

IN THE UNITED STATES DISTRICT COURT  
FOR THE DISTRICT OF DELAWARE

SOUND VIEW INNOVATIONS, LLC,	)	
	)	
Plaintiff,	)	
	)	
v.	)	Civil Action No. _____
	)	
FACEBOOK, INC.	)	<b>JURY TRIAL DEMANDED</b>
	)	
Defendant.	)	

**COMPLAINT FOR PATENT INFRINGEMENT**

Plaintiff Sound View Innovations, LLC (“Sound View”), for its Complaint for Patent Infringement against Facebook, Inc. (“Facebook”) alleges as follows:

**INTRODUCTION**

1. Sound View is an intellectual property licensing company. Sound View’s patent portfolio includes more than 900 active and pending patents worldwide, including approximately 500 U.S. Patents. Sound View’s patents were developed by researchers at Alcatel Lucent (“Lucent”) and its predecessors. Lucent is home to the world-renowned Bell Laboratories, which has a long and storied history of innovation. Researchers at Lucent’s Bell Laboratories have developed a wide variety of key innovations that have greatly enhanced the capabilities and utility of personal computers. Common features such as computer networking and user interfaces have benefited from Lucent’s research and development efforts.

2. Patents enjoy the same fundamental protections as real property. Sound View, like any property owner, is entitled to insist that others respect its property and to demand compensation from those who take it for their own use. Facebook has, and continues to use Sound View’s patents. Moreover, despite Sound View’s repeated attempts to negotiate,

Facebook refuses to take a license, but continues to use Sound View's property.

### **NATURE OF THE CASE**

3. This action arises under 35 U.S.C. § 271 for Facebook's infringement of Sound View's United States Patent Nos. 5,991,845, (the "845 patent"), 6,125,371 (the "371 patent"), 6,732,181 (the "181 patent"), 7,366,786 (the "786 patent"), 7,412,486 (the "486 patent"), 8,095,593 (the "593 patent"), and 8,135,860 (the "860 patent") (collectively the "Patents-In-Suit").

### **THE PARTIES**

4. Plaintiff Sound View is a Delaware limited liability company, with its principal place of business at 2001 Route 46, Waterview Plaza, Suite 310, Parsippany, New Jersey 07054.

5. Defendant Facebook is a Delaware corporation with its principal place of business at 1601 South California Avenue, Palo Alto, California 94304. Facebook may be served with process by serving its registered agent, Corporation Service Company, 2711 Centerville Road Suite 400, Wilmington, Delaware 19808.

### **JURISDICTION AND VENUE**

6. This action arises under the patent laws of the United States, including 35 U.S.C. § 271 *et seq.* The jurisdiction of this Court over the subject matter of this action is proper under 28 U.S.C. §§ 1331 and 1338(a).

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

8. This Court has personal jurisdiction over Facebook because, among other things: Facebook is incorporated under the laws of the State of Delaware; Facebook has committed, aided, abetted, contributed to and/or participated in the commission of acts giving rise to this action within the State of Delaware and this judicial district and has established minimum

contacts within the forum such that the exercise of jurisdiction over Facebook would not offend traditional notions of fair play and substantial justice; Facebook has placed products and services that practice the claims of the Patents-in-Suit into the stream of commerce with the reasonable expectation and/or knowledge that actual or potential users of such products and/or services were located within this judicial district; and Facebook has sold, advertised, solicited customers, marketed and distributed its services that practice the claims of the Patents-in-Suit in this judicial district.

### **THE PATENTS-IN-SUIT**

9. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

10. The '845 patent, titled "Recoverable Spin Lock System," was duly and properly issued by the USPTO on November 23, 1999. A copy of the '845 patent is attached hereto as Exhibit A.

11. Sound View is the assignee of the '845 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

12. The '371 patent, titled "System And Method For Aging Versions Of Data In A Main Memory Database," was duly and properly issued by the USPTO on September 26, 2000. A copy of the '371 patent is attached hereto as Exhibit B.

13. Sound View is the assignee of the '371 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

14. The '181 patent, titled "Internet-Enabled Service Management And Authorization System And Method," was duly and properly issued by the USPTO on May 4, 2004. A copy of the '181 patent is attached hereto as Exhibit C.

15. Sound View is the assignee of the '181 patent and holds the right to sue for and

recover all damages for infringement thereof, including past infringement.

16. The '786 patent, also titled "Internet-Enabled Service Management And Authorization System And Method," was duly and properly issued by the USPTO on April 29, 2008. The '786 patent stems from a continuation application on which the '181 patent is based. A copy of the '786 patent is attached hereto as Exhibit D.

17. Sound View is the assignee of the '786 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

18. The '486 patent, titled "Method And Apparatus Providing A Web Based Messaging System," was duly and properly issued by the USPTO on August 12, 2008. A copy of the '486 patent is attached hereto as Exhibit E.

19. Sound View is the assignee of the '486 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

20. The '593 patent, titled "Method for Managing Electronic Information, the Related User Terminal and the Related Information Source," was duly and properly issued by the USPTO on January 10, 2012. A copy of the '593 patent is attached hereto as Exhibit F.

21. Sound View is the assignee of the '593 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

22. The '860 patent, titled "Content Interpolating Web Proxy Server," was duly and properly issued by the USPTO on March 13, 2012. A copy of the '860 patent is attached hereto as Exhibit G.

23. Sound View is the assignee of the '860 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

#### **BACKGROUND FACTS**

24. On July 15, 2014, Sound View sent a letter to Facebook notifying Facebook of its

infringement of ten patents. Of those ten patents, three of them are Patents-in-Suit. More specifically, with respect to those three Patents-In-Suit, Sound View notified Facebook of representative Facebook features that infringe those patents, including, for example, that its Target Advertisement feature was infringing the '593 patent, and that its Account Verification was infringing the '181 and '786 patents. In its July 15, 2014 letter, Sound View explained its intention to allow Facebook to continue to use the inventions covered in those patents through a license from Sound View. Sound View further requested a meeting to discuss the matter in more detail and review claim charts with Facebook.

25. Several letters were exchanged between the parties, but Facebook did not accept Sound View's request for a meeting.

26. On August 10, 2015, Sound View sent an additional letter to Facebook notifying it of its infringement of an additional six patents. Of those six patents, three of them are Patents-in-Suit. More specifically, with respect to those three Patents-In-Suit, Sound View notified Facebook of representative Facebook features that were infringing those patents, including, for example, that its server applications that use POSIX threads were infringing the '845 patent, that its Messages platform and other systems that use HBase and/or HydraBase data structures were infringing the '371 patent, and that its Chat, Messenger, and other applications that utilize Extensible Messaging and Presence Protocol ("XMPP") were infringing the '486 patent. In the August 10, 2015 letter, Sound View again requested to meet with Facebook to discuss the matter. Sound View indicated that it hoped to be able to present Facebook with a company overview as well as some representative claim charts in an effort to reach a licensing agreement.

27. Several additional letters and emails were exchanged, but Facebook has not accepted Sound View's request to meet. For example, on February 4, 2016, Sound View

informed Facebook via email that it infringes two additional Sound View patents. One of those patents, the '860 patent, is at issue in this lawsuit. Sound View informed Facebook that all Facebook systems that utilize a Wireless Universal Resource File infringe the '860 patent.

28. As of the date of the filing of this Complaint, Facebook has not explained why it continues to use Sound View's patents without permission. As of the date of the filing of this Complaint, Facebook has not claimed that it does not infringe any of the Patents-in-Suit, and has not challenged the validity of any of the Patents-in-Suit.

29. Facebook has refused to engage in any meaningful discussions about reaching a license agreement to end its infringement of Sound View's patents. Instead, Facebook continues to willfully infringe Sound View's patents so as to obtain their significant benefits without paying any compensation to Sound View.

30. After about eighteen months of Facebook's failure to provide any substantive response and unwillingness to meet in an attempt to negotiate a license agreement that remedies Facebook's unlawful conduct, Sound View is now forced to seek relief through litigation.

### **COUNT ONE**

#### **INFRINGEMENT OF THE '845 PATENT**

31. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

32. The '845 patent generally relates to improved, recoverable computer spin locks that prevent simultaneous access of shared memory structures in multi-processing environments. In the context of the '845 patent, a "lock" is a computer mechanism that ensures mutually exclusive access to a shared computer resource. In the context of the '845 patent, if a lock is busy, a process attempting to acquire the lock can wait or "spin" until the lock is released. An implementation in which a process repeatedly tries to acquire the lock in a tight loop is called a

“spin lock.”

33. At the time of invention of the '845 patent, both software-based and hardware-based spin locks were known as mechanisms of providing exclusive access to shared resources. In one such spin lock algorithm, all processes operating in a multi-processor system frequently access, attempt to write to, and cache a spin lock control variable. Because each update will lead to cache invalidation messages being sent to all other processors, such algorithms can overburden the caching system and hence, are not viable for processor-scalable architectures, i.e., shared memory multiprocessors.

34. A scalable spin lock protocol known before the '845 patent was called the MCS-lock protocol. However, the MCS-lock protocol was vulnerable to process failure in that if a process terminated while waiting for the lock, once it received the lock it would never release it. Similarly, while owning or releasing the lock, the death of the process would prevent ownership from being passed on to a subsequent process. In other words, the MCS-lock protocol was not recoverable.

35. Another vulnerability of the MSC-lock protocol was the window of time between the initial swap in the lock acquisition code and a subsequent assignment into the queue. If a process died or terminated after executing the swap, but before setting the next field, then the queue would become “broken” or fragmented at that point. Attempting to recover from such a fragmented list would have been difficult.

36. The '845 patent solved that discrete computer-based problem and improved upon the MCS-lock protocol by providing a recoverable spin lock, i.e., one that does not become permanently unavailable even if one or more of the processes in the queue structure (i.e. spinning on the lock) terminates. To ensure the integrity of the queue structure, the recovering spin lock

system disclosed in the '845 patent reassembles the linked list queue structure after processes that have failed or terminated are removed, and assigns exclusive access of the lock to a new process if the process that had previously owned the lock had terminated.

37. The capability of recovering a spinning lock as described in the '845 patent is particularly useful for servers such as transaction processors which consist of several processes that are often in continuous operation. The ability to determine the process having exclusive access of a standard spinning lock in spite of any sequence of process failures is a key requirement for recoverability. If the process having exclusive access can be reliably determined, then the shared data guarded by the lock can be returned to use if that process has terminated.

38. In violation of 35 U.S.C. § 271, Facebook has infringed at least claim 13 of the '845 patent by having made, designed, offered for sale, sold, provided, used, maintained, and/or supported a Portable Operating System Interface (POSIX) and robust mutual extension locks (Robust mutexes). Facebook's infringement is continuing.

39. On August 10, 2015, Sound View informed Facebook that its server applications that use POSIX threads infringe the '845 patent. However, Facebook has not stopped infringing.

40. Facebook uses POSIX threads for concurrent computations, with the goal of maximizing throughput by fully utilizing its machines' CPU and RAM. POSIX threads allow Facebook to run multiple processes using the same shared resources.

41. Facebook utilizes Robust mutexes to protect shared resources. The Robust mutex acts like a key to control access to a resource. Only the thread that has the key (or mutex) can use the resource. In order for a thread to use a resource, the thread must first acquire the key (or mutex). If the mutex is already locked by a thread, the next thread waits for the mutex to become available. Nodes representing the threads are queued in a linked list, which is a common



data structure. More generally, a linked list is an ordered collection of finite homogeneous data elements called nodes, where the linear order is maintained using links or pointers. Facebook uses a singly linked list data structure for its POSIX threads. When a thread waits for a mutex, that thread is in a linked list associated with the mutex being waited upon. Once the mutex is released, the first thread in the mutex queue will be executed. If a waiting thread dies, it is removed from the linked list and the linked list is reassembled. If a thread that owns the Robust mutex terminates, another thread will have the opportunity to take over and clean up the state that was protected by the mutex.

42. For example, Facebook infringes claim 13 by:
  - a. performing a method for providing multiple processes (such as threads) with mutually exclusive access to a shared resource in a system having a lock (such as a Robust mutex) associated with the shared resource, possession of the lock signifying exclusive access to the shared resource, wherein processes desiring access to the shared resource spin on the lock (e.g., wait for ownership of the mutex) until the lock is acquired,
  - b. maintaining a linked queue structure of data records (such as the nodes in the linked list) corresponding to a queue of processes including processes spinning on the lock (such as the threads waiting for the mutex) and a process possessing the lock (such as the thread possessing the mutex), one data record per process (such as a thread);
  - c. transferring the lock (the mutex) from the process possessing the lock (such as the thread with the mutex) to a process next in the queue (such as the thread waiting for the mutex in queue);
  - d. conducting a cleanup process (such as in a pthread cleanup) if one or more processes in the queue have terminated (such as when they are dead threads), said cleanup

process removing said one or more terminated processes from the queue (such as releasing the mutex and removing a dead waiter from the chain) and reassembling the linked queue structure.

43. Facebook has infringed and continues to infringe the '845 patent, which has damaged Sound View. Sound View is entitled to recover from Facebook the damages sustained by Sound View as a result of Facebook's wrongful acts in an amount adequate to compensate Sound View for Facebook's infringement pursuant to 35 U.S.C. § 284. In committing these acts of infringement, Facebook acted despite an objectively high likelihood that its actions constituted infringement of a valid patent, and Facebook actually knew or should have known that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

44. The infringement by Facebook of the '845 patent was, and continues to be, deliberate and willful, entitling Sound View to increased damages under 35 U.S.C. § 284 and to attorney fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

## **COUNT TWO**

### **INFRINGEMENT OF THE '371 PATENT**

45. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

46. The '371 patent generally relates to an improved multi-versioned database management system and method that creates multiple versions of data records affected by update transactions and increases capacity of memory by deleting versions of data records in response to associated time stamps and a measurable characteristic of the memory. In the context of the '371 patent, "measurable characteristics of the memory" are measurable characteristics describing, relating to, or otherwise associated with a utilization or capacity of the memory. In the context of the '371 patent, "update transactions" refers to transactions that modify portions of the database such as "write" or "put" transactions.

47. Database managers (DBMs) have long been used in computer systems to manage large amounts of data. A DBM is a control application that supervises or manages interactions between application tasks and a database. The '371 patent inventors recognized that two important DBM functions are: to ensure: (i) data recovery (in response to a database crash caused by, for example, a power outage, or a program crash), and (ii) data integrity. Data recovery involves rebuilding at least part of a database after all or part of its data is corrupted or lost, based on the last known valid or uncorrupted state. With respect to data integrity, latency in DBMs was largely intolerable. Latency refers to the time differential between a request for data and subsequent receipt of data. Latency is largely impacted by the type of computer memory on which the database is stored. There are two classifications of computer memory, volatile memory and non-volatile memory. Volatile memory is memory which does not retain data after power is lost, and is typically characterized by fast access to data. Non-volatile memory is memory that retains data after power is lost and is typically characterized by slower access to data. As a general matter, volatile memory is more expensive than non-volatile memory. Early computer database systems were divided among main (volatile) memory and disk (non-volatile memory). Those disk-based DBMs frequently failed to meet the performance requirements of contemporary information management systems (IMSS) because of the latencies inherent with non-volatile memory transactions.

48. One popular method to solve that latency problem was to map the entire database into the main memory. For data integrity purposes, however, those conventional main memory DBMs had to delay the processing of update transactions. For example, the conventional main memory DBMs had to prevent an update transaction from modifying a data record while another process was simultaneously relying on that data record. In order to reduce conflicts between

update transactions and read-only transactions, contemporary databases created multiple versions of data records, known as multi-versioning. In those multi-version DBMs, read-only transactions were given consistent, but out-of-date views of certain data records or data record types.

49. Although those multi-versioning techniques reduced “waits” and conflicts among transactions, they conflicted with DBM efforts to utilize main memory capacity efficiently because main memory continuously expended processing resources collecting data record versions that were longer needed. The ’371 patent solved this computer-based problem—that of lacking an efficient means to reclaim main memory space no longer used by multi-version techniques—by logically and economically aging data record versions in the database. The ’371 patent inventions extend to, and provide benefits to, DBMs that utilize secondary or mass storage as opposed to main memory.

50. To solve this discrete computer-centric problem, the ’371 patent teaches a system that includes each of a time stamping controller, a versioning controller and an aging controller. The time stamping controller assigns a time stamp to transactions to be performed on the database, and may be assigned as a function of a time stamp counter. The time stamp operates to preserve an order of the transactions. The versioning controller creates multiple versions of data records of the database that are affected by update transactions. The aging controller, which may be associated, directly or indirectly, with each of the time stamping and versioning controllers, monitors at least one measurable characteristic and deletes prior ones of the multiple data record versions in response to the time stamp and the at least one measured characteristic to thereby increase the data capacity of the database, thus increasing memory capacity.

51. In violation of 35 U.S.C. § 271, Facebook has infringed at least claims 1, 2, 3, 8, 9

and 10 of the '371 patent by having made, designed, offered for sale, sold, provided, used, maintained, and/or supported its Facebook platforms, including for example its messaging platform, that use HBase and/or HydraBase. Facebook's infringement is continuing.

52. On August 10, 2015, Sound View informed Facebook that its systems that utilize the HBase and/or HydraBase infringe the '371 patent. However, Facebook has not stopped infringing.

53. HBase and/or HydraBase is used by Facebook in its Messages platform as well as its internal monitoring system, its Nearby Friends feature, search indexing, streaming data analysis, and data scraping for Facebook's internal data warehouses. HBase was chosen as the underlying durable data store for Facebook's Messages platform because it provided high write throughput and low latency random read performance. It also provided other features including horizontal scalability, strong consistency, and high availability via automatic failover. Beginning around 2014, Facebook upgraded the HBase database system with a new open source system called HydraBase. HydraBase utilizes the HBase framework and minimizes downtime when a server fails by having each region hosted on multiple region servers ("RegionServers"). HydraBase did not change the concept of Tables in HBase, and as in HBase, each table in HydraBase is divided into regions hosted by the RegionServers.

54. HBase and HydraBase are column-oriented database management systems that run on top of a Hadoop Distributed File System. Applications store data into HBase and/or HydraBase tables that are made up of rows and columns. Table cells—the intersection of row and column coordinates—are versioned. When something is written into one of Facebook's HBase and/or HydraBase databases, it is first written to an in-memory store (memstore), and then is flushed into a store file (StoreFile). When Facebook puts data into HBase and/or

HydraBase, a timestamp is required and is generated by the RegionServer. Performing a “put” operation to HBase and/or HydraBase creates a new version of a cell at a certain timestamp. The RegionServer generates globally unique, time-based transaction IDs for each transaction that is started and manages the versions.

55. Facebook controls the number of versions stored in HBase and/or HydraBase. For example, Facebook sets the maximum number of versions to store at a level that will ensure that the StoreFile size does not become too large.

56. During major compaction, excess versions are deleted from the StoreFile. The number of versions to be deleted is determined by comparing the number of versions stored to the MaxVersions and/or the Compaction Threshold. If the number of stored versions in the StoreFile is greater than the Compaction Threshold and/or Max Versions, then the excess versions are deleted. The versions that are deleted are generally the versions with the oldest timestamp.

57. For example, Facebook infringes claim 1 through its:

- a. processing system (such as Facebook’s servers) for use with a database of data records (such as those within the database managed by HBase / HydraBase), said database stored in a memory (such as the memstore and StoreFile), comprising:
  - b. a time stamping controller (such as the controller that uses the `TIMESTAMP_NOW` variable) that assigns a time stamp to transactions (such as put transactions) to be performed on said database;
  - c. a versioning controller (such as that within the RegionServer) that creates multiple versions of ones of said data records affected by said transactions that are update transactions (such as put transactions that modify the database); and

d. an aging controller (such as the major compaction function) that monitors a measurable characteristic of said memory (such as the number of versions being stored in the columns in the StoreFile) and deletes ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory (such as the StoreFile).

58. Facebook has infringed and continues to infringe the '371 patent, which has damaged Sound View. Sound View is entitled to recover from Facebook the damages sustained by Sound View as a result of Facebook's wrongful acts in an amount adequate to compensate Sound View for Facebook's infringement pursuant to 35 U.S.C. § 284. In committing these acts of infringement, Facebook acted despite an objectively high likelihood that its actions constituted infringement of a valid patent, and Facebook actually knew or should have known that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

59. The infringement by Facebook of the '371 patent was, and continues to be, deliberate and willful, entitling Sound View to increased damages under 35 U.S.C. § 284 and to attorney fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

### **COUNT THREE**

#### **INFRINGEMENT OF THE '181 PATENT**

60. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

61. The '181 patent generally relates to a system that uses a web browser, a login Common Gateway Interface ("CGI"), and a server's shared memory's simple user database to allow a user to access a list of other users for which the user has access based on validation of at least the user's identification and password information, and the user access type. In the context of the '181 patent, a "CGI" is a standard for interfacing external applications with information

servers, such as HTTP or Web servers. A CGI program is executed in real-time, so that it can output dynamic information. The CGI script is sent to a client's web browser, which allows the client to enter a user ID and password.

62. At the time of the '181 patent, service provider subscribers were demanding increased visibility and control of their subscribed communications services in order to ensure that their networks remained cost-effective and responsive to rapidly changing needs. The '181 patent extended the customer service management market beyond large corporate customers by developing a system that provided Internet-based access to an account based on user authentication that was also able to accommodate additional access to other users from a single login event.

63. The '181 patent solved this discrete computer-centric problem of accessing information relating to other users without requiring an additional login event by using access types, CGI scripts and user databases. The '181 patent provides a web-based presentation environment and a number of valuable customer service management capabilities that could be accessed via the web medium. The '181 patent also provides the flexibility and manageability of web browser and Internet/Intranet technologies. The customer service management features were designed to work with the industry standard web browsers of the time, including Netscape Navigator and Microsoft Internet Explorer.

64. In violation of 35 U.S.C. § 271, Facebook has infringed and/or induced others to infringe at least claims 5, 6, 7, and 8 of the '181 patent by having made, designed, offered for sale, sold, provided, used, maintained, and/or supported its Facebook platform that includes an authorization system with different levels of access, such as the "Page roles" feature with Facebook Pages. Facebook's infringement is continuing.



65. On July 15, 2014, Sound View informed Facebook that its Account Verification systems infringe the '181 patent. However, Facebook has not stopped infringing.

66. Facebook Pages help businesses, organizations, and brands share their stories and connect with people. Like profiles, Pages can be customized by publishing stories, hosting events, adding apps, and more. Facebook users can create and manage a Page. Users with authorization can publish posts as the Page. Page posts can appear in the News Feeds of people who "like" the Page. Users with authorization can also create customized apps for the Page and check Page Insights to track the Page's growth and activity. There are five different types of roles for people who manage Pages. Only an "Admin" can change someone's role. The other Page roles include "Editor", "Moderator", "Advertiser" and "Analyst".

67. An authorized user of a Page (e.g., an Admin) can publish, post, "like" or comment as that Page, as opposed to being limited to doing so using his or her own account profile. For example, from the Page a user wants to post or publish on, a user can also click the flag icon or the profile picture in the top right, and select whether to post as the user or as one of the Pages. Facebook stores its users' email addresses, passwords, and names. Facebook also stores the Page roles its users have. Facebook Graph API is composed of nodes, edges, and fields. The edges are connections between the nodes, and the fields contain information about the nodes. For each user, Facebook uses the "accounts" edge to store the Facebook Pages for which that user is assigned one of the five roles (e.g., an Admin).

68. To access Facebook, a user may type his or her email address and password into a Facebook provided CGI and click "Log In." Facebook then validates that user's information. In a drop-down menu on the top right-hand side of the Facebook page, a user will see a list of Pages the user is permitted to access. An Admin, an Editor, or a Moderator can send messages as a

Page, and respond to and delete comments and posts on the Page. A Page Admin can more broadly configure the Page for all users.

69. For example, Facebook's Account Verification infringes claim 5 because it is a system for providing a user of an Internet-based communication system selective access to information relating to other users comprising:

a. a server having means to store a list of users including user access type, identification, password and name (such as the fields within Facebook's server database that include, for example, "id", and "last\_name");

b. a user client having means for a user to input identification and password information (such as the client browser or mobile application that runs the login CGI program that can be seen on the login page beginning at <form id="login\_form" and ending at </form>);

c. and means at said server to compare said user input information with stored information (such as Facebook's login CGI and Facebook's user database) and based on user verification and user access type (such as one of the five page roles) provide said user with a list of other users for which said user has access (such as a user's profile and/or any Pages for which the user is assigned a page role).

70. Facebook indirectly infringes the '181 patent by inducing infringement by others, such as its Facebook customers, by, for example, encouraging and instructing end-user customers to use the claimed system and functionality identified, such as Page roles. Facebook distributes instructions through its website on how to use the infringing functionality, including through its help center webpages located at <https://www.facebook.com/help>, and through its Facebook for business webpages located at <https://www.facebook.com/business>. Facebook's customers put the claimed system into service when they log in to Facebook. When they do so,

they cause Facebook's servers to authenticate them and give them the appropriate access as described above. Facebook takes these actions intending to cause infringing acts by its customers.

71. Facebook was aware of the '181 patent and knew that its customers' actions, if taken, would constitute infringement of that patent. Alternatively, Facebook subjectively believed there was a high probability that its customers would infringe the '181 patent but took deliberate steps to avoid confirming that it was actively inducing infringement by customers. Facebook therefore infringes the Patents-In-Suit under 35 U.S.C. § 271(b).

72. Facebook has infringed and continues to infringe the '181 patent, which has damaged Sound View. Sound View is entitled to recover from Facebook the damages sustained by Sound View as a result of Facebook's wrongful acts in an amount adequate to compensate Sound View for Facebook's infringement pursuant to 35 U.S.C. § 284. In committing these acts of infringement, Facebook acted despite an objectively high likelihood that its actions constituted infringement of a valid patent, and Facebook actually knew or should have known that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

73. The infringement by Facebook of the '181 patent was, and continues to be, deliberate and willful, entitling Sound View to increased damages under 35 U.S.C. § 284 and to attorney fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

#### **COUNT FOUR**

#### **INFRINGEMENT OF THE '786 PATENT**

74. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

75. The '786 patent generally relates to a system that uses a web browser, a login Common Gateway Interface ("CGI"), TCP/IP messages and a server's shared memory's simple

user database to allow a user to log in to a system after the user's identification (ID), password, and IP address are sent to the server via TCP/IP messages, and providing the user with a cookie that allows access based on validation of the user ID and password and the user's dynamic status information. In the context of the '786 patent, a "CGI" is a standard for interfacing external applications with information servers, such as HTTP or Web servers. A CGI program is executed in real-time, so that it can output dynamic information. The CGI is sent to a client's web browser, which allows the client to enter a user ID and password. In the context of the '786 patent, "dynamic status information", includes at least an enabled, disabled, or active dynamic status information.

76. The '786 patent provides a system that was designed to work with the industry standard web browsers of the time, including Netscape Navigator and Microsoft Internet Explorer, and allows for additional logons of identical user IDs. Due to the statelessness of the HTTP protocol, prior mechanisms did not allow multiple logons of identical user IDs because a user would remain logged on to a system until either the user logged out of the application or the user's session was timed out by the server application. That system presented a problem if the user's Internet browser crashed and the user wished to re-logon to the system because the user would have had to wait until the previous session was timed out by the server application before the user could re-logon to the application. Additionally, that logon mechanism also did not allow a user to switch to another workstation to logon to the application whilst having a current active session on another workstation. While having a current active session on another workstation, the user either needed to log out from the application from his original workstation—which was not possible if his original workstation had crashed or had become temporarily inoperable—or wait until his current session was timed out by the server application before he could logon from

another workstation.

77. To solve this computer-based problem, the '786 patent system described above includes dynamic status information that creates the ability to accommodate a subsequent logon when a user's web browser has crashed or the user is operating from another workstation and provides access authorization using a cookie that is sent to the user. For each user, the application stores at least a user ID, a user password, dynamic status, and an IP address. When a user enters a user ID and password and requests access, the application validates the information provided against the information stored for that user. If the user ID and password matches, the application checks the user's status in the application. The user's IP address also may be compared against the IP address that has previously been stored for that user in order to determine whether the user is attempting to log in from a different device.

78. In violation of 35 U.S.C. § 271, Facebook has infringed and/or induced others to infringe at least claims 1, 2, 3, 4, and 7 of the '786 patent by having made, designed, offered for sale, sold, provided, used, maintained, and/or supported its Facebook platforms that include an authorization system that tracks dynamic status information. Facebook's infringement is continuing.

79. On July 15, 2014, Sound View informed Facebook that its Facebook Account Verification system infringes the '786 patent. However, Facebook has not stopped infringing.

80. To access Facebook, a user may type his or her email address and password into a Facebook provided CGI and click "Log In." The user's email address and password, as well as the user's IP address are then sent to Facebook's servers over a TCP/IP connection. Before the user's information can be sent to the Facebook server, a handshake using the Transport Layer Security (TLS) protocol occurs between the client and the server. During that TLS handshake

the server sends a certificate to the client for authentication. After that certificate is received by the client, the client and server can exchange application data relating to authentication credentials. Facebook validates that user's credentials before allowing a user to log in.

81. Facebook user accounts may be disabled for various reasons. If a Facebook account has been disabled, the user will see a disabled message when the user tries to log in. In other words, in order to be allowed to log in, a Facebook account must be enabled. Facebook also stores information regarding its users' active sessions, including the IP address, machine cookie and browser information for the active sessions.

82. Once a user is allowed to log in based on a valid email address, password, and non-disabled account, the Facebook server provides the user with a cookie that contains an access token.

83. For example, Facebook's Account Verification infringes claim 1 because it is a system for authorizing a user of a client to have access to a server via the Internet comprising:

- a. means in said client for inputting a user identification (ID) and user password (such as the client browser or mobile application that runs the login CGI program that can be seen on the login page beginning at `<form id="login_form">` and ending at `</form>`);
- b. means in said client for storing a unique client address (such as the IP address field within the web browser corresponding to the REMOTE\_ADDR variable, e.g., X-CLIENTIP);
- c. communication means at said client for passing said ID, password and address to said server via said Internet (such as the TCP/IP message sent during Facebook login) in response to a request therefrom (such as the server message in the TLS handshake);
- d. means at said server to store information respecting said client and to

compare said stored information with said user ID and user password (such as Facebook's login CGI and Facebook's user database);

e. means at said server to store dynamic status information respecting said user, said dynamic status information being one of enabled, disabled or active (such as the Facebook databases that store information on Current Session and Disabled Accounts); and

f. means to authorize log in of said user if said ID and password agree with said stored information and if said user status is enabled (such as a cookie with an access token).

84. Facebook indirectly infringes the '786 patent by inducing infringement by others, such as its Facebook customers, by, for example, encouraging and instructing customers to use the claimed system and functionality identified. Facebook distributes instructions through its website on how to use the infringing functionality. Facebook seeks continually to increase its number of customers and also to increase the amount of time those customers spend on Facebook's website. Facebook's customers put the claimed system into service when they log in to Facebook. When they do so, they cause Facebook's servers to authenticate them and give them access as described above. Facebook takes these actions intending to cause infringing acts by its customers.

85. Facebook was aware of the '786 patent and knew that its customers' actions, if taken, would constitute infringement of that patent. Alternatively, Facebook subjectively believed there was a high probability that its customers would infringe the '786 patent but took deliberate steps to avoid confirming that it was actively inducing infringement by customers. Facebook therefore infringes the Patents-In-Suit under 35 U.S.C. § 271(b).

86. Facebook has infringed and continues to infringe the '786 patent, which has damaged Sound View. Sound View is entitled to recover from Facebook the damages sustained

by Sound View as a result of Facebook's wrongful acts in an amount adequate to compensate Sound View for Facebook's infringement pursuant to 35 U.S.C. § 284. In committing these acts of infringement, Facebook acted despite an objectively high likelihood that its actions constituted infringement of a valid patent, and Facebook actually knew or should have known that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

87. The infringement by Facebook of the '786 patent was, and continues to be, deliberate and willful, entitling Sound View to increased damages under 35 U.S.C. § 284 and to attorney fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

#### **COUNT FIVE**

#### **INFRINGEMENT OF THE '486 PATENT**

88. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

89. The '486 patent generally relates to a system and method for sending and receiving messages between computers that is independent of the operating system of the messaging client and exclusive of proprietary messaging software residing and previously stored on the messaging client, by sending a message processing script and message over a first connection and maintaining that connection in an open state after transmitting the first message so that additional messages can be transmitted over the same connection.

90. At the time of the invention of the '486 patent there existed conventional instant messaging (IM) systems, which allowed two or more computers on a network to quickly exchange messages. An example of such an IM system was AOL Instant Messenger. The problem with those conventional IM systems is that they required a user to download a proprietary messaging client software application in order to be able to send or receive messages. The messaging client software applications typically utilized a specialized messaging protocol



for exchange of instant messages between one client and a messaging server operating at the instant messaging service provider facility. That server would then use a similarly proprietary protocol to forward the instant message to the recipient's client messaging software in real-time or near-real-time.

91. One of the problems associated with the conventional IM systems was that they required that developers of the messaging client software applications write different versions of such client applications specifically for each separate computerized device platform upon which the IM system was to operate. An additional problem associated with the conventional IM systems was that the user had to configure his or her client computer system to support the instant messaging by downloading and installing the proper messaging client software application. For that reason, if new features were added to the messaging system, the user had to reinstall or upgrade the messaging client application to enable the use of such new features. Also, since such applications were executables, they were susceptible to attack by computer viruses that might maliciously have been incorporated into new versions of the messaging client software. Additionally, such messaging client software occupies space within users' computer systems and installation of such software was prone to errors.

92. Although some conventional IM systems were Java-based, and did not require a user to manually download and install an executable program, Java-based messaging systems suffered from other deficiencies. For example, Java (which is a full-blown programming language, not to be confused with JavaScript) required installation and operation of a Java virtual machine in order to operate. For that reason, although Java was widely used at the time of the '486 patent, it was not the case that every data-connected device could support an application (i.e., an applet) written in Java since such applets required extensive Java virtual machine

resources to be installed and enabled on client platforms in order for downloaded Java applets to properly operate.

93. The '486 patent solved these computer-specific communication problems and provided an improved way for data-connected devices to communicate with messages such as with instant messages. In order to process message data, the '486 patent teaches the use of baseline technologies such as standard browser functionality, HTML, HTTP and JavaScript—technologies that were provided within almost all data-connected devices ranging from cell phones to personal computers.

94. One embodiment of the '486 patent operates in a messaging server and provides a method for processing message data that comprises the steps of receiving a first message to be sent to a messaging client and establishing a first connection to the messaging client. The first connection may be, for example, a long-lived HTTP connection. The method transmits a message processing script and the first message over the first connection to the messaging client. The message processing script enables the messaging client to display the first message and to receive and display at least one second message over the first connection (i.e., the same connection) to the messaging client. The method maintains the first connection in an open state after transmitting the first message to the messaging client. Thus, the message page and the first message both appear, from the browser's perspective, to be part of a web page that the browser is receiving over a single HTTP connection or session. The method then receives a second message to be sent to the messaging client and transmits the second message over the first connection to the messaging client for receipt by the message processing script, while continuing to maintain the first connection in an open state.

95. In violation of 35 U.S.C. § 271, Facebook has infringed at least claims 1-8 and

19-27 of the '486 patent by having made, designed, offered for sale, sold, provided, used, maintained, and/or supported features based on XMPP (Extensible Messaging and Presence Protocol) over BOSH (Bidirectional streams Over Synchronous HTTP) such as Facebook Chat, and Facebook Messenger. Facebook's infringement is continuing.

96. On August 10, 2015, Sound View informed Facebook that its applications that use XMPP, specifically including Facebook Chat and Facebook Messenger, infringe the '486 patent. However, Facebook has not stopped infringing.

97. To make Facebook Chat available everywhere, Facebook uses XMPP, an open messaging protocol. The Facebook Chat client is written in JavaScript and has no plugins. Facebook Chat allows Facebook users near-real-time exchange of messages. At Facebook, the Channel Servers are the backend portion that is responsible for queuing a given user's messages and pushing the messages to the user's web browser via HTTP. Facebook's Channel Servers are XMPP servers. The method Facebook uses to get text from one user to another involves loading an iframe on each Facebook page, and having that iframe's Javascript make an HTTP GET request over a persistent connection that does not return until the server has data for the client. Facebook uses BOSH for its persistent XMPP connections.

98. For example, Facebook Chat infringes claim 20 because it utilizes a messaging server computer system comprising:

- a. at least one communications interface (such as the TCP/IP interface of Facebook's Channel / XMPP servers);
  - b. a memory (such as the memory in Facebook's Channel / XMPP servers);
  - c. a processor (such as the CPU in Facebook's Channel / XMPP servers);
- and

- d. an interconnection mechanism coupling the at least one communications interface, the memory and the processor (such as the bus in Facebook's Channel / XMPP servers);
- e. wherein the memory is encoded with a server message handler application (such as Facebook Chat and Facebook Messenger) that when performed on the processor, produces a server message handler process that causes the messaging server computer system (such as the Channel / XMPP server) to processing message data by performing the operations of:
  - f. receiving, over the at least one communications interface, a first message (such as a Chat message sent by a first Facebook user) to be sent to a messaging client (such as that of a second Facebook user);
  - g. establishing a first connection (such as a TCP/IP connection) to the messaging client over the at least one communications interface;
  - h. transmitting a message processing script (such as the iframe JavaScript functions) and the first message over the first connection (such as the TCP/IP connection) to the messaging client (such as that of a second Facebook user), the message processing script enabling the messaging client to display the first message (such as in the chat window) and to receive and display at least one second message (such as a second chat message sent by the first Facebook user) over the first connection (such as the TCP/IP connection) to the messaging client (such as that of the second Facebook user) independent of the operating system (such as Microsoft Windows or Apple OS X) thereof and exclusive of proprietary messaging software (such as browser plugins) residing and previously stored on the messaging client;
  - i. maintaining the first connection in an open state after transmitting the first

message to the messaging client (such as a persistent connection);

j. receiving, over the at least one communications interface, a second message to be sent to the messaging client (such as a second chat message sent by the first Facebook user);

k. transmitting the second message over the first connection (such as the TCP/IP connection) to the messaging client (such as that of the second Facebook user) for receipt by the message processing script (such as the JavaScript), while continuing to maintain the first connection in an open state; and

l. repeating, for subsequent second messages, the steps of maintaining the first connection in an open state, receiving a second message, and transmitting the second message such that separate second messages are transmitted from the messaging server to the messaging client over the first connection (such as occurs when multiple messages are sent in a Facebook Chat).

99. Facebook has infringed and continues to infringe the '486 patent, which has damaged Sound View. Sound View is entitled to recover from Facebook the damages sustained by Sound View as a result of Facebook's wrongful acts in an amount adequate to compensate Sound View for Facebook's infringement pursuant to 35 U.S.C. § 284. In committing these acts of infringement, Facebook acted despite an objectively high likelihood that its actions constituted infringement of a valid patent, and Facebook actually knew or should have known that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

100. The infringement by Facebook of the '486 patent was, and continues to be, deliberate and willful, entitling Sound View to increased damages under 35 U.S.C. § 284 and to attorney fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

**COUNT SIX**

**INFRINGEMENT OF THE '593 PATENT**

101. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

102. The '593 patent generally relates to a community interest management module for use in a communications network with a plurality of electronic information sources and a plurality of terminals of users, that uses information preferences of an individual user as well as information preferences of users related to that user based on pre-defined relationships and also at least one decision rule for each user to determine community information preferences.

103. Prior to the '593 patent, profiling a user based on his or her behavior provided only a narrow and incomplete picture of that user. The querying of end users for their personal interests did not provide a broad picture of those users' interests.

104. One of the goals of the '593 patent inventors was to provide an electronic information management system that is better suited to the information needs of a user than previous systems where information was based on a user's interests, such as Rich Site Summary (or "RSS") feeds, and was not otherwise personalized for that user. The '593 patent inventors solved that computer-specific problem by utilizing information of other users having pre-defined relationships with the first user, such as buddies, and using a community interest management module that determines community information preferences for each user based on information preferences of a user, information preferences gathered from that user's buddies, as well as decision rules associated with each user.

105. As described in the '593 patent, the community information preferences enable electronic information to be sent towards the user that is intended to be of heightened interest to that user, because it matches not only the preferences of the user but also the preferences of the

buddies of the user. Electronic information may be any kind of electronic message such as an advertisement, an e-mail, a multimedia message, etc.

106. Additionally, the '593 patent describes a rules managing part that derives and uses decision rules for each user in order to determine community information preferences for that user. For example, these decision rules are used to determine how to apply the user's information preferences and also how to apply the information preferences of that user's buddies in determining community information preferences for that user.

107. In violation of 35 U.S.C. § 271, Facebook has infringed at least claims 1, 5, 9, 13, 14, and 15 of the '593 patent by having made, designed, offered for sale, sold, provided, used, maintained, and/or supported Facebook Advertisements. Facebook's infringement is continuing.

108. On July 15, 2014, Sound View informed Facebook that its Facebook Target Advertisement feature infringes the '593 patent. However, Facebook has not stopped infringing.

109. Facebook Ads show up in several places on Facebook, including, for example users' News Feeds. The News Feed of a user is the constantly updating list of stories on Facebook. Facebook seeks to show users ads that will be as interesting and useful to those users as possible. Until recently, Facebook's choice of which ads to display on a user's News Feed had been mostly based on things the user did on Facebook and the things the user and the user's friends "liked" on Facebook. Facebook now includes information from elsewhere through its Facebook Exchange platform. Using Facebook Exchange, advertisers can target groups of users to see an ad based on other things they have done on the Internet. Facebook does not share information with advertisers. Ad targeting at Facebook is performed by an automated system. Sometimes ad content on Facebook is paired with news about social actions taken by friends, such as how many of a user's Facebook friends "liked" a particular ad.

110. Facebook leverages the disaggregation concept in Multifeed, a distributed backend system that is involved in News Feed. When a user goes to his or her Facebook News Feed, Multifeed looks up the user's friends, finds all their recent actions, and decides what should be rendered based on a certain relevance and ranking algorithm. The major high-level components of Multifeed include:

- an Aggregator: the query engine that accepts user queries and retrieves News Feed information from backend storage. It also does News Feed aggregation, ranking and filtering, and returns results to clients. The aggregator is CPU intensive but not memory intensive;
- a Leaf: the distributed storage layer that indexes most recent News Feed actions and stores them in memory. Usually twenty leaf servers work as a group and make up one full replica containing the index data for all the users. Each leaf serves data retrieval requests coming from aggregators. Each leaf is memory intensive but not CPU intensive;
- a Tailer: the input data pipelines direct user actions and feedback into a leaf storage layer in real time; and
- Persistent storage: the raw logs and snapshots for reloading a leaf from scratch.

111. For example, Facebook infringes claim 9 because it provides:

a. a Community interest management module (CIMM) (such as the module in the Facebook servers that service Facebook Ads), for use in a communications network (CN) said communications network (CN) comprising a plurality of electronic information sources (IS . . . ISx) (such as Facebook Ad managers) and a plurality of terminals of users (UI . . . Ux) (such as Facebook users), at least one electronic information source (such as Facebook Ad managers)



of said plurality of electronic information sources (IS . . . ISx) being adapted to forward electronic information (such as an advertisement) towards a terminal of a first user (UI) (such as a Facebook user), said Community interest management module (CIMM) comprising:

b. an information preferences reception part (IPRP) (such as the Facebook databases that store Page likes or other information about users, or the Tailer and Leaf storage components of the News Feed), adapted to receive information preferences from said terminal of said first user (such as Pages the first user “likes”, other information the first user shares on Facebook, and/or actions taken on Facebook by the first user such as Pages and/or Groups visited) and information preferences of users related to said first user (U2, UX) (such as Pages the user’s friends “like”, other information the user’s friends share on Facebook, and/or actions taken on Facebook by the user’s friends such Pages and/or Groups visited), said users being related to said first user based on pre-defined relationships (such as being friends on Facebook);

c. community information preferences determination part (CIPDP) (such as Facebook’s automated Ads system, or the Aggregator component of the News Feed), adapted to determine community information preferences (such as the preferences used to display Ads in a user’s News Feed) based on information preferences (such as those identified above) of said first user and information preferences (such as those identified above) of said users related to said first user (UI) (such as the user’s friends on Facebook); and

d. a rules managing part (RMP) adapted to keep at least one decision rule for each user of said plurality of users (such as not displaying an ad when the user clicks the “x” on the top right corner of the ad and chooses “I don’t want to see this”, or the other decision rules used by the Aggregator of the News Feed), wherein that said community information determination part (CIPDP) (such as Facebook’s automated Ads system) is further adapted to

determine said community information preferences additionally based on said at least one decision rule for each user of said plurality of users (such as not displaying Ads in a user's News Feed that the user chose not to see).

112. Facebook has infringed and continues to infringe the '593 patent, which has damaged Sound View. Sound View is entitled to recover from Facebook the damages sustained by Sound View as a result of Facebook's wrongful acts in an amount adequate to compensate Sound View for Facebook's infringement pursuant to 35 U.S.C. § 284. In committing these acts of infringement, Facebook acted despite an objectively high likelihood that its actions constituted infringement of a valid patent, and Facebook actually knew or should have known that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

113. The infringement by Facebook of the '593 patent was, and continues to be, deliberate and willful, entitling Sound View to increased damages under 35 U.S.C. § 284 and to attorney fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

### **COUNT SEVEN**

#### **INFRINGEMENT OF THE '860 PATENT**

114. Sound View incorporates by reference the preceding paragraphs as if fully set forth herein.

115. The '860 patent generally relates to an apparatus for use in a computer network that utilizes augmentation files to customize web content based on the client device that is seeking to access the web content by using a pattern matching process to alter component structures of the web content. Within the context of the '860 patent, augmentation files are associated with a particular client type, and the term "augmentation file" is intended to include an interpolation script, auxiliary content file, patch file or any other type of information which defines a translation, alteration or other type of interpolation for at least a portion of a requested

piece of web content. Within the context of the '860 patent, the term "web content" is intended to include one or more documents, files, web pages or any other type of arrangement of information accessible over the World Wide Web, over other portions of the Internet, or over communications networks.

116. At the time of the '860 patent invention, a known web proxy server was the Spyglass Prism proxy server that used a Document Object Model (DOM) to represent the structure of a web page, and a template to describe the web content to be extracted and modified. One disadvantage with the Spyglass Prism approach that the '860 patent sought to solve was that the DOM and the template resided on the proxy server itself, which made it necessary for the content provider to own or to control the proxy server in order to keep the templates synchronized with changes in web content. In addition, the DOM and template formats were sufficiently complicated that specially trained personnel typically had to hand craft those modules for proxy customers.

117. The '860 patent provided a computer-based solution that was able to interpolate web content so as to provide customized web pages to a wide variety of different clients in an efficient manner without requiring the content provider to own or control the web proxy server.

118. In accordance with another aspect of the '860 patent, the proxy server alters the retrieved web content by first parsing the web content into one or more component structures, and then applying a pattern matching process to recognize designated component structures subject to alteration in accordance with the augmentation file(s). The pattern matching process may utilize a pattern matching expression which includes, for example, context, pattern, precedence and/or replacement elements.

119. In violation of 35 U.S.C. § 271, Facebook has infringed at least claims 1-3, 5, 7,

8-10, 13, and 18 of the '860 patent by having made, designed, offered for sale, sold, provided, used, maintained, and/or supported services that use WURFL (Wireless Universal Resource File). Facebook's infringement is continuing.

120. On February 4, 2016, Sound View informed Facebook that all Facebook systems that utilize WURFL infringe the '860 patent. However, Facebook has not stopped infringing.

121. WURFL is a device description repository which serves as a central source of device information for mobile web applications.

122. Facebook uses WURFL to map a given mobile device to its capabilities. Using WURFL, Facebook has created a framework for adapting applications to different mobile platforms and devices. The framework allows Facebook's engineers to focus on building core technology while the framework handles rendering it appropriately for each device. Facebook's framework takes into account different hardware characteristics and display capabilities of end-users' devices including, for example, varying screen resolution and size and whether a user's device has a keypad or touchpad, and then adapts its mobile website to show the right user interface for the particular device accessing the Facebook content.

123. For example, Facebook infringes claim 1 by:

- a. an apparatus for use in a computer network, the apparatus comprising:
- b. at least one server within the network (such as a Facebook application or web server that accesses WURFL), the server being operative to process a client request (such as a request for a Facebook webpage) generated by a client device (such as a user's laptop or mobile device) to determine a particular client type associated with the client device, to retrieve web content (such as a Facebook webpage) identified in the client request, to retrieve one or more augmentation files (such as a WURFL file) associated with at least one of the web content and

the particular client type (such as a mobile device), and to alter the retrieved web content in accordance with the one or more augmentation files, wherein the altered web content is delivered to the client device;

c. wherein the server parses the retrieved web content into one or more component structures (such as buttons, tables, links or textboxes), and subsequently applies a pattern matching process to recognize designated component structure (such as buttons, tables, links or textboxes) subject to alteration in accordance with the one or more augmentation files (such as WURFL files); and

d. wherein the pattern matching process comprises comparing a given one of the component structures (such as buttons, tables, links or textboxes) of the retrieved web content to predetermined component structures represented by respective tokens in the one or more augmentation files (such as the capability information in the WURFL files).

124. Facebook has infringed and continues to infringe the '860 patent, which has damaged Sound View. Sound View is entitled to recover from Facebook the damages sustained by Sound View as a result of Facebook's wrongful acts in an amount adequate to compensate Sound View for Facebook's infringement pursuant to 35 U.S.C. § 284. In committing these acts of infringement, Facebook acted despite an objectively high likelihood that its actions constituted infringement of a valid patent, and Facebook actually knew or should have known that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

125. The infringement by Facebook of the '860 patent was, and continues to be, deliberate and willful, entitling Sound View to increased damages under 35 U.S.C. § 284 and to attorney fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

**RELIEF REQUESTED**

126. Wherefore, Sound View respectfully requests that this Court enter judgment

against Facebook as follows:

- a) that Facebook has infringed each of the Patents-In-Suit;
- b) that Facebook's infringement of the '845 patent, '371 patent, '181 patent, '786 patent, '486 patent, '593 patent, and '860 patent is willful;
- c) that Sound View be awarded damages in accordance with 35 U.S.C. § 284, and, if necessary to adequately compensate Sound View for Facebook's infringement, an accounting;
- d) that this case is exceptional under 35 U.S.C. § 285;
- e) that Sound View be awarded the attorney fees, costs, and expenses that it incurs in prosecuting this action; and
- f) that Sound View be awarded such further relief at law or in equity as the Court

deems just and proper.

**DEMAND FOR JURY TRIAL**

Sound View hereby demands trial by jury on all claims and issues so triable.

Dated: February 29, 2016

PHILLIPS, GOLDMAN & SPENCE, P.A.

/s/ John C. Phillips, Jr.

John C. Phillips, Jr. (No. 110)

Megan C. Haney (No. 5016)

1200 North Broom Street

Wilmington, Delaware 19806-4204

Telephone: (302) 655-4200

jcp@pgslaw.com

mch@pgslaw.com

*Of Counsel:*

Alan S. Kellman

Tamir Packin

DESMARAIS LLP

230 Park Avenue

New York, NY 10169

(212) 351-3400

(212) 351-3401

akellman@desmaraisllp.com

tpackin@desmaraisllp.com

*Attorneys for Plaintiff Sound View Innovations,  
LLC*