, CPU resources, and memory resources. '127, Col. 17:2-15. Because existing solutions had high power consumption and consumption of other resources, a need existed for a solution that would improve power and resource consumption. The specification of the '127 patent provides such a solution. '127, Col. 4:3-7.

52. The claims of the '127 patent do not merely recite the performance of some business practice known from the pre-computer world along with the requirement to perform it on a computer. Instead, the claims of the '127 patent recite one or more inventive concepts that are rooted in computerized technology, and overcome technical problems specifically arising in that realm. These inventive solutions overcome one or more problems of the prior art. As detailed by the specification, the prior techniques solved by the new and novel invention.

53. A person of ordinary skill in the art reading the '127 patent and its claims would understand that the patent's disclosure and claims are drawn to solving a specific, technical problem arising in mobile devices. Moreover, a person of ordinary skill in the art would understand that the claims' subject matter presents advancements in the field. The claims do not preempt all types of mobile device resource management. For example, the claims do not preempt the use of techniques taught in the dozens of prior art references cited on the face of the patent.

54. A person of ordinary skill in the art would understand that the claims of the '127 patent are directed to specific improvements in the field. Accordingly, each claim of the '127 patent recites a combination of elements sufficient to ensure that the claim in practice amounts to significantly more than a patent claiming an abstract concept.

V. CLAIMS FOR PATENT INFRINGEMENT

55. The allegations below are exemplary and without prejudice to Plaintiff's infringement contentions and expert reports to be provided pursuant to the Court's scheduling order and local rules. Plaintiff's claim construction contentions regarding the meaning and scope of the claim terms will be provided under the Court's scheduling order and local rules. As detailed below, each element of at least one claim of each of the Patents-in-Suit is literally present in the Accused Products. To the extent that any element is not literally present, each such element is present under the doctrine of equivalents. Plaintiff's analysis below should not be taken as an admission that the preamble of the claims is limiting. While publicly available information is cited below, Plaintiff may rely on other forms of evidence to show infringement.

56. The Accused Products include at least the following products, as well as products with reasonably similar functionality: Motorola Moto G 5G Plus, Motorola One Fusion, Motorola One Fusion+, Motorola Moto G Fast, Motorola Moto E (2020), Motorola Moto G Pro, Motorola Edge+, Motorola Edge, Motorola Moto G8 Power Lite, Motorola Moto E6s (2020), Motorola Moto G8, Motorola Moto G Stylus, Motorola Moto G Power, Motorola Moto G8 Power, Motorola Razr (2020), Motorola One Hyper, Motorola Razr 2019, Motorola Moto G8 Play, Motorola Moto E6 Play, Motorola Moto G8 Plus, Motorola One Macro, Motorola Moto E6 Plus, Motorola One Zoom, Motorola One Action, Motorola Moto E6, Motorola Moto Z4, Motorola One Vision, Motorola Moto G7 Plus, Motorola Moto G7, Motorola Moto G7 Power, Motorola Moto G7 Play, Motorola One (P30 Play), Motorola One Power (P30 Note), Motorola

Moto Z3, Motorola Moto Z3 Play, Motorola Moto E5 Cruise, Motorola Moto E5 Play Go, Motorola Moto E5 Play, Motorola Moto E5 Plus, Motorola Moto E5, Motorola P30, Motorola Moto G6 Play, Motorola Moto G6 Plus, Motorola Moto G6, Motorola Moto X4, Motorola Moto G5S Plus, Motorola Moto G5S, Motorola Moto Z2 Force, Motorola Moto E4 Plus, Motorola Moto E4 Plus (USA), Motorola Moto E4, Motorola Moto E4 (USA), Motorola Moto Z2 Play, Motorola Moto C Plus, Motorola Moto C, Motorola Moto G5 Plus, Motorola Moto G5, Motorola Moto M, Motorola Moto E3 Power, Motorola Moto Z Play, Motorola Moto E3, Motorola Moto Z Force, Motorola Moto Z, Motorola Moto G4 Plus, Motorola Moto G4, Motorola Moto G4 Play, Moto Z2 Force, Moto G Stylus, One 5G Ace, Moto G Power, Moto G9, Moto G9 Plus, Moto G9 Power, Moto G9 Play, Motorola Moto G Turbo, Motorola Moto X Force, Motorola Droid Turbo 2, Motorola Droid Maxx 2, Motorola Moto X Style, Motorola Moto X Play Dual SIM, Motorola Moto X Play, Motorola Moto G Dual SIM (3rd gen), Motorola Moto G (3rd gen).

57. The Accused Products also include varieties, including with size differences (such as regular and plus versions) and screen display differences. Further identification of the Accused Products will be provided with Plaintiff's infringement contentions pursuant to the Court's scheduling order and local rules.

58. In accordance with 35 U.S.C. § 287, Motorola has had actual notice and knowledge of all of the Patents-in-Suit no later than the filing of this complaint and/or the date this complaint was served upon Motorola. On information and belief, Motorola continues without license to make, use, import/export into/from, market, offer for sale, and/or sell in the United States products that infringe the Patents-in-Suit.

59. Motorola has directly and indirectly infringed and continues to directly and indirectly infringe the Patents-in-Suit by engaging in acts constituting infringement under 35 U.S.C. § 271.

60. On information and belief, Motorola makes, uses, sells, and/or offers to sell Accused Products and/or components thereof in this district and elsewhere in the United States.

61. On information and belief, Motorola imports Accused Products and/or components into the United States.

62. Motorola instructs its customers to use the Accused Products in manners that infringe the Patents-in-Suit. For example, Motorola provides instruction manuals for the Accused Products and describes, markets, and/or advertises infringing functionality on its website and in other Motorola documentation.

63. Motorola tests the Accused Products in the United States, thereby infringing the Patents-in-Suit. On information and belief, Motorola uses the Accused Products, in this district, thus infringing the Patents-in-Suit.

64. Motorola's acts of infringement have caused damage to SEVEN. SEVEN is entitled to recover from Motorola the damages sustained by SEVEN as a result of Motorola's wrongful acts in an amount subject to proof at trial.

65. In the interest of providing detailed averments of infringement, SEVEN has identified below at least one claim per patent to demonstrate infringement. However, the selection of claims should not be considered limiting; additional claims of the Patents-in-Suit that are infringed by Motorola will be disclosed in compliance with the Court's rules related to infringement contentions. Further, SEVEN has provided exemplary evidence of infringement, however, this evidence should not be considered limiting; additional evidence demonstrating the

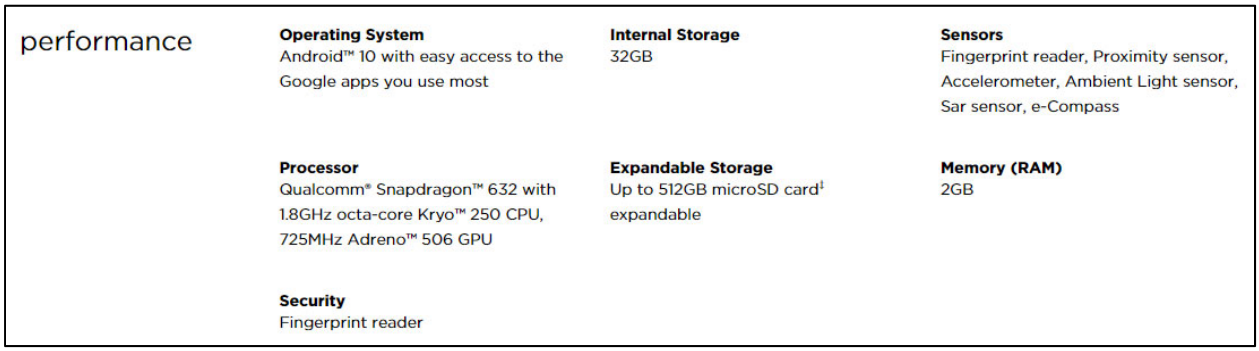
claims of the Patents-in-Suit that are infringed by Motorola will be disclosed in compliance with the Court's rules related to infringement contentions and expert reports.

COUNT I: PATENT INFRINGEMENT OF THE '103 PATENT

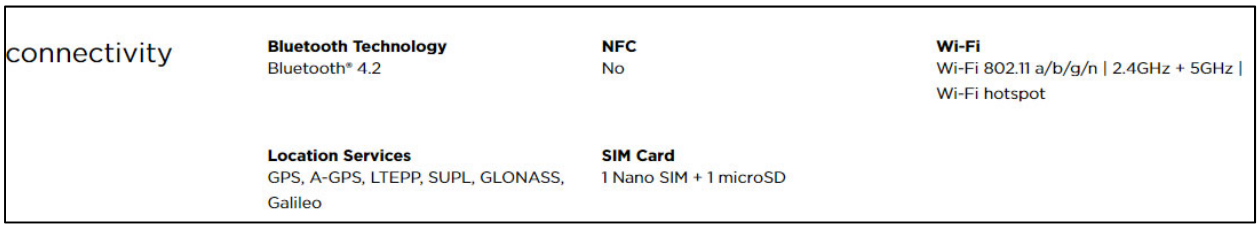
66. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

67. Motorola infringes (literally and/or under the doctrine of equivalents) the '103 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '103 patent including at least the Accused Products.

68. For example and as shown below, the Accused Products infringe at least claim 1 of the '103 patent. As shown below by exemplary evidence, Accused Products are "configured for aligning data transfer from the mobile device to optimize connections made by the mobile device in a wireless network" and comprise "a memory, a backlight, a radio, and a processor."

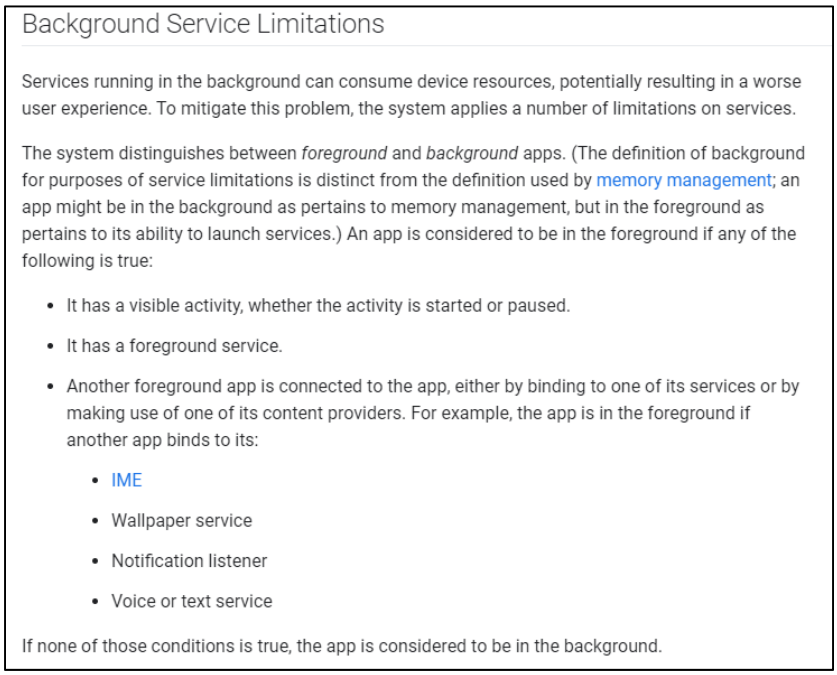


https://www.motorola.com/us/smartphones-moto-e/p



https://www.motorola.com/us/smartphones-moto-e/p

69. The Accused Products are configured to, "while the backlight of the mobile device is on," "detect that a first application is executing in a background of the mobile device" and "detect that a second application is executing in a foreground of the mobile device," as shown in the exemplary documentation below.



Background Service Limitations

Services running in the background can consume device resources, potentially resulting in a worse user experience. To mitigate this problem, the system applies a number of limitations on services.

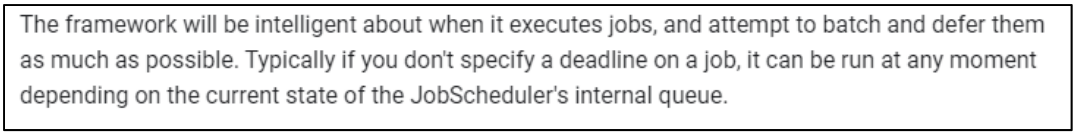
The system distinguishes between *foreground* and *background* apps. (The definition of background for purposes of service limitations is distinct from the definition used by memory management; an app might be in the background as pertains to memory management, but in the foreground as pertains to its ability to launch services.) An app is considered to be in the foreground if any of the following is true:

- It has a visible activity, whether the activity is started or paused.
- It has a foreground service.
- Another foreground app is connected to the app, either by binding to one of its services or by making use of one of its content providers. For example, the app is in the foreground if another app binds to its:
  - IME
  - Wallpaper service
  - Notification listener
  - Voice or text service

If none of those conditions is true, the app is considered to be in the background.

https://developer.android.com/about/versions/oreo/background

70. The Accused Products are configured to, "while the backlight of the mobile device is on," "batch a first set of data for the first application" and "transmit the first set of batched data for the first application," as shown by the exemplary documentation below.



The framework will be intelligent about when it executes jobs, and attempt to batch and defer them as much as possible. Typically if you don't specify a deadline on a job, it can be run at any moment depending on the current state of the JobScheduler's internal queue.

https://developer.android.com/reference/android/app/job/JobScheduler

> **When to use WorkManager**
>
> WorkManager handles background work that needs to run when various constraints are met, regardless of whether the application process is alive or not. Background work can be started when the app is in the background, when the app is in the foreground, or when the app starts in the foreground but goes to the background. Regardless of what the application is doing, background work should continue to execute, or be restarted if Android kills its process.

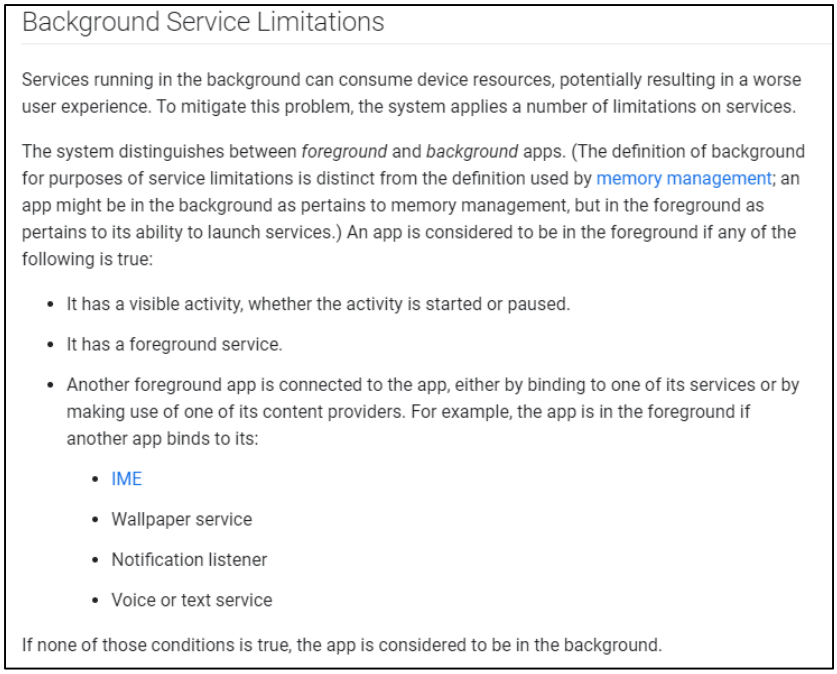
https://medium.com/androiddevelopers/introducing-workmanager-2083bcfc4712

71. The Accused Products are configured to, "while the backlight of the mobile device is on," "transmit data for the second application at a time when the second application requests transmission," as shown by the exemplary documentation below.

> While an app is in the foreground, it can create and run both foreground and background services freely. When an app goes into the background, it has a window of several minutes in which it is still allowed to create and use services. At the end of that window, the app is considered to be *idle*. At this time, the system stops the app's background services, just as if the app had called the services' `Service.stopSelf()` methods.

https://developer.android.com/about/versions/oreo/background

72. The Accused Products are configured to, "while the backlight of the mobile device is off in response to inactivity of the mobile device," "detect that the second application is executing in the background of the mobile device," as shown by the exemplary documentation below.



Background Service Limitations

Services running in the background can consume device resources, potentially resulting in a worse user experience. To mitigate this problem, the system applies a number of limitations on services.

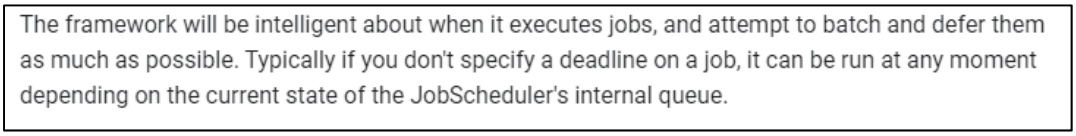
The system distinguishes between *foreground* and *background* apps. (The definition of background for purposes of service limitations is distinct from the definition used by memory management; an app might be in the background as pertains to memory management, but in the foreground as pertains to its ability to launch services.) An app is considered to be in the foreground if any of the following is true:

- It has a visible activity, whether the activity is started or paused.
- It has a foreground service.
- Another foreground app is connected to the app, either by binding to one of its services or by making use of one of its content providers. For example, the app is in the foreground if another app binds to its:
  - IME
  - Wallpaper service
  - Notification listener
  - Voice or text service

If none of those conditions is true, the app is considered to be in the background.

https://developer.android.com/about/versions/oreo/background

73. The Accused Products are configured to, "while the backlight of the mobile device is off in response to inactivity of the mobile device," "batch a second set of data for the first application and the second application" and "transmit the second set of batched data for the first application and the second application, wherein the transmission of the second set of batched data occurs after at least a predetermined period of time," as shown by the exemplary documentation below.



The framework will be intelligent about when it executes jobs, and attempt to batch and defer them as much as possible. Typically if you don't specify a deadline on a job, it can be run at any moment depending on the current state of the JobScheduler's internal queue.

https://developer.android.com/reference/android/app/job/JobScheduler

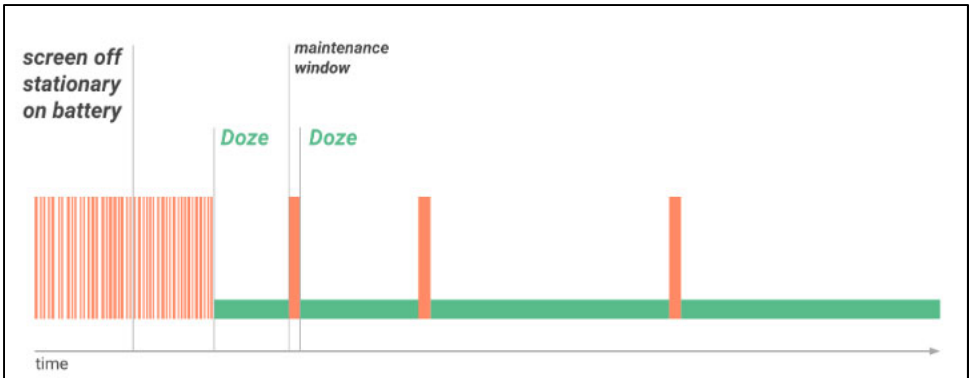


Figure 1. Doze provides a recurring maintenance window for apps to use the network and handle pending activities.

https://developer.android.com/training/monitoring-device-state/doze-standby

74. Thus, the Accused Products directly infringe one or more claims of the '103 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '103 patent.

75. Motorola indirectly infringes the '103 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-users directly infringe through their use of the inventions claimed in the '103 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products

in an infringing manner. As a result of Motorola's inducement, Motorola's customers and end-users use the Motorola products in the way Motorola intends and directly infringe the '103 patent. Motorola performs these affirmative acts with knowledge of the '103 patent and with the intent, or willful blindness, that the induced acts directly infringe the '103 patent. Motorola has had knowledge and notice of the '103 patent at least as of the filing of this complaint.

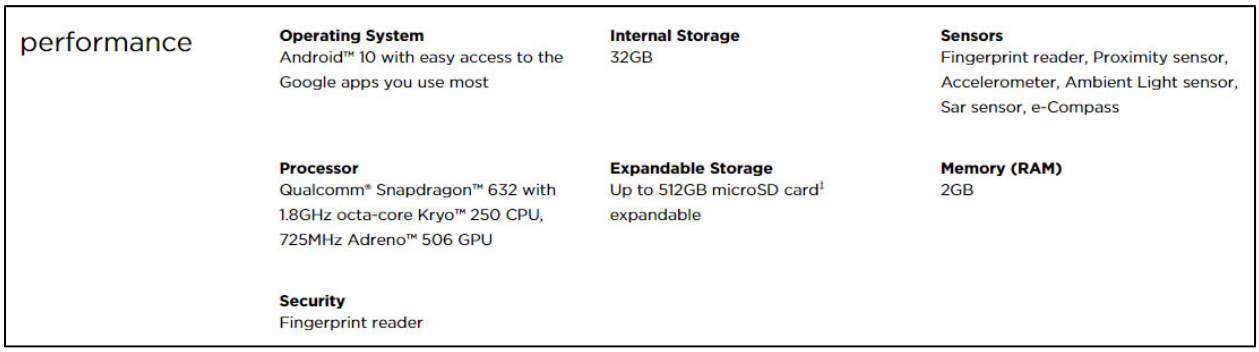
76. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

COUNT II: PATENT INFRINGEMENT OF THE '486 PATENT

77. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

78. Motorola infringes (literally and/or under the doctrine of equivalents) the '486 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '486 patent including at least the Accused Products.

79. For example and as shown below, the Accused Products infringe at least claim 11 of the '486 patent. As shown below by the exemplary evidence, the Accused Products are "mobile devices" with "a memory and a processor."



https://www.motorola.com/us/smartphones-moto-e/p

80. The Accused Products are configured for "detecting user inactivity on a mobile device," as shown in the exemplary documentation below.

> If a user leaves a device unplugged and stationary for a period of time, with the screen off, the device enters Doze mode. In Doze mode, the system attempts to conserve battery by restricting apps' access to network and CPU-intensive services. It also prevents apps from accessing the network and defers their jobs, syncs, and standard alarms.

https://developer.android.com/training/monitoring-device-state/doze-standby

81. The Accused Products are configured for, "in response to detected inactivity," "blocking a first channel to reduce network signaling in a network and to reduce battery consumption, wherein the first channel is a channel specific to a first application executing on the mobile device,", as shown by the exemplary evidence below.

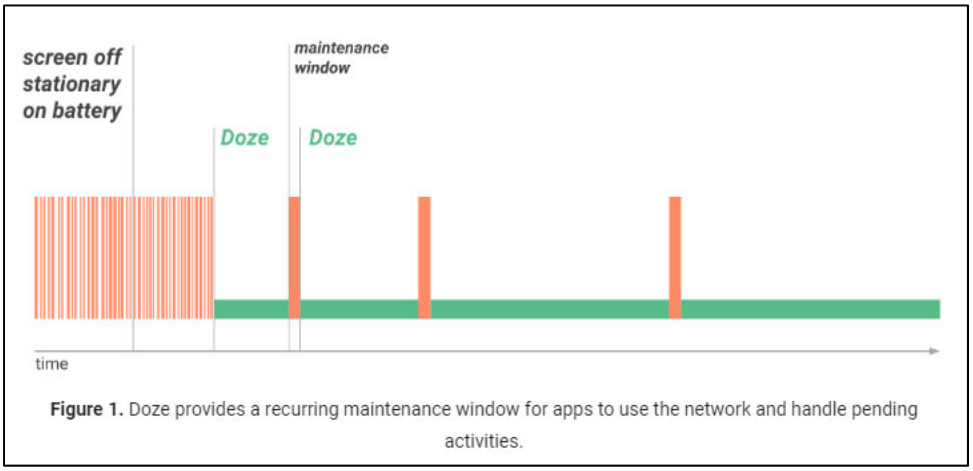


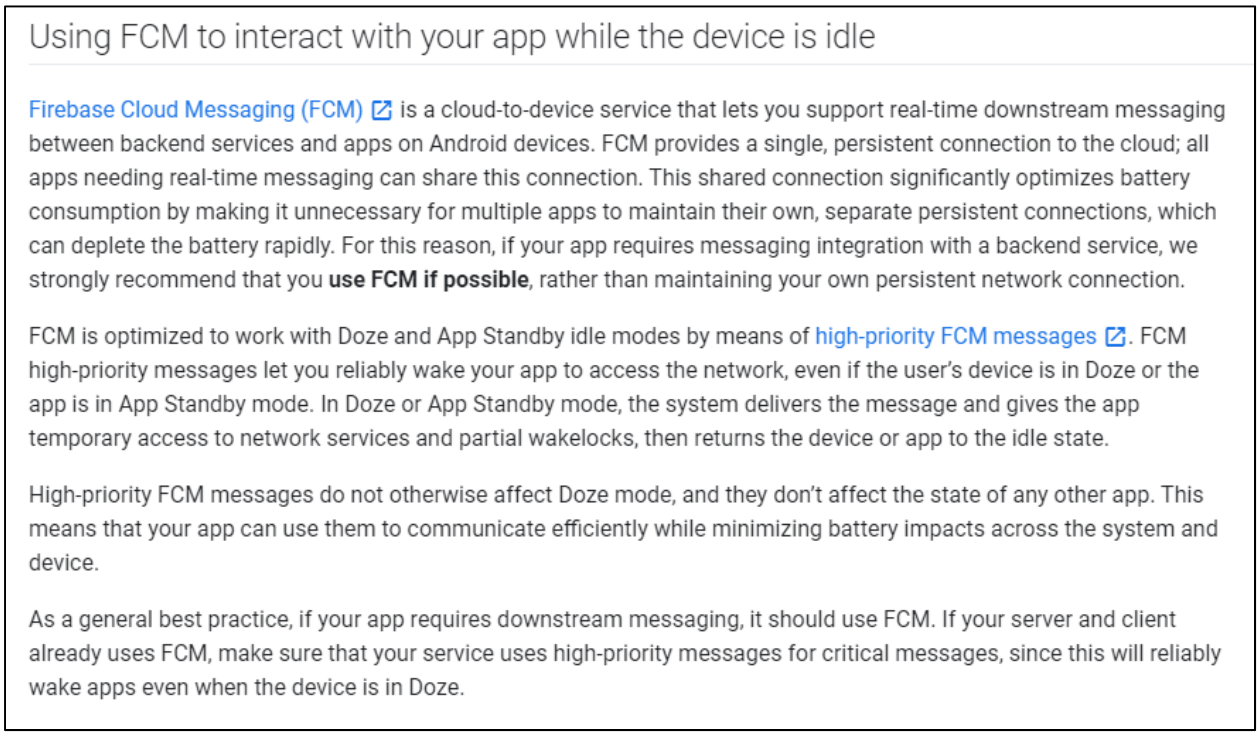
Figure 1. Doze provides a recurring maintenance window for apps to use the network and handle pending activities.

https://developer.android.com/training/monitoring-device-state/doze-standby

> ## Support for other use cases
>
> Almost all apps should be able to support Doze by managing network connectivity, alarms, jobs, and syncs properly, and by using FCM high-priority messages. For a narrow set of use cases, this might not be sufficient. For such cases, the system provides a configurable whitelist of apps that are **partially exempt** from Doze and App Standby optimizations.
>
> An app that is whitelisted can use the network and hold partial wake locks during Doze and App Standby. However, **other restrictions still apply** to the whitelisted app, just as they do to other apps. For example, the whitelisted app's jobs and syncs are deferred (on API level 23 and below), and its regular `AlarmManager` alarms do not fire. An app can check whether it is currently on the exemption whitelist by calling `isIgnoringBatteryOptimizations()`.
>
> Users can manually configure the whitelist in **Settings > Battery > Battery Optimization.** Alternatively, the system provides ways for apps to ask users to whitelist them.

https://developer.android.com/training/monitoring-device-state/doze-standby

82. The Accused Products are configured for, "in response to detected inactivity," the first application "to receive communications over a second channel that is established over the network" and "a second application executing on the mobile device [to] also receive[] communications over the second channel," as shown by the exemplary evidence below.



Using FCM to interact with your app while the device is idle

Firebase Cloud Messaging (FCM) is a cloud-to-device service that lets you support real-time downstream messaging between backend services and apps on Android devices. FCM provides a single, persistent connection to the cloud; all apps needing real-time messaging can share this connection. This shared connection significantly optimizes battery consumption by making it unnecessary for multiple apps to maintain their own, separate persistent connections, which can deplete the battery rapidly. For this reason, if your app requires messaging integration with a backend service, we strongly recommend that you **use FCM if possible**, rather than maintaining your own persistent network connection.

FCM is optimized to work with Doze and App Standby idle modes by means of high-priority FCM messages. FCM high-priority messages let you reliably wake your app to access the network, even if the user's device is in Doze or the app is in App Standby mode. In Doze or App Standby mode, the system delivers the message and gives the app temporary access to network services and partial wakelocks, then returns the device or app to the idle state.

High-priority FCM messages do not otherwise affect Doze mode, and they don't affect the state of any other app. This means that your app can use them to communicate efficiently while minimizing battery impacts across the system and device.

As a general best practice, if your app requires downstream messaging, it should use FCM. If your server and client already uses FCM, make sure that your service uses high-priority messages for critical messages, since this will reliably wake apps even when the device is in Doze.

https://developer.android.com/training/monitoring-device-state/doze-standby

83. The Accused Products are configured for, "in response to detected inactivity," "monitoring the application traffic for receipt of a message for the first application over the second channel, wherein the message informs the mobile device that there is new data for receipt at an application server associated with the first application." "unblocking the first channel based on the monitored application traffic so that the first application can perform an action over the first channel," and "re-blocking the first channel after a predetermined period of time," as shown by the exemplary evidence below.

> FCM is optimized to work with Doze and App Standby idle modes by means of high-priority FCM messages. FCM high-priority messages let you reliably wake your app to access the network, even if the user's device is in Doze or the app is in App Standby mode. In Doze or App Standby mode, the system delivers the message and gives the app temporary access to network services and partial wakelocks, then returns the device or app to the idle state.

https://developer.android.com/training/monitoring-device-state/doze-standby

84. The Accused Products are configured for "unblocking the first channel when user activity is detected, wherein the user activity is based on of whether the mobile device is being interacted with by a user," as shown by the exemplary evidence below.

> As soon as the user wakes the device by moving it, turning on the screen, or connecting a charger, the system exits Doze and all apps return to normal activity.

https://developer.android.com/training/monitoring-device-state/doze-standby

85. Thus, the Accused Products directly infringe one or more claims of the '486 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '486 patent.

86. Motorola indirectly infringes the '486 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-users directly infringe through their use of the inventions claimed in the '486 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary evidence of infringement. By way of example, Motorola user manuals and documentation

(including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola's inducement, Motorola's customers and end-users use the Motorola products in the way Motorola intends and directly infringe the '486 patent. Motorola performs these affirmative acts with knowledge of the '486 patent and with the intent, or willful blindness, that the induced acts directly infringe the '486 patent. Motorola has had knowledge and notice of the '486 patent at least as of the filing of this complaint.

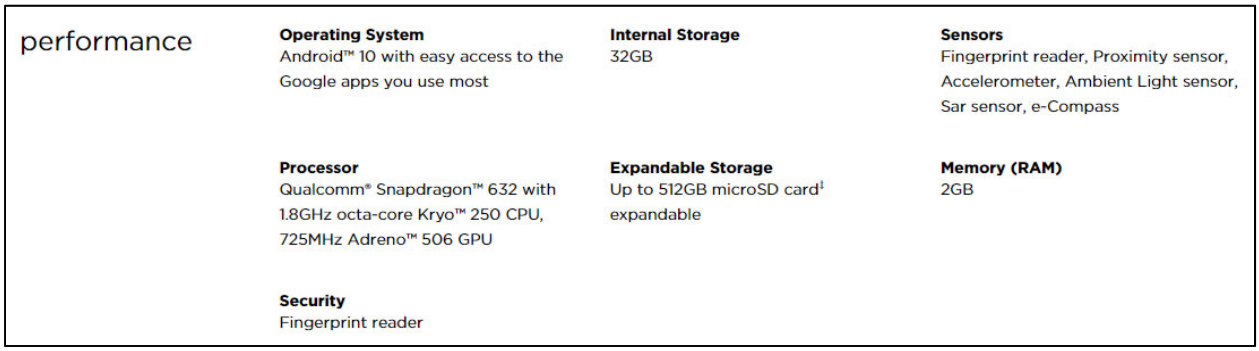
87. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

COUNT III: PATENT INFRINGEMENT OF THE '457 PATENT

88. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

89. Motorola infringes (literally and/or under the doctrine of equivalents) the '457 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '457 patent including at least the Accused Products.

90. For example and as shown below, the Accused Products infringe at least claim 1 of the '457 patent. As shown by the exemplary evidence below, the Accused Products are "mobile devices" with "a battery and a processor."



https://www.motorola.com/us/smartphones-moto-e/p

91. The Accused Products are configured to "send application data requests from an application operating on the mobile device to a first server," as shown by the exemplary evidence below.

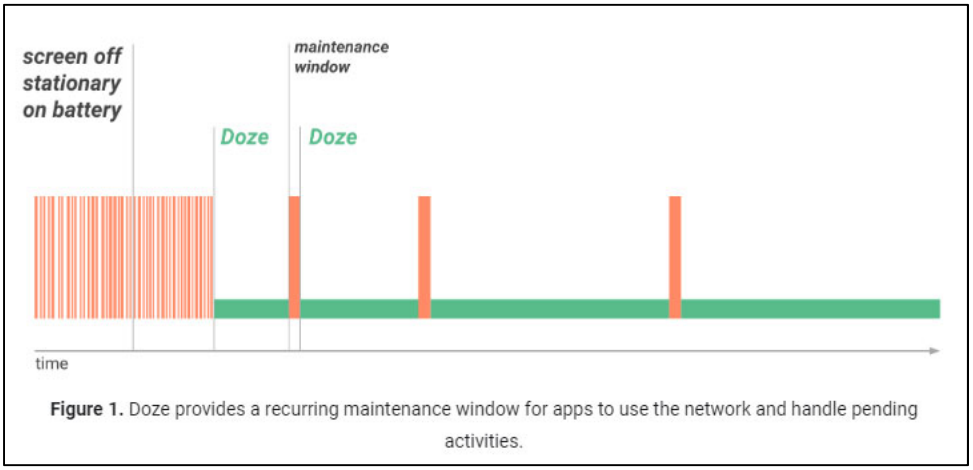


Figure 1. Doze provides a recurring maintenance window for apps to use the network and handle pending activities.

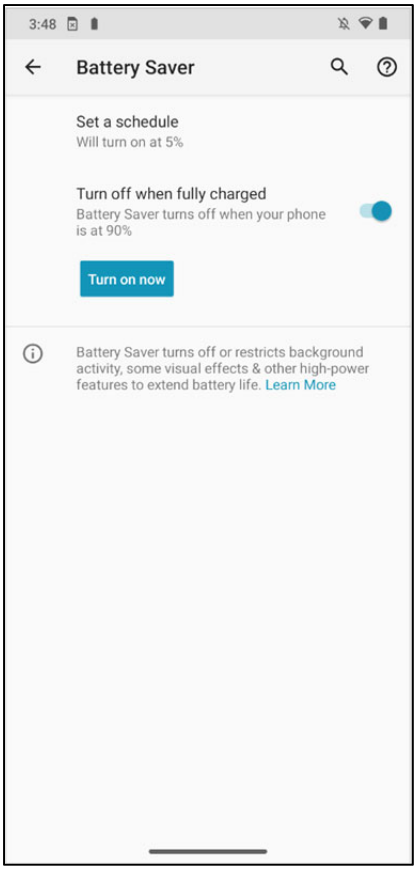
https://developer.android.com/training/monitoring-device-state/doze-standby

> Idle devices in Doze periodically enter a maintenance window, during which apps can complete pending work (syncs, jobs, etc.). Doze then resumes sleep for a longer period of time, followed by another maintenance window. The platform continues the Doze sleep/maintenance sequence, increasing the length of idle each time, until a maximum of a few hours of sleep time is reached. At all times, a device in Doze remains aware of motion and immediately leaves Doze if motion is detected.

https://source.android.com/devices/tech/power/platform_mgmt

92. The pending work described above – "syncs" and "jobs" – include application data requests, potentially including both upload and download of data. See, e.g., https://developer.android.com/reference/android/content/AbstractThreadedSyncAdapter; https://developer.android.com/reference/android/app/job/JobInfo; https://developer.android.com/topic/libraries/architecture/workmanager.

93. The Accused Products are configured to "enter a power save mode based on a battery level," as shown by the exemplary evidence below.



Screenshot from a Motorola Moto E (2020) device.

94. The Accused Products are configured to "while in the power save mode: alter behavior of sending application data requests from the application," as shown by the exemplary evidence below.

Battery saver improvements

Android 9 makes a number of improvements to battery saver mode. The device manufacturer determines the precise restrictions imposed. For example, on AOSP builds, the system applies the following restrictions:

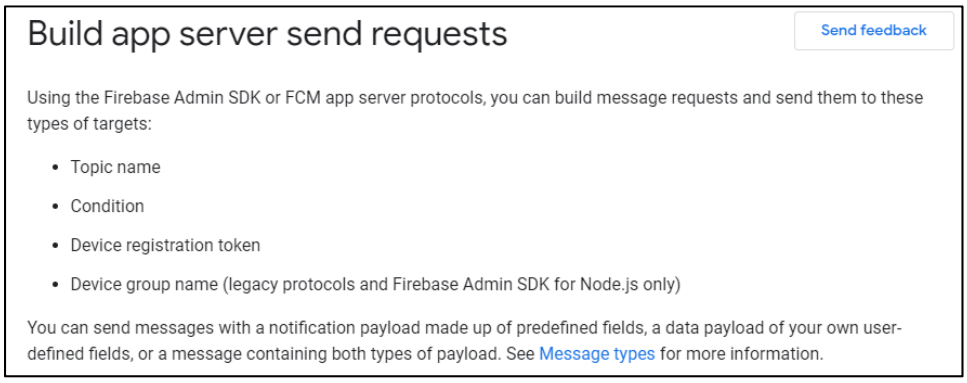
- The system puts apps in app standby mode more aggressively, instead of waiting for the app to be idle.
- Background execution limits apply to all apps, regardless of their target API level.
- Location services may be disabled when the screen is off.
- Background apps do not have network access.

In addition, there are other, device-specific power optimizations. For full details, see the page that describes power management restrictions.

As always, it's a good idea to test your app while battery saver is active. You can turn on battery saver manually through the device's **Settings > Battery Saver** screen.

https://developer.android.com/about/versions/pie/power

95. The Accused Products are configured to "receive a message over a connection with a second server when content received at the second server is new content that is associated with a predetermined identifier, wherein the message is related to the application sending application data requests," as shown by the exemplary evidence below.



Build app server send requests

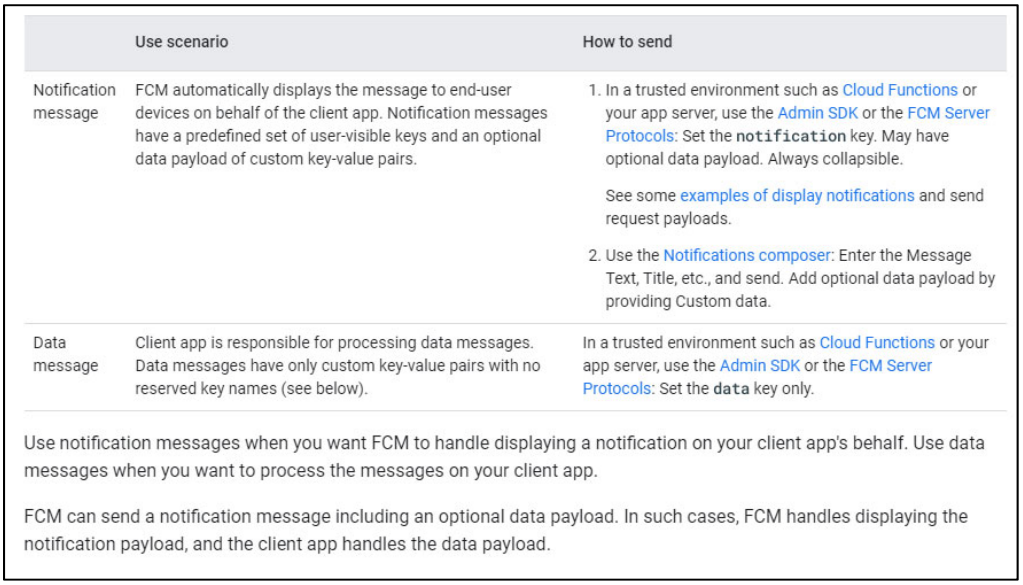
Send feedback

Using the Firebase Admin SDK or FCM app server protocols, you can build message requests and send them to these types of targets:

- Topic name
- Condition
- Device registration token
- Device group name (legacy protocols and Firebase Admin SDK for Node.js only)

You can send messages with a notification payload made up of predefined fields, a data payload of your own user-defined fields, or a message containing both types of payload. See Message types for more information.

https://firebase.google.com/docs/cloud-messaging/send-message



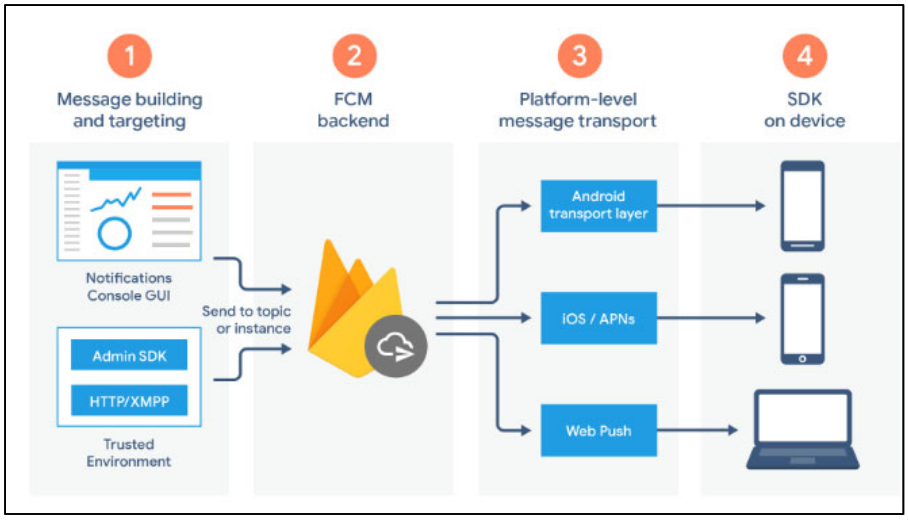
	Use scenario	How to send
Notification message	FCM automatically displays the message to end-user devices on behalf of the client app. Notification messages have a predefined set of user-visible keys and an optional data payload of custom key-value pairs.	1. In a trusted environment such as Cloud Functions or your app server, use the Admin SDK or the FCM Server Protocols: Set the `notification` key. May have optional data payload. Always collapsible. See some examples of display notifications and send request payloads. 2. Use the Notifications composer: Enter the Message Text, Title, etc., and send. Add optional data payload by providing Custom data.
Data message	Client app is responsible for processing data messages. Data messages have only custom key-value pairs with no reserved key names (see below).	In a trusted environment such as Cloud Functions or your app server, use the Admin SDK or the FCM Server Protocols: Set the `data` key only.

Use notification messages when you want FCM to handle displaying a notification on your client app's behalf. Use data messages when you want to process the messages on your client app.

FCM can send a notification message including an optional data payload. In such cases, FCM handles displaying the notification payload, and the client app handles the data payload.

https://firebase.google.com/docs/cloud-messaging/concept-options

96. The Accused Products are configured to "establish a subsequent connection with the first server to receive data related to the new content based on receipt of the message, wherein the second server is an intermediate server that is configured for providing connectivity between the first server and the mobile device," as shown by the exemplary evidence below.



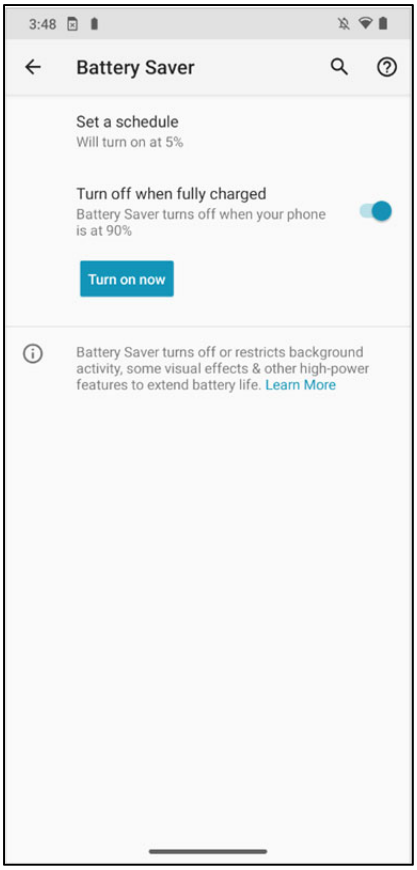
https://firebase.google.com/docs/cloud-messaging/fcm-architecture

Using FCM to interact with your app while the device is idle

Firebase Cloud Messaging (FCM) is a cloud-to-device service that lets you support real-time downstream messaging between backend services and apps on Android devices. FCM provides a single, persistent connection to the cloud; all apps needing real-time messaging can share this connection. This shared connection significantly optimizes battery consumption by making it unnecessary for multiple apps to maintain their own, separate persistent connections, which can deplete the battery rapidly. For this reason, if your app requires messaging integration with a backend service, we strongly recommend that you **use FCM if possible**, rather than maintaining your own persistent network connection.

https://developer.android.com/training/monitoring-device-state/doze-standby.html

97. The Accused Products are configured to enter "the power save mode" "when an amount of battery power remaining is below a predetermined amount," as shown by the exemplary evidence below.



Screenshot from a Motorola Moto E (2020) device.

98. The Accused Products are configured to "send login data to authenticate a user of the mobile device," as shown by the exemplary evidence below.

> **Add & use accounts on your device**
>
> To start downloading and buying items on the Google Play Store app, you need to add a Google Account on your device. You can repeat the steps below to add multiple accounts to your device.

https://support.google.com/googleplay/answer/2521798?hl=en

99. Thus, the Accused Products directly infringe one or more claims of the '457 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '457 patent.

100. Motorola indirectly infringes the '457 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-

users directly infringe through their use of the inventions claimed in the ’457 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola’s inducement, Motorola’s customers and end-users use the Motorola products in the way Motorola intends and directly infringe the ’457 patent. Motorola performs these affirmative acts with knowledge of the ’457 patent and with the intent, or willful blindness, that the induced acts directly infringe the ’457 patent. Motorola has had knowledge and notice of the ’457 patent at least as of the filing of this complaint.

101. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

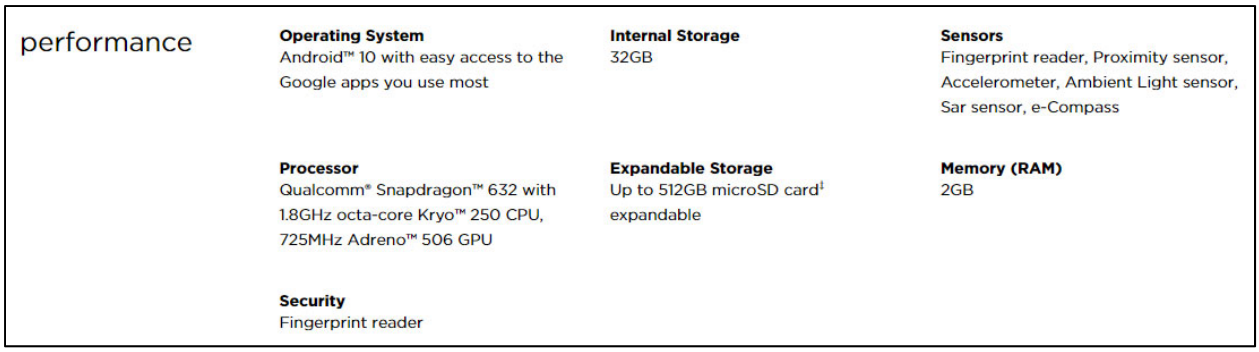
COUNT IV: PATENT INFRINGEMENT OF THE ’432 PATENT

102. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

103. Motorola infringes (literally and/or under the doctrine of equivalents) the ’432 patent by making, using, offering for sale, selling and/or importing into the United States

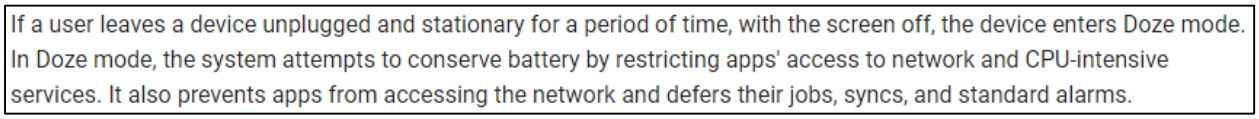
products and/or methods covered by one or more claims of the ’432 patent including at least the Accused Products.

104. For example and as shown below, the Accused Products infringe at least claim 1 of the ’432 patent. As shown by the exemplary evidence below, the Accused Products are “mobile devices” with “a memory and a processor.”



https://www.motorola.com/us/smartphones-moto-e/p

105. The Accused Products are configured for “entering a power save mode based on an idle status of a screen of the mobile device exceeding an amount of time,” as shown by the exemplary evidence below.



https://developer.android.com/training/monitoring-device-state/doze-standby

106. The Accused Products are configured for “while in the power save mode,” “suppressing transmission of application data requests during a first time period,” “allowing transmission of application data requests after expiration of the first time period,” and “suppressing transmission of application data requests during a second time period, wherein the second time period occurs after the first time period,” as shown by the exemplary evidence below.

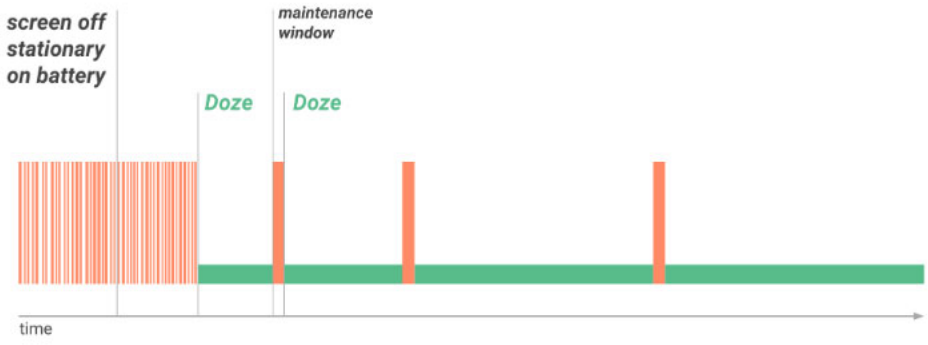


Figure 1. Doze provides a recurring maintenance window for apps to use the network and handle pending activities.

https://developer.android.com/training/monitoring-device-state/doze-standby

> Idle devices in Doze periodically enter a maintenance window, during which apps can complete pending work (syncs, jobs, etc.). Doze then resumes sleep for a longer period of time, followed by another maintenance window. The platform continues the Doze sleep/maintenance sequence, increasing the length of idle each time, until a maximum of a few hours of sleep time is reached. At all times, a device in Doze remains aware of motion and immediately leaves Doze if motion is detected.

https://source.android.com/devices/tech/power/platform_mgmt

107. The Accused Products are configured for "while in the power save mode," "maintaining a connection with a remote server during each of the first time period and the second time period for receipt of a message during either of the first time period and the second time periods" and "allowing transmission of an application data request in response to receipt of the message," as shown by the exemplary evidence below.

> ## Using FCM to interact with your app while the device is idle
>
> Firebase Cloud Messaging (FCM) is a cloud-to-device service that lets you support real-time downstream messaging between backend services and apps on Android devices. FCM provides a single, persistent connection to the cloud; all apps needing real-time messaging can share this connection. This shared connection significantly optimizes battery consumption by making it unnecessary for multiple apps to maintain their own, separate persistent connections, which can deplete the battery rapidly. For this reason, if your app requires messaging integration with a backend service, we strongly recommend that you **use FCM if possible**, rather than maintaining your own persistent network connection.
>
> FCM is optimized to work with Doze and App Standby idle modes by means of high-priority FCM messages. FCM high-priority messages let you reliably wake your app to access the network, even if the user's device is in Doze or the app is in App Standby mode. In Doze or App Standby mode, the system delivers the message and gives the app temporary access to network services and partial wakelocks, then returns the device or app to the idle state.
>
> High-priority FCM messages do not otherwise affect Doze mode, and they don't affect the state of any other app. This means that your app can use them to communicate efficiently while minimizing battery impacts across the system and device.

https://developer.android.com/training/monitoring-device-state/doze-standby

108. The Accused Products are configured for "while in the power save mode," "exiting the power save mode in response to detected user activity," as shown by the exemplary evidence below.

> As soon as the user wakes the device by moving it, turning on the screen, or connecting a charger, the system exits Doze and all apps return to normal activity.

https://developer.android.com/training/monitoring-device-state/doze-standby

109. The Accused Products are configured for "wherein the message indicates that new data is available at one of a first application server and a second application server associated with a respective first application and a second application executing on the mobile device," as shown by the exemplary evidence below.

> Using FCM, you can notify a client app that new email or other data is available to sync. You can send notification messages to drive user re-engagement and retention. For use cases such as instant messaging, a message can transfer a payload of up to 4KB to a client app.

https://firebase.google.com/docs/cloud-messaging/

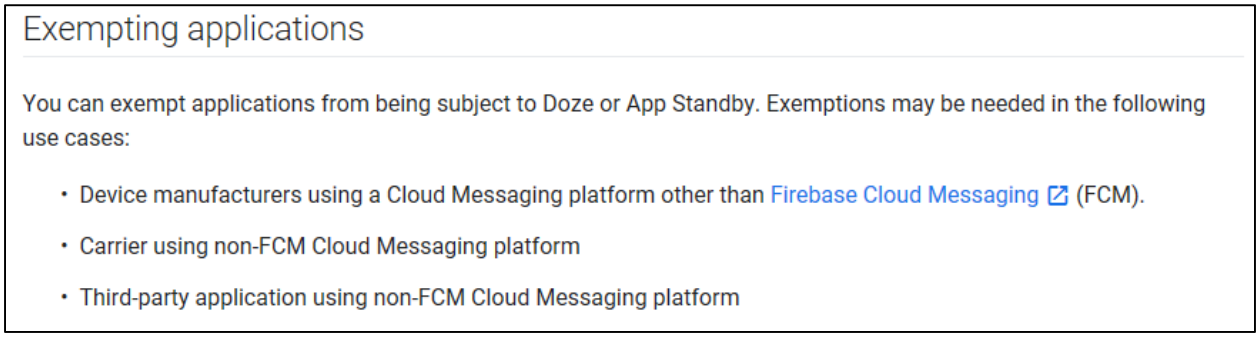
110. The Accused Products are configured for "wherein the remote server provides connectivity between the first application server and the mobile device, and between the second application server and the mobile device," as shown by the exemplary evidence below.

> Using FCM to interact with your app while the device is idle
>
> Firebase Cloud Messaging (FCM) is a cloud-to-device service that lets you support real-time downstream messaging between backend services and apps on Android devices. FCM provides a single, persistent connection to the cloud; all apps needing real-time messaging can share this connection. This shared connection significantly optimizes battery consumption by making it unnecessary for multiple apps to maintain their own, separate persistent connections, which can deplete the battery rapidly. For this reason, if your app requires messaging integration with a backend service, we strongly recommend that you **use FCM if possible**, rather than maintaining your own persistent network connection.

https://developer.android.com/training/monitoring-device-state/doze-standby.html

111. The Accused Products are configured for "wherein the suppressing is enabled and disabled for each of the first application and the second application based on a user selection for

each of the first application and the second application," as shown by the exemplary evidence below.

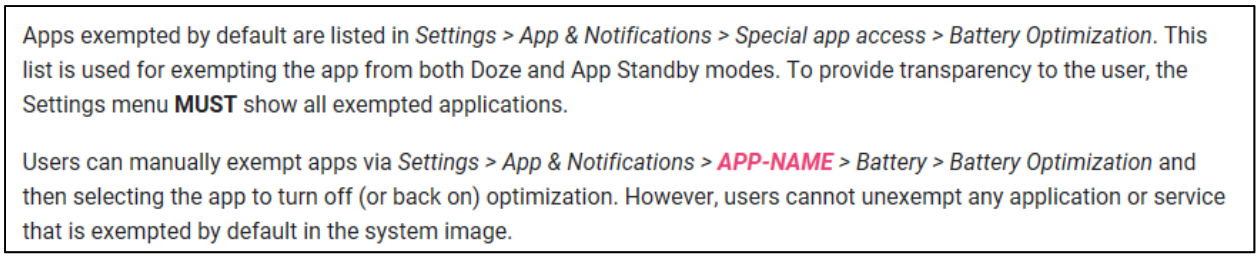


Exempting applications

You can exempt applications from being subject to Doze or App Standby. Exemptions may be needed in the following use cases:

- Device manufacturers using a Cloud Messaging platform other than Firebase Cloud Messaging (FCM).
- Carrier using non-FCM Cloud Messaging platform
- Third-party application using non-FCM Cloud Messaging platform

https://source.android.com/devices/tech/power/mgmt#exempt-apps



Apps exempted by default are listed in *Settings > App & Notifications > Special app access > Battery Optimization*. This list is used for exempting the app from both Doze and App Standby modes. To provide transparency to the user, the Settings menu **MUST** show all exempted applications.

Users can manually exempt apps via *Settings > App & Notifications > APP-NAME > Battery > Battery Optimization* and then selecting the app to turn off (or back on) optimization. However, users cannot unexempt any application or service that is exempted by default in the system image.

https://source.android.com/devices/tech/power/mgmt#exempt-apps

112. Thus, the Accused Products directly infringe one or more claims of the '432 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '432 patent.

113. Motorola indirectly infringes the '432 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-users directly infringe through their use of the inventions claimed in the '432 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing

materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola's inducement, Motorola's customers and end-users use the Motorola products in the way Motorola intends and directly infringe the '432 patent. Motorola performs these affirmative acts with knowledge of the '432 patent and with the intent, or willful blindness, that the induced acts directly infringe the '432 patent. Motorola has had knowledge and notice of the '432 patent at least as of the filing of this complaint.

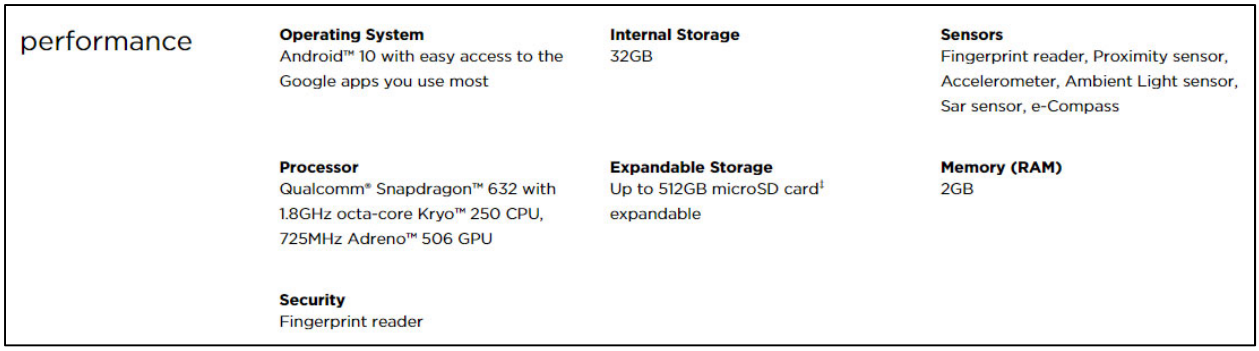
114. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

COUNT V: PATENT INFRINGEMENT OF THE '199 PATENT

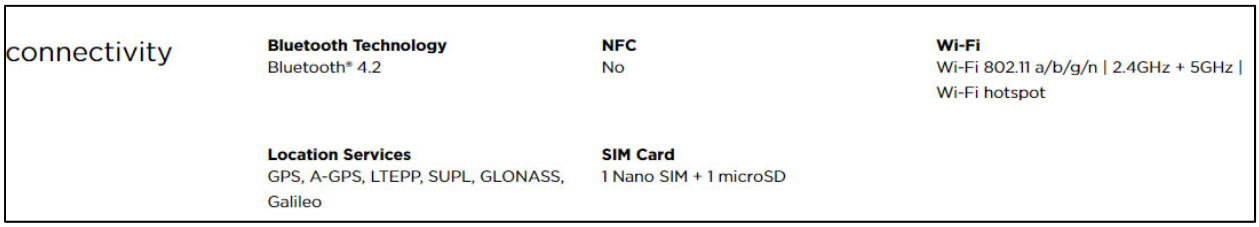
115. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

116. Motorola infringes (literally and/or under the doctrine of equivalents) the '199 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '199 patent including at least the Accused Products.

117. For example and as shown below, the Accused Products infringe at least claim 11 of the '199 patent. As shown by the exemplary evidence below, the Accused Products are "a mobile device" with "a memory" and "a processor."

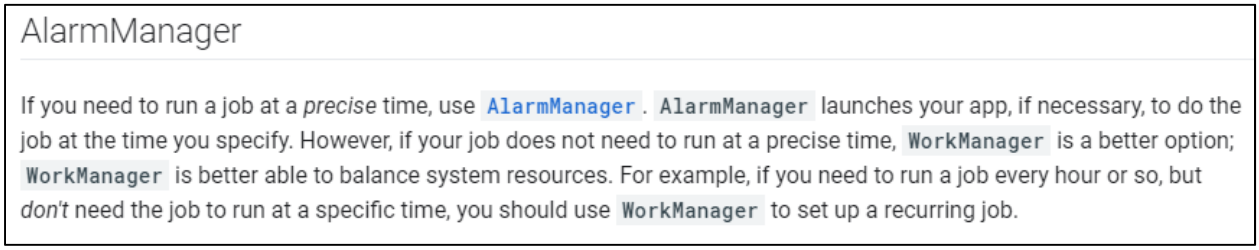


https://www.motorola.com/us/smartphones-moto-e/p



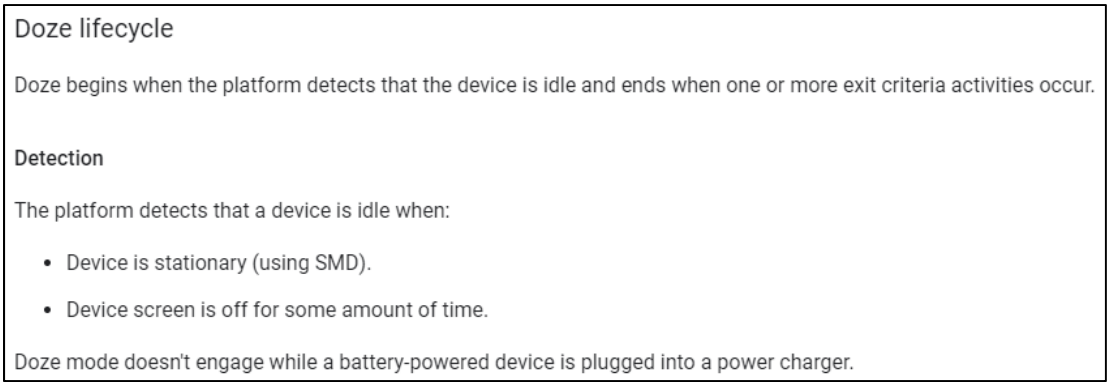
https://www.motorola.com/us/smartphones-moto-e/p

118. The Accused Products are configured for "executing alarms set by multiple applications executing on the mobile device," as shown by the exemplary evidence below.



https://developer.android.com/guide/background

119. The Accused Products are configured for "altering a behavior of the mobile device based on a detected activity status, wherein the detected activity status is based on a screen status and sensed motion of the mobile device," as shown by the exemplary evidence below.



Doze lifecycle

Doze begins when the platform detects that the device is idle and ends when one or more exit criteria activities occur.

Detection

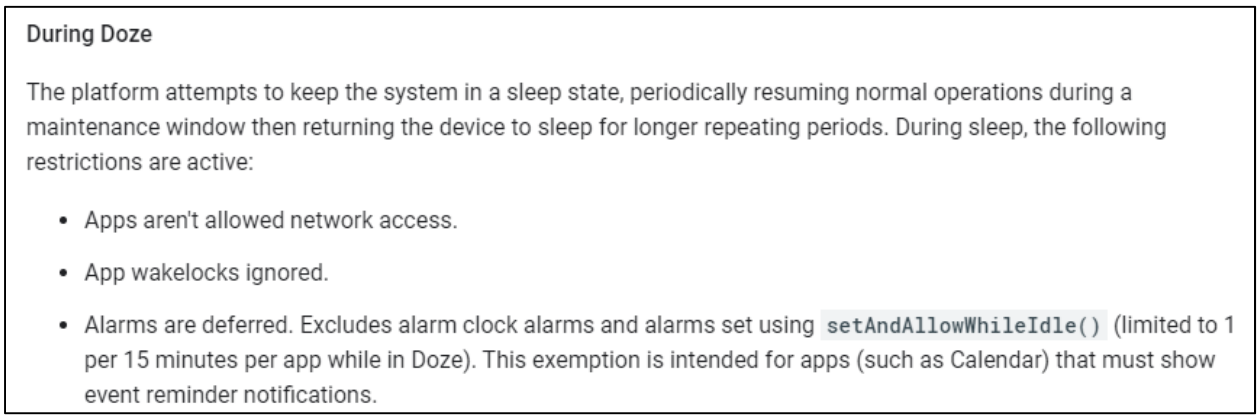
The platform detects that a device is idle when:

- Device is stationary (using SMD).
- Device screen is off for some amount of time.

Doze mode doesn't engage while a battery-powered device is plugged into a power charger.

https://source.android.com/devices/tech/power/platform_mgmt

120. The Accused Products are configured for "the altering behavior [to] include delaying a timing of one or more alarms for the multiple applications, wherein the timing is delayed such that the one or more delayed alarms execute within a window of time, and wherein at least a subset of the one or more delayed alarms are associated with one or more wakelocks," as shown by the exemplary evidence below.



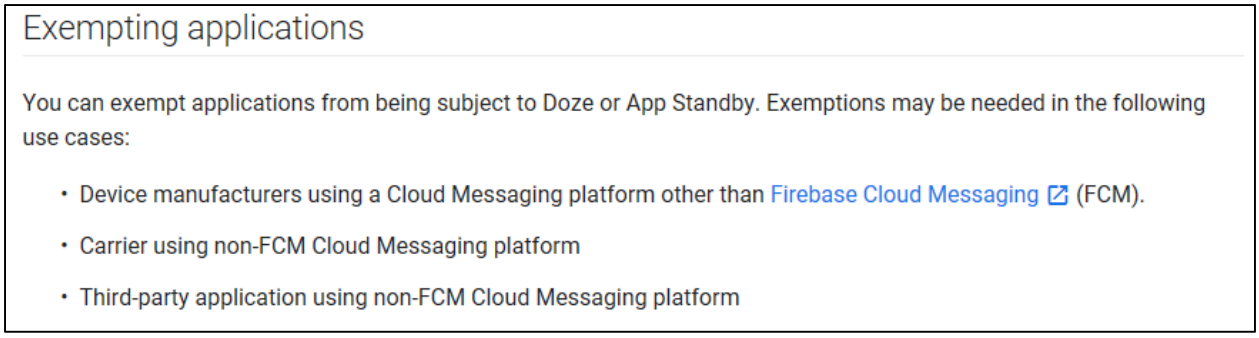
During Doze

The platform attempts to keep the system in a sleep state, periodically resuming normal operations during a maintenance window then returning the device to sleep for longer repeating periods. During sleep, the following restrictions are active:

- Apps aren't allowed network access.
- App wakelocks ignored.
- Alarms are deferred. Excludes alarm clock alarms and alarms set using `setAndAllowWhileIdle()` (limited to 1 per 15 minutes per app while in Doze). This exemption is intended for apps (such as Calendar) that must show event reminder notifications.

https://source.android.com/devices/tech/power/platform_mgmt

121. The Accused Products are configured for "wherein the altering behavior is based on application settings, wherein the application settings enable the altering behavior in response to a user selecting the application for the altering behavior and disable the altering behavior in response to a user deselecting the application for the altering behavior," as shown by the exemplary evidence below.

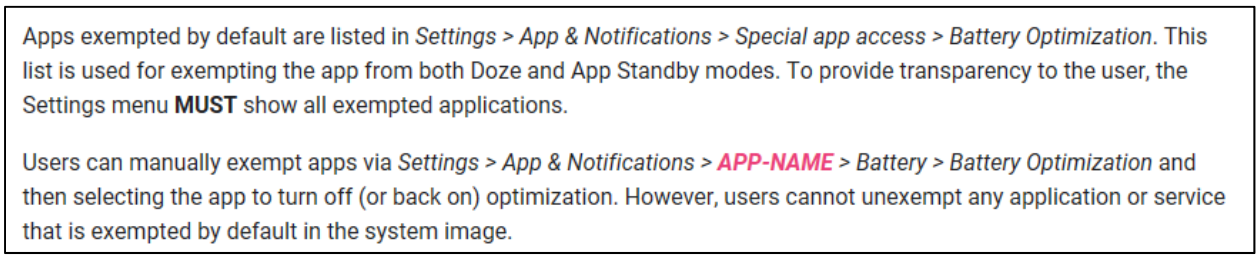


Exempting applications

You can exempt applications from being subject to Doze or App Standby. Exemptions may be needed in the following use cases:

- Device manufacturers using a Cloud Messaging platform other than Firebase Cloud Messaging (FCM).
- Carrier using non-FCM Cloud Messaging platform
- Third-party application using non-FCM Cloud Messaging platform

https://source.android.com/devices/tech/power/mgmt#exempt-apps



Apps exempted by default are listed in *Settings > App & Notifications > Special app access > Battery Optimization*. This list is used for exempting the app from both Doze and App Standby modes. To provide transparency to the user, the Settings menu **MUST** show all exempted applications.

Users can manually exempt apps via *Settings > App & Notifications > APP-NAME > Battery > Battery Optimization* and then selecting the app to turn off (or back on) optimization. However, users cannot unexempt any application or service that is exempted by default in the system image.

https://source.android.com/devices/tech/power/mgmt#exempt-apps

122. Thus, the Accused Products directly infringe one or more claims of the '199 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '199 patent.

123. Motorola indirectly infringes the '199 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-users directly infringe through their use of the inventions claimed in the '199 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary

evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola's inducement, Motorola's customers and end-users use the Motorola products in the way Motorola intends and directly infringe the '199 patent. Motorola performs these affirmative acts with knowledge of the '199 patent and with the intent, or willful blindness, that the induced acts directly infringe the '199 patent. Motorola has had knowledge and notice of the '199 patent at least as of the filing of this complaint.

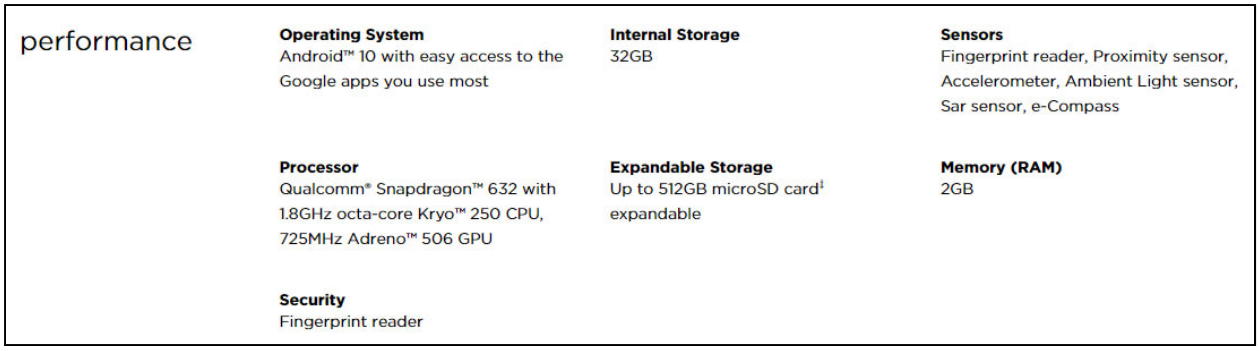
124. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

COUNT VI: PATENT INFRINGEMENT OF THE '161 PATENT

125. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

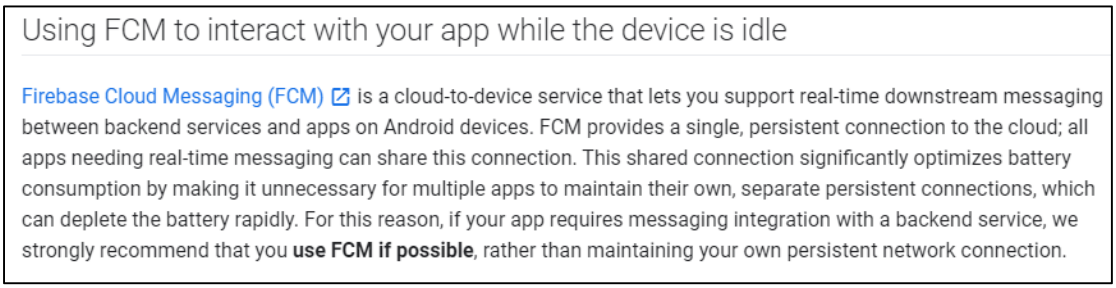
126. Motorola infringes (literally and/or under the doctrine of equivalents) the '161 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '161 patent including at least the Accused Products.

127. For example and as shown below, the Accused Products infringe at least claim 1 of the '161 patent. As shown by the exemplary evidence below, the Accused Products are "mobile[s] device having improved resource management" with "a memory" and "a processor."



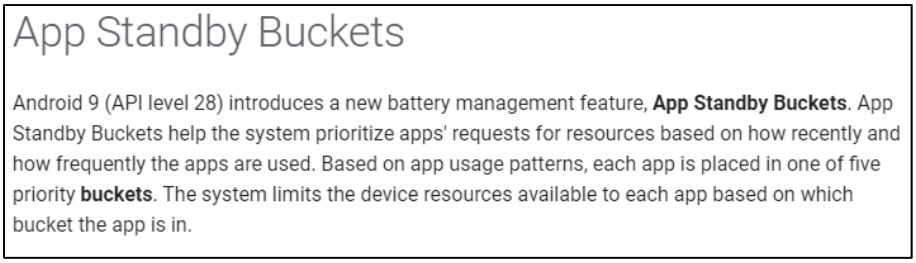
https://www.motorola.com/us/smartphones-moto-e/p (last visited 7/23/20)

128. The Accused Products are configured for "receiving a notification from a remote server directed towards an application indicating new data for receipt by the mobile device, wherein the remote server is an intermediary server that provides connectivity between the mobile device and a first application server and a second application server, wherein the notification is received over an established multiplexed connection," as shown in the exemplary documentation below.

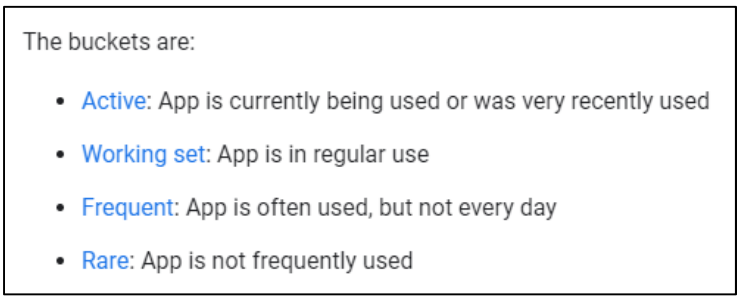


https://developer.android.com/training/monitoring-device-state/doze-standby.html

129. The Accused Products are configured for "predicting that a user is likely to access the application based on prior application access history, wherein the application is in a background of the mobile device," as shown by the exemplary documentation below.

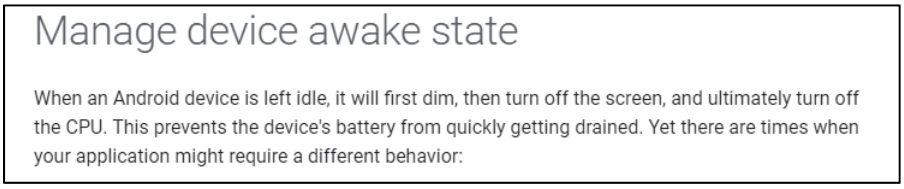


https://developer.android.com/topic/performance/appstandby



https://developer.android.com/topic/performance/appstandby

130. The Accused Products are configured for "wherein the user of the mobile device is inactive and a screen status of the mobile device is off," as shown by the exemplary documentation below.



https://developer.android.com/training/scheduling

131. The Accused Products are configured for "wherein a second connection is established while the established multiplexed connection remains connected, wherein the second connection is other than the established multiplexed connection," as shown by the exemplary documentation below.

A URLConnection with support for HTTP-specific features. See the spec for details.

Uses of this class follow a pattern:

1. Obtain a new `HttpURLConnection` by calling `URL#openConnection()` and casting the result to `HttpURLConnection`.
2. Prepare the request. The primary property of a request is its URI. Request headers may also include metadata such as credentials, preferred content types, and session cookies.
3. Optionally upload a request body. Instances must be configured with `setDoOutput(true)` if they include a request body. Transmit data by writing to the stream returned by `URLConnection.getOutputStream()`.
4. Read the response. Response headers typically include metadata such as the response body's content type and length, modified dates and session cookies. The response body may be read from the stream returned by `URLConnection.getInputStream()`. If the response has no body, that method returns an empty stream.
5. Disconnect. Once the response body has been read, the `HttpURLConnection` should be closed by calling `disconnect()`. Disconnecting releases the resources held by a connection so they may be closed or reused.

https://developer.android.com/reference/java/net/HttpURLConnection

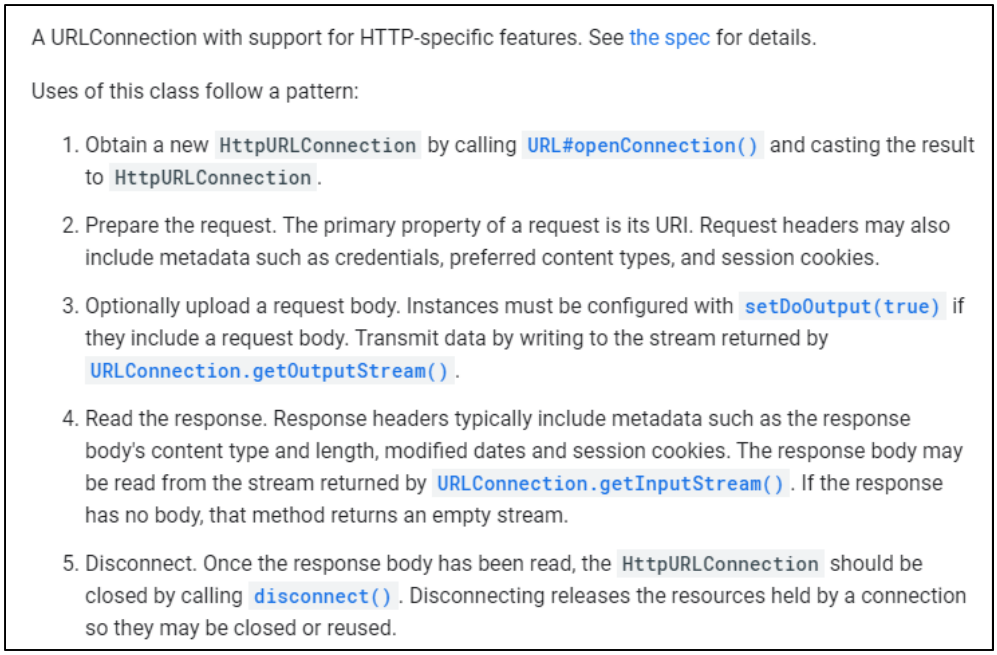
132. The Accused Products are configured for "wherein data for the application is fetched based on the prediction, wherein data for the application is fetched over the second connection before the application is accessed," as shown by the exemplary documentation below.

App Standby Buckets

Android 9 (API level 28) introduces a new battery management feature, **App Standby Buckets**. App Standby Buckets help the system prioritize apps' requests for resources based on how recently and how frequently the apps are used. Based on app usage patterns, each app is placed in one of five priority **buckets**. The system limits the device resources available to each app based on which bucket the app is in.

https://developer.android.com/topic/performance/appstandby

133. The Accused Products are configured for "wherein the fetched data is for background requests made by the application on the mobile device, wherein the fetched data is delivered to the application, and disconnecting the second connection," as shown by the exemplary documentation below.



A URLConnection with support for HTTP-specific features. See the spec for details.

Uses of this class follow a pattern:

1. Obtain a new `HttpURLConnection` by calling `URL#openConnection()` and casting the result to `HttpURLConnection`.
2. Prepare the request. The primary property of a request is its URI. Request headers may also include metadata such as credentials, preferred content types, and session cookies.
3. Optionally upload a request body. Instances must be configured with `setDoOutput(true)` if they include a request body. Transmit data by writing to the stream returned by `URLConnection.getOutputStream()`.
4. Read the response. Response headers typically include metadata such as the response body's content type and length, modified dates and session cookies. The response body may be read from the stream returned by `URLConnection.getInputStream()`. If the response has no body, that method returns an empty stream.
5. Disconnect. Once the response body has been read, the `HttpURLConnection` should be closed by calling `disconnect()`. Disconnecting releases the resources held by a connection so they may be closed or reused.

https://developer.android.com/reference/java/net/HttpURLConnection

134. Thus, the Accused Products directly infringe one or more claims of the '161 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '161 patent.

135. Motorola indirectly infringes the '161 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-users directly infringe through their use of the inventions claimed in the '161 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary

evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola's inducement, Motorola's customers and end-users use the Motorola products in the way Motorola intends and directly infringe the '161 patent. Motorola performs these affirmative acts with knowledge of the '161 patent and with the intent, or willful blindness, that the induced acts directly infringe the '161 patent. Motorola has had knowledge and notice of the '161 patent at least as of the filing of this complaint.

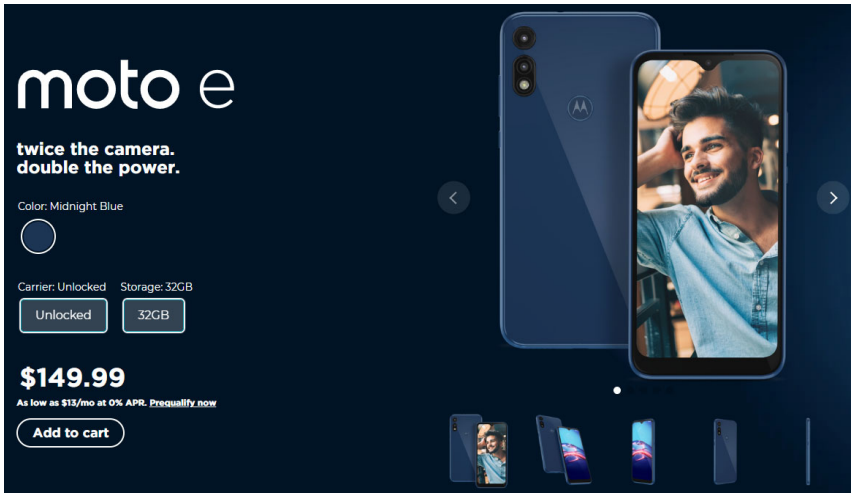
136. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

COUNT VII: PATENT INFRINGEMENT OF THE '339 PATENT

137. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

138. Motorola infringes (literally and/or under the doctrine of equivalents) the '339 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '339 patent including at least the Accused Products.

139. For example and as shown below, the Accused Products infringe at least claim 1 of the '339 patent. As shown by the exemplary evidence below, Accused Products are "mobile device[s] configured to manage connections made by the mobile device in a wireless network" with "a memory; a radio; a screen; and a processor."



https://www.motorola.com/us/smartphones-moto-e/p

performance			
	Operating System Android™ 10 with easy access to the Google apps you use most	**Internal Storage** 32GB	**Sensors** Fingerprint reader, Proximity sensor, Accelerometer, Ambient Light sensor, Sar sensor, e-Compass
	Processor Qualcomm® Snapdragon™ 632 with 1.8GHz octa-core Kryo™ 250 CPU, 725MHz Adreno™ 506 GPU	**Expandable Storage** Up to 512GB microSD card[‡] expandable	**Memory (RAM)** 2GB
	Security Fingerprint reader		

https://www.motorola.com/us/smartphones-moto-e/p

connectivity			
	Bluetooth Technology Bluetooth® 4.2	**NFC** No	**Wi-Fi** Wi-Fi 802.11 a/b/g/n | 2.4GHz + 5GHz | Wi-Fi hotspot
	Location Services GPS, A-GPS, LTEPP, SUPL, GLONASS, Galileo	**SIM Card** 1 Nano SIM + 1 microSD	

https://www.motorola.com/us/smartphones-moto-e/p

140. The Accused Products are configured to "delay a content request from a first application for transmission to a first application server over the wireless network, wherein the delayed content request from the first application includes a background data request" and "delay a content request from a second application for transmission to a second application server over the wireless network wherein the delayed content request from the second application includes a background data request," "wherein, the delayed content request from the first application and

the delayed content request from the second application are delayed while the mobile device screen is idle in response to inactivity of the mobile device," as shown in the exemplary documentation below.

> The framework will be intelligent about when it executes jobs, and attempt to batch and defer them as much as possible. Typically if you don't specify a deadline on a job, it can be run at any moment depending on the current state of the JobScheduler's internal queue.

https://developer.android.com/reference/android/app/job/JobScheduler

> Understanding Doze
>
> If a user leaves a device unplugged and stationary for a period of time, with the screen off, the device enters Doze mode. In Doze mode, the system attempts to conserve battery by restricting apps' access to network and CPU-intensive services. It also prevents apps from accessing the network and defers their jobs, syncs, and standard alarms.
>
> Periodically, the system exits Doze for a brief time to let apps complete their deferred activities. During this *maintenance window*, the system runs all pending syncs, jobs, and alarms, and lets apps access the network.

https://developer.android.com/training/monitoring-device-state/doze-standby

141. The Accused Products are configured to "allow a first message from a remote server distinct from the first application server and the second application server to be received while the content request from the first application and the content request from the second application are delayed" "wherein the first message from the remote server is directed to the first application and contains data from the first application server and is associated with the mobile device and the first application, wherein the first message informs the first application that there is new data to be fetched from the first application server," as shown by the exemplary documentation below.

> FCM is optimized to work with Doze and App Standby idle modes by means of high-priority FCM messages. FCM high-priority messages let you reliably wake your app to access the network, even if the user's device is in Doze or the app is in App Standby mode. In Doze or App Standby mode, the system delivers the message and gives the app temporary access to network services and partial wakelocks, then returns the device or app to the idle state.

https://developer.android.com/training/monitoring-device-state/doze-standby#using_fcm

142. The Accused Products are configured to "transmit a second message associated with the first application to the remote server or the first application server in response to receipt of the first message from the remote server," as shown by the exemplary documentation below.

> FCM is optimized to work with Doze and App Standby idle modes by means of high-priority FCM messages. FCM high-priority messages let you reliably wake your app to access the network, even if the user's device is in Doze or the app is in App Standby mode. In Doze or App Standby mode, the system delivers the message and gives the app temporary access to network services and partial wakelocks, then returns the device or app to the idle state.

https://developer.android.com/training/monitoring-device-state/doze-standby#using_fcm

> Understanding Doze
>
> If a user leaves a device unplugged and stationary for a period of time, with the screen off, the device enters Doze mode. In Doze mode, the system attempts to conserve battery by restricting apps' access to network and CPU-intensive services. It also prevents apps from accessing the network and defers their jobs, syncs, and standard alarms.
>
> Periodically, the system exits Doze for a brief time to let apps complete their deferred activities. During this *maintenance window*, the system runs all pending syncs, jobs, and alarms, and lets apps access the network.

https://developer.android.com/training/monitoring-device-state/doze-standby

143. The Accused Products are configured to "transmit the delayed content request from the first application to the first application server over the wireless network and the delayed content request from the second application to the second application server over the wireless network while the mobile device screen remains idle, wherein transmitting the delayed content request from the first application and the delayed content request from the second application occurs at about the same time," as shown by the exemplary documentation below.

> The framework will be intelligent about when it executes jobs, and attempt to batch and defer them as much as possible. Typically if you don't specify a deadline on a job, it can be run at any moment depending on the current state of the JobScheduler's internal queue.

https://developer.android.com/reference/android/app/job/JobScheduler

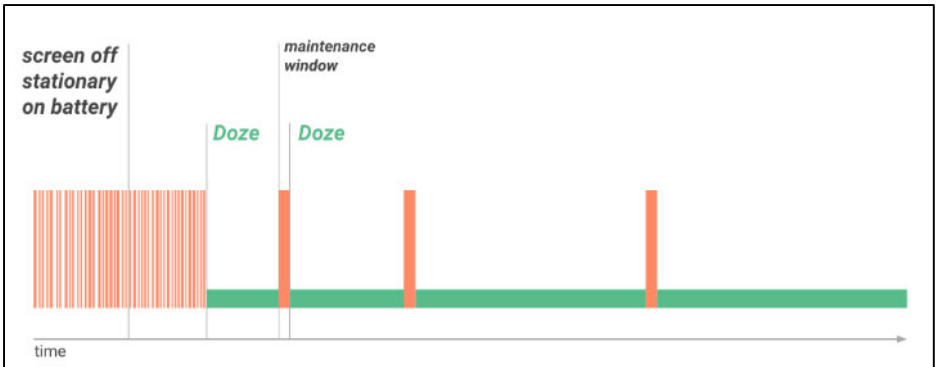


Figure 1. Doze provides a recurring maintenance window for apps to use the network and handle pending activities.

https://developer.android.com/training/monitoring-device-state/doze-standby

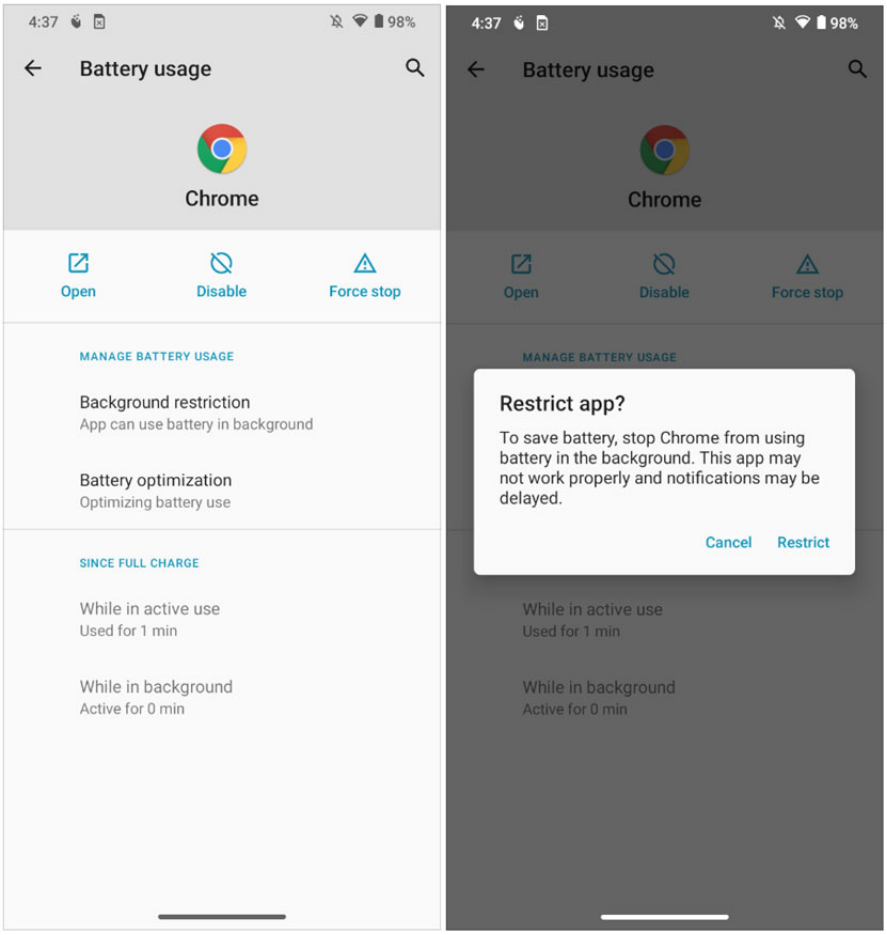
Understanding Doze

If a user leaves a device unplugged and stationary for a period of time, with the screen off, the device enters Doze mode. In Doze mode, the system attempts to conserve battery by restricting apps' access to network and CPU-intensive services. It also prevents apps from accessing the network and defers their jobs, syncs, and standard alarms.

Periodically, the system exits Doze for a brief time to let apps complete their deferred activities. During this *maintenance window*, the system runs all pending syncs, jobs, and alarms, and lets apps access the network.

https://developer.android.com/training/monitoring-device-state/doze-standby

144. The Accused Products are configured to "wherein the delaying of the content request for the second application can be enabled or disabled by a user of the mobile device on an application-by-application basis," as shown by the exemplary documentation below.



Screenshot from a Motorola Moto E (2020) device.

145. Thus, the Accused Products directly infringe one or more claims of the '339 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '339 patent.

146. Motorola indirectly infringes the '339 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-users directly infringe through their use of the inventions claimed in the '339 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they

use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola's inducement, Motorola's customers and end-users use the Motorola products in the way Motorola intends and directly infringe the '339 patent. Motorola performs these affirmative acts with knowledge of the '339 patent and with the intent, or willful blindness, that the induced acts directly infringe the '339 patent. Motorola has had knowledge and notice of the '339 patent at least as of the filing of this complaint.

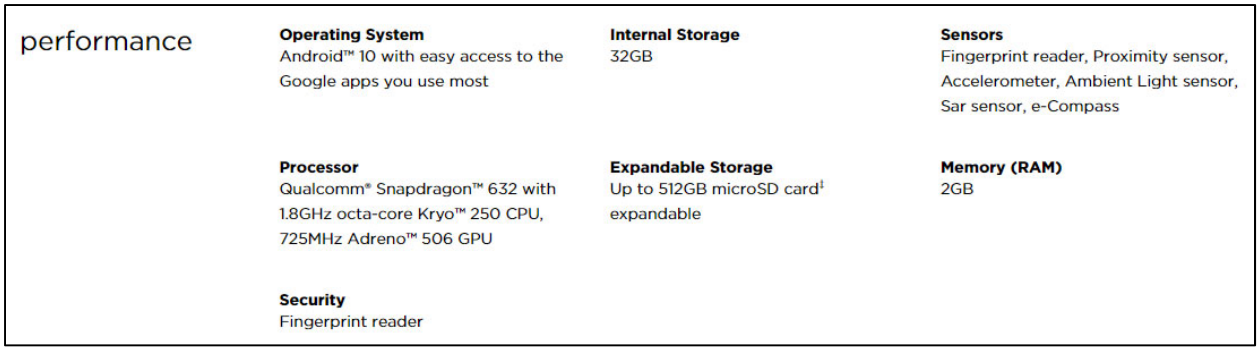
147. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

COUNT VIII: PATENT INFRINGEMENT OF THE '228 PATENT

148. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

149. Motorola infringes (literally and/or under the doctrine of equivalents) the '228 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '228 patent including at least the Accused Products.

150. For example and as shown below, the Accused Products infringe at least claim 1 of the '228 patent. As shown by the exemplary evidence below, the Accused Products are "mobile devices" with "a memory, and a processor."



performance

Operating System
Android™ 10 with easy access to the Google apps you use most

Internal Storage
32GB

Sensors
Fingerprint reader, Proximity sensor, Accelerometer, Ambient Light sensor, Sar sensor, e-Compass

Processor
Qualcomm® Snapdragon™ 632 with 1.8GHz octa-core Kryo™ 250 CPU, 725MHz Adreno™ 506 GPU

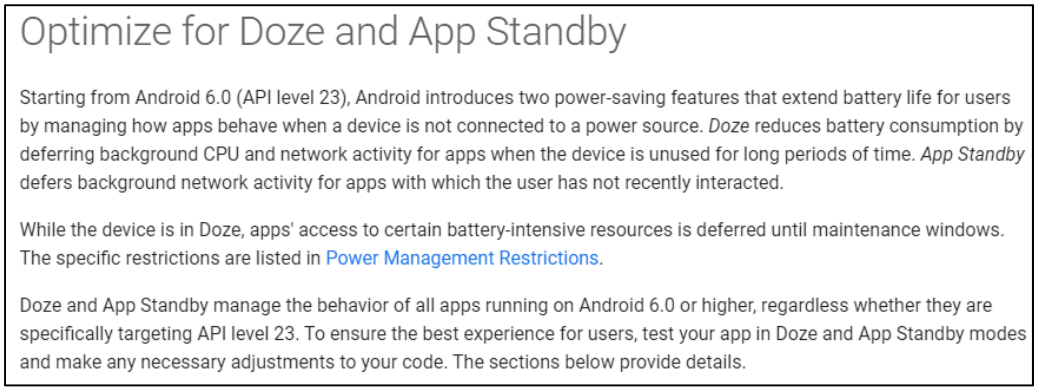
Expandable Storage
Up to 512GB microSD card[‡] expandable

Memory (RAM)
2GB

Security
Fingerprint reader

https://www.motorola.com/us/smartphones-moto-e/p

151. The Accused Products are configured for "adjusting a timing of background application data requests in order to conserve battery" "wherein adjusting a timing is enabled automatically by the mobile device on an application by application basis," as shown by the exemplary evidence below.



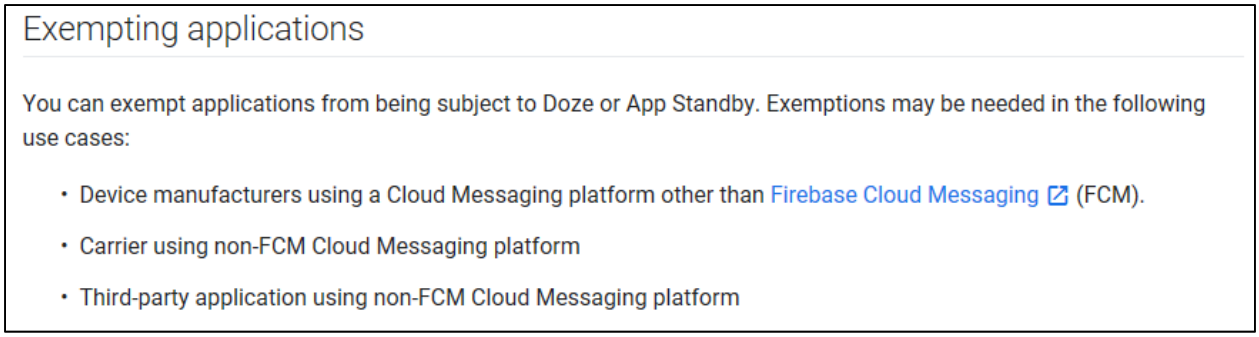
Optimize for Doze and App Standby

Starting from Android 6.0 (API level 23), Android introduces two power-saving features that extend battery life for users by managing how apps behave when a device is not connected to a power source. *Doze* reduces battery consumption by deferring background CPU and network activity for apps when the device is unused for long periods of time. *App Standby* defers background network activity for apps with which the user has not recently interacted.

While the device is in Doze, apps' access to certain battery-intensive resources is deferred until maintenance windows. The specific restrictions are listed in Power Management Restrictions.

Doze and App Standby manage the behavior of all apps running on Android 6.0 or higher, regardless whether they are specifically targeting API level 23. To ensure the best experience for users, test your app in Doze and App Standby modes and make any necessary adjustments to your code. The sections below provide details.

https://developer.android.com/training/monitoring-device-state/doze-standby



Exempting applications

You can exempt applications from being subject to Doze or App Standby. Exemptions may be needed in the following use cases:

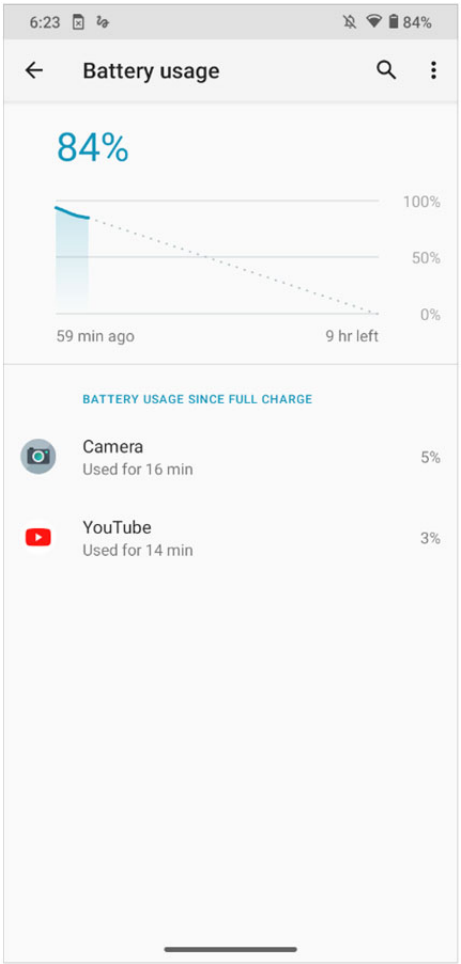
- Device manufacturers using a Cloud Messaging platform other than Firebase Cloud Messaging (FCM).
- Carrier using non-FCM Cloud Messaging platform
- Third-party application using non-FCM Cloud Messaging platform

https://source.android.com/devices/tech/power/mgmt#exempt-apps

> Apps exempted by default are listed in *Settings > App & Notifications > Special app access > Battery Optimization*. This list is used for exempting the app from both Doze and App Standby modes. To provide transparency to the user, the Settings menu **MUST** show all exempted applications.
>
> Users can manually exempt apps via *Settings > App & Notifications > **APP-NAME** > Battery > Battery Optimization* and then selecting the app to turn off (or back on) optimization. However, users cannot unexempt any application or service that is exempted by default in the system image.

https://source.android.com/devices/tech/power/mgmt#exempt-apps

152. The Accused Products are configured for "monitoring battery resource consumption of multiple applications operating on a mobile device," as shown by the exemplary evidence below.



Screenshot from a Motorola Moto E (2020) device.

153. The Accused Products are configured for "monitoring a time of use of the multiple applications operating on the mobile device," as shown by the exemplary evidence below.

UsageStatsManager

Added in API level 21

Kotlin | **Java**

```
public final class UsageStatsManager
extends Object
```

java.lang.Object
↳ android.app.usage.UsageStatsManager

Provides access to device usage history and statistics. Usage data is aggregated into time intervals: days, weeks, months, and years.

When requesting usage data since a particular time, the request might look something like this:

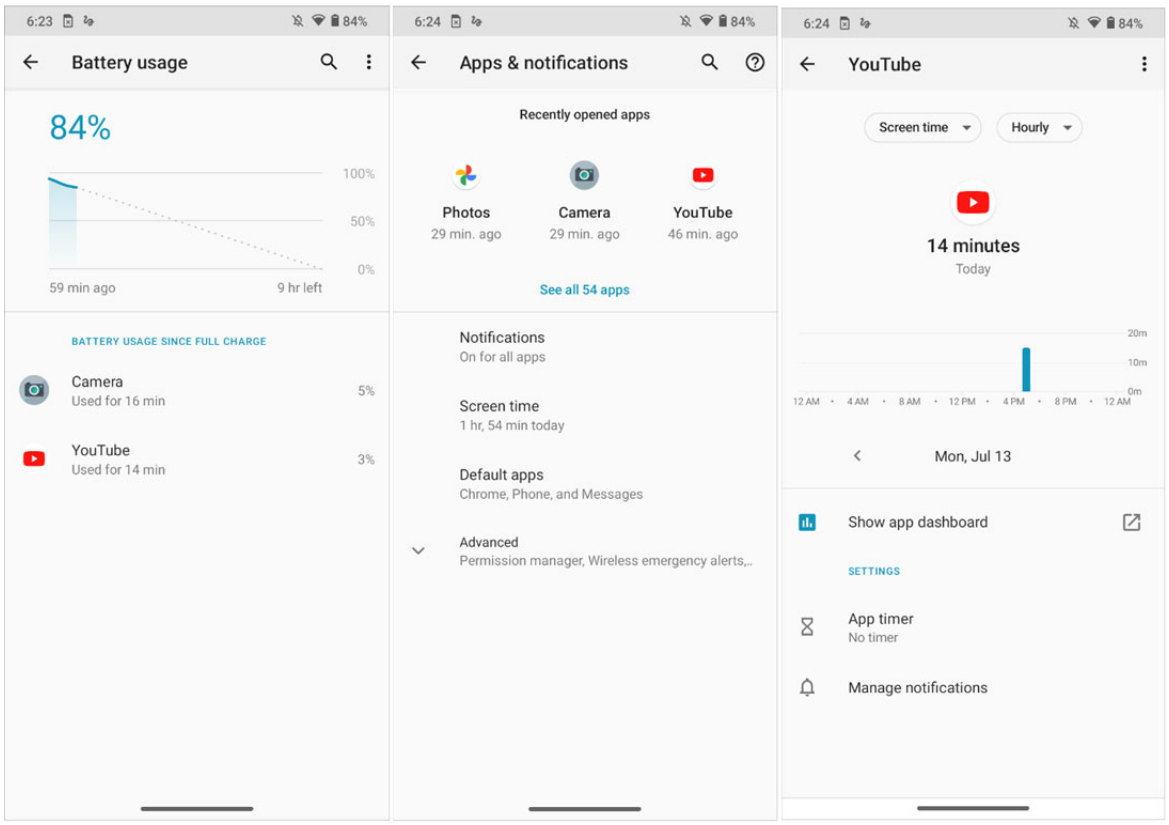
```
PAST                   REQUEST_TIME                    TODAY                   FUTURE
----------------------------||---------------------------¦-----------------------|
                        YEAR ||                           ¦                       |
----------------------------||---------------------------¦-----------------------|
 MONTH            |         ||                MONTH      ¦                       |
------------------|---------||---------------------------¦-----------------------|
  |      WEEK     |     WEEK||    |     WEEK     |     WE¦EK     |      WEEK     |
----------------------------||-------------------|-------¦-----------------------|
                            ||           |DAY|DAY|DAY|DAY¦DAY|DAY|DAY|DAY|DAY|DAY|
----------------------------||---------------------------¦-----------------------|
```

A request for data in the middle of a time interval will include that interval.

NOTE: Most methods on this API require the permission android.permission.PACKAGE_USAGE_STATS. However, declaring the permission implies intention to use the API and the user of the device still needs to grant permission through the Settings application. See `Settings.ACTION_USAGE_ACCESS_SETTINGS`. Methods which only return the information for the calling package do not require this permission. E.g. `getAppStandbyBucket()` and `queryEventsForSelf(long, long)`.

https://developer.android.com/reference/android/app/usage/UsageStatsManager

154. The Accused Products are configured for "wherein a first report displays proportional battery consumption of each of a plurality of the multiple applications operating on the mobile device, wherein the displayed proportional battery consumption includes: battery resource consumption associated with background application data requests; and battery resource consumption that is expressed as a battery percentage representing a proportional battery consumption of each of the plurality of the multiple applications over a time period" "wherein a second report displays the time of use of the multiple applications on the mobile device," as shown by the exemplary evidence below.



Screenshots from a Motorola Moto E (2020) device.

> Battery use statistics
>
> The framework automatically determines battery use statistics by tracking how long device components spend in different states. As components (Wi-Fi chipset, cellular radio, Bluetooth, GPS, display, CPU) change states (OFF/ON, idle/full power, low/high brightness, etc.), the controlling service reports to the framework BatteryStats service. BatteryStats collects information over time and stores it for use across reboots. The service doesn't track battery current draw directly, but instead collects timing information that can be used to approximate battery consumption by different components.
>
> The framework gathers statistics using the following methods:
>
> - **Push**. Services aware of component changes push state changes to the BatteryStats service.
> - **Pull**. For components such as the CPU use by apps, the framework automatically pulls the data at transition points (such as starting or stopping an activity) to take a snapshot.
>
> Resource consumption is associated with the application using the resource. When multiple applications simultaneously use a resource (such as wakelocks that prevent the system from suspending), the framework spreads consumption across those applications, although not necessarily equally.

https://source.android.com/devices/tech/power

155. Thus, the Accused Products directly infringe one or more claims of the '228 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '228 patent.

156. Motorola indirectly infringes the ’228 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola’s customers and end-users, in this district and elsewhere in the United States. For example, Motorola’s customers and end-users directly infringe through their use of the inventions claimed in the ’228 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola’s inducement, Motorola’s customers and end-users use the Motorola products in the way Motorola intends and directly infringe the ’228 patent. Motorola performs these affirmative acts with knowledge of the ’228 patent and with the intent, or willful blindness, that the induced acts directly infringe the ’228 patent. Motorola has had knowledge and notice of the ’228 patent at least as of the filing of this complaint.

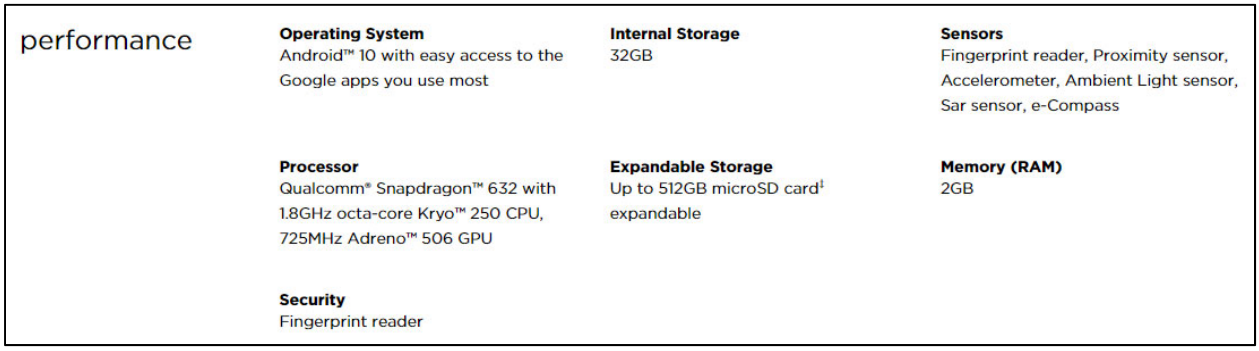
157. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

COUNT IX: PATENT INFRINGEMENT OF THE ’127 PATENT

158. Plaintiff incorporates by reference the preceding paragraphs as though fully set forth herein.

159. Motorola infringes (literally and/or under the doctrine of equivalents) the '127 patent by making, using, offering for sale, selling and/or importing into the United States products and/or methods covered by one or more claims of the '127 patent including at least the Accused Products.

160. For example and as shown below, the Accused Products infringe at least claim 10 of the '127 patent. As shown by the exemplary evidence below, the Accused Products are "mobile devices" with "a memory" and "a processor in communication with the memory and configured to execute instructions stored in the memory."



performance

Operating System
Android™ 10 with easy access to the Google apps you use most

Internal Storage
32GB

Sensors
Fingerprint reader, Proximity sensor, Accelerometer, Ambient Light sensor, Sar sensor, e-Compass

Processor
Qualcomm® Snapdragon™ 632 with 1.8GHz octa-core Kryo™ 250 CPU, 725MHz Adreno™ 506 GPU

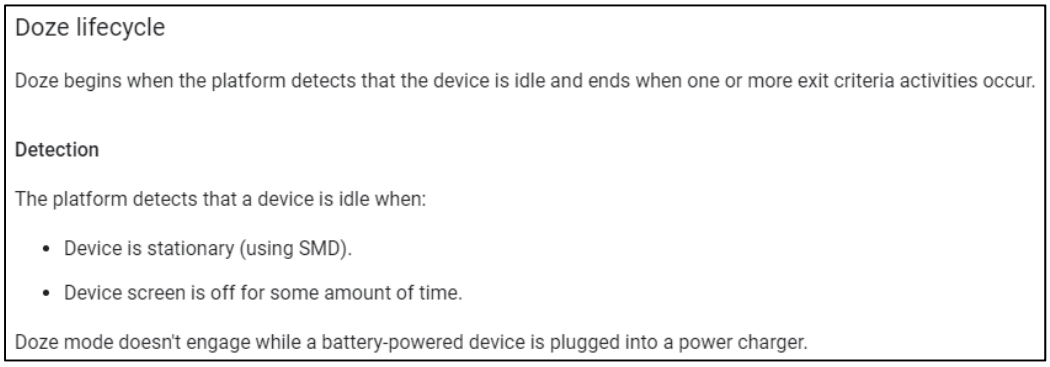
Expandable Storage
Up to 512GB microSD card[‡] expandable

Memory (RAM)
2GB

Security
Fingerprint reader

https://www.motorola.com/us/smartphones-moto-e/p

161. The Accused Products are configured to "enter a power save mode based on a backlight status and sensed motion of a mobile device," as shown by the exemplary evidence below.



Doze lifecycle

Doze begins when the platform detects that the device is idle and ends when one or more exit criteria activities occur.

Detection

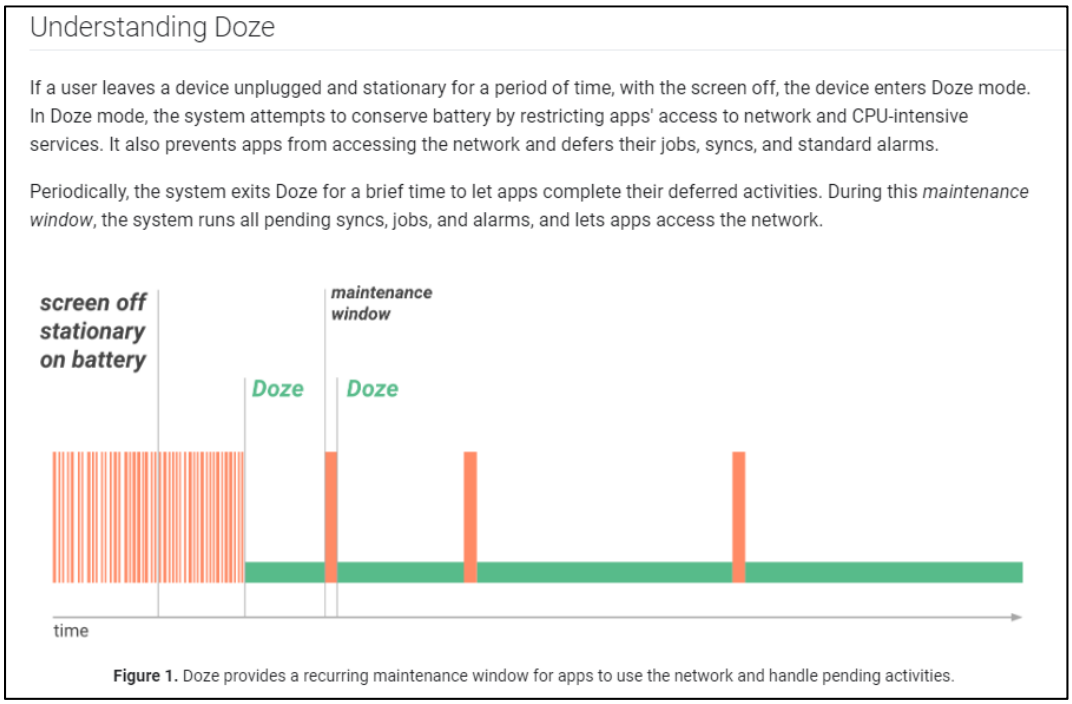
The platform detects that a device is idle when:

- Device is stationary (using SMD).
- Device screen is off for some amount of time.

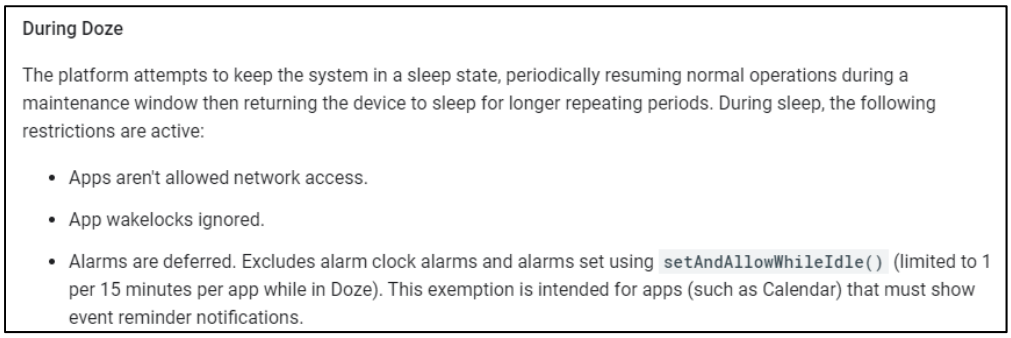
Doze mode doesn't engage while a battery-powered device is plugged into a power charger.

https://source.android.com/devices/tech/power/platform_mgmt

162. The Accused Products are configured to "delay a timing of one or more triggers for multiple applications on the mobile device, wherein the timing is delayed such that the triggers execute within a window of time, wherein at least a subset of the triggers are associated with wakelocks," as shown by the exemplary evidence below.

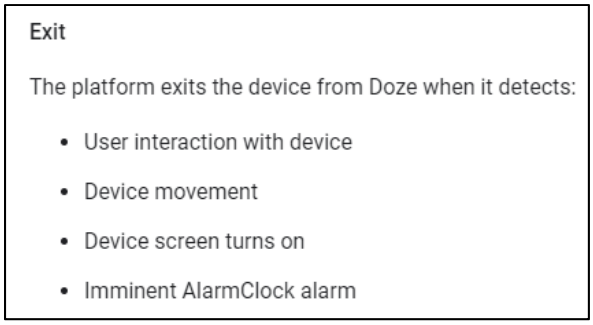


https://developer.android.com/training/monitoring-device-state/doze-standby



https://source.android.com/devices/tech/power/platform_mgmt

163. The Accused Products are configured to "exit the power save mode when the backlight of the mobile device turns on or motion of the mobile device is sensed," as shown by the exemplary evidence below.



https://source.android.com/devices/tech/power/platform_mgmt

164. Thus, the Accused Products directly infringe one or more claims of the '127 patent. Motorola makes, uses, sells, offers for sale, and/or imports, in this district and/or elsewhere in the United States, these devices and thus directly infringes the '127 patent.

165. Motorola indirectly infringes the '127 patent, as provided in 35 U.S.C. § 271(b), including by inducing infringement by others, such as Motorola's customers and end-users, in this district and elsewhere in the United States. For example, Motorola's customers and end-users directly infringe through their use of the inventions claimed in the '127 patent. Motorola induces this direct infringement through its affirmative acts of manufacturing, selling, distributing, repairing, and/or otherwise making available the Accused Products, and providing instructions, documentation, and other information to customers and end-users suggesting they use the Accused Products in an infringing manner, including in-store technical support, online technical support, marketing, product manuals, advertisements, online documentation, marketing materials, technical specifications, data sheets, web pages on its website, press releases, user manuals, and trade shows, including the Motorola documentation cited herein as exemplary evidence of infringement. By way of example, Motorola user manuals and documentation (including those cited herein) instruct, promote, and encourage the use of the Accused Products in an infringing manner. As a result of Motorola's inducement, Motorola's customers and end-users use the Motorola products in the way Motorola intends and directly infringe the '127

patent. Motorola performs these affirmative acts with knowledge of the '127 patent and with the intent, or willful blindness, that the induced acts directly infringe the '127 patent. Motorola has had knowledge and notice of the '127 patent at least as of the filing of this complaint.

166. Motorola, by way of its infringing activities, has caused and continues to cause SEVEN to suffer damages, the exact amount to be determined at trial.

VI. PRAYER FOR RELIEF

167. WHEREFORE, Plaintiff prays for the following relief:

168. A judgment in favor of Plaintiff that Motorola, has infringed, directly and indirectly, including by way of inducement infringement, literally and/or under the doctrine of equivalents, the Patents-in-Suit;

169. Plaintiff's actual damages in an amount sufficient to compensate Plaintiff for Motorola's infringement of the Patents-in-Suit until such time as Motorola ceases its infringing conduct, including supplemental damages post-verdict;

170. Pre- and post-judgment interest as allowed by law on any damages awarded to Plaintiff;

171. A judgment and order requiring Motorola to pay the expenses and costs of this action (including all disbursements), as well as attorneys' fees as provided by 35 U.S.C. § 285;

172. A judgment and order requiring that Motorola pay to Plaintiff compulsory ongoing licensing fees, as determined by the Court in equity; and

173. Such other and further relief in law or in equity to which Plaintiff may be justly entitled.

VII. DEMAND FOR JURY TRIAL

Plaintiff demands a trial by jury of any and all issues triable of right before a jury, except for future patent infringement, which is an issue in equity to be determined by the Court.

Dated: May 7, 2021

MCKOOL SMITH, P.C.

/s/ *Sam Baxter*
Samuel F. Baxter
Lead Attorney
Texas State Bar No. 01938000
sbaxter@mckoolsmith.com
Jennifer Truelove
Texas State Bar No. 24012906
jtruelove@mckoolsmith.com
MCKOOL SMITH, P.C.
104 East Houston, Suite 300
Marshall, Texas 75670
Telephone: (903) 923-9000
Facsimile: (903) 923-9099

Kevin Schubert
kschubert@mckoolsmith.com
MCKOOL SMITH, P.C.
395 9th Avenue, 50th Floor
New York, NY 10001

Eric Hansen
Texas State Bar No. 19196650
ehansen@mckoolsmith.com
MCKOOL SMITH, P.C.
300 Crescent Court, Suite 1500
Dallas, Texas 75201

ATTORNEYS FOR PLAINTIFF SEVEN NETWORKS, LLC