IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

SOUND VIEW INNOVATIONS, LLC, Plaintiff, v. CIGNA CORPORATION, AND CIGNA HEALTH AND LIFE INSURANCE COMPANY Defendant.	)	C.A. No. _____ **JURY TRIAL DEMANDED**

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Sound View Innovations, LLC ("Sound View"), for its Complaint for Patent Infringement against Cigna Corporation and Cigna Health and Life Insurance Company (collectively, "Cigna"), alleges as follows:

INTRODUCTION

1. Sound View is an intellectual property licensing company with a patent portfolio including approximately 350 active U.S. Patents. Those patents were developed by researchers at Alcatel Lucent ("Lucent") and its predecessors. Lucent was home to the world-renowned Bell Laboratories, which has a long and storied history of innovation. Researchers at Lucent's Bell Laboratories developed a wide variety of key innovations that have greatly enhanced the capabilities and utility of computer systems and networks. This has resulted in benefits such as better and more efficient computer networking, computer security, and user experiences.

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 that property for their own use. Cigna has used, and continues to use, Sound

View's patents without authorization. Moreover, despite Sound View's attempt to negotiate, Cigna refuses to take a license though it continues to use Sound View's property.

NATURE OF THE CASE

3. This action arises under 35 U.S.C. § 271 for Defendant's infringement of Sound View's United States Patent Nos. 6,502,133 (the "'133 patent"), 6,725,456 (the "'456 patent"), 7,426,715 (the "'715 patent"), 5,806,062 (the "'062 patent"), and 6,125,371 (the "'371 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. On information and belief, Defendant Cigna Corporation is a Delaware corporation, with its principal place of business at 900 Cottage Grove Road, Bloomfield, CT 06002. Cigna Corporation may be served with process by serving its registered agent, The Corporation Trust Company, at Corporation Trust Center, 1209 Orange St., Wilmington, Delaware 19801.

6. On information and belief, Defendant Cigna Health and Life Insurance Company is a Connecticut Company, with its principal place of business at 900 Cottage Grove Road, Bloomfield, CT 06002. Cigna Health and Life Insurance Company maintains a regular and established place of business at 300 Bellevue Parkway, Wilmington, DE 19809.

JURISDICTION AND VENUE

7. 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).

8. Venue is proper in this Court pursuant to 28 U.S.C. §§ 1391(b) and (c) and 1400(b) as to Cigna Corporation, at least because Cigna Corporation resides in this judicial district.

9. Venue is proper in this Court pursuant to 28 U.S.C. §§ 1391(b) and (c) and 1400(b) as to Cigna Health and Life Insurance Company, at least because Cigna Health and Life Insurance Company has a regular and established place of business in this judicial district, and has committed acts of infringement in this judicial district.

10. This Court has personal jurisdiction over Cigna Corporation because it, among other things: is incorporated under the laws of the State of Delaware and has placed services that practice the claims of the Patents-in-Suit into the stream of commerce with the knowledge, or reasonable expectation, that actual or potential users of such services were located within this judicial district.

11. This Court has personal jurisdiction over Cigna Health and Life Insurance Company because it, among other things: transacts business and performs work or service in Delaware; contracts to supply services or things in Delaware; has infringed the patents-in-suit in Delaware; has infringed the patents-in-suit outside of Delaware and regularly does or solicits business in Delaware; and/or has an interest in, uses, or possesses real property in Delaware. Moreover, Cigna Health and Life Insurance intend to serve the Delaware market, such as by maintaining a physical place of business in the state, and have introduced services that practice the claims of the Patents-in-Suit into the stream of commerce with the knowledge, or reasonable expectation, that actual or potential users of such services were located within Delaware.

THE PATENTS-IN-SUIT

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

A. The ’133 Patent

13. The ’133 patent, titled “Real-Time Event Processing System with Analysis Engine Using Recovery Information,” was duly and properly issued by the USPTO on December 31, 2002.

A copy of the ’133 patent is attached hereto as Exhibit A.

14. Sound View is the owner and assignee of the ’133 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

15. The ’133 patent generally relates to real-time event processing in applications such as telecommunications and computer networks, and more particularly, to a method, apparatus, and system for processing events in a real-time analysis engine, and storing recovery information in a main-memory database system associated with the real-time analysis engine.

16. At the time of the invention of the ’133 patent, high performance real-time event processing applications had performance requirements that could not be met by conventional general purpose database management systems. For example, some real-time event processing applications required the service time for such events to not exceed a few milliseconds. However, with conventional database technology, the service time costs of invoking a structured query language operation over a client-server interface, or the service time costs associated with a single access to secondary storage, could account for hundreds of milliseconds. These limitations led real-time event processing applications instead to rely on the use of custom database systems.

17. These custom database systems had disadvantages: (1) there was a high cost of developing and maintaining custom systems; (2) those high costs could not be amortized across a number of different applications; and (3) custom database systems were generally inflexible and difficult to adapt to unforeseen or evolving requirements.

18. At the time of the invention of the ’133 patent, a need therefore existed for an improved real-time event processing system that could provide the performance benefits of custom database systems, but without sacrificing the flexibility and maintainability typically associated with conventional general-purpose database systems.

19. The inventors of the ’133 patent solved that discrete computer-based problem and improved upon the existing real-time event processing systems by providing a real-time event processing system that avoids the problems associated with custom systems.

20. Using a real-time analysis engine operating in the manner described by the ’133 patent is particularly useful because it can provide transactional access to persistent data, but at the speed of a main-memory system, and it also incorporates a recovery model which stores recovery information in order to facilitate roll-back to a recovery point after a failure.

21. The ’133 Patent claims an improved real-time event-processing system delivering increased performance in telecommunications and computer networks. Conventional event-processing systems were only compatible with specialized custom database systems, which were costly to develop and maintain. The inventions of the ’133 Patent claim an improvement in computer functionality—including a real-time analysis engine that is associated with a main-memory system. By associating a real-time analysis engine with a main-memory system (which is much faster than “secondary” storage used in the prior art), the invention provides the performance benefits of custom database systems with the cost savings and flexibility associated with conventional general-purpose database systems.

22. In accordance with the ’133 patent, recovery information regarding a recovery point for a given real-time analysis engine may be stored in a memory portion of the main-memory database system. This way, the real-time event processing system provides a critical path for event processing that is specifically designed for high performance, while also retaining many desirable features of conventional database systems, including high-level declarative programming interfaces, and the transactional correctness properties of atomicity, consistency, isolation and durability. These features of the ’133 patent enhance the reliability, robustness, usability and

maintainability of the real-time event processing system and any applications built thereon.

B. The '456 Patent

23. The '456 patent, titled "Methods and Apparatus for Ensuring Quality of Service in an Operating System," was duly and properly issued by the USPTO on April 20, 2004. A copy of the '456 patent is attached hereto as Exhibit B.

24. Sound View is the owner and assignee of the '456 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

25. The '456 patent generally relates to computer systems, and more particularly to techniques for providing a desired quality of service ("QoS") for an application running in a computer system.

26. At the time of the invention of the '456 patent, in a typical computer system multiple applications would contend for the same physical resources, such as a central processing unit, memory, and disk or network bandwidth. Conventional time-sharing operating systems could achieve acceptably low response time and high system throughput in some environments, but several trends made resource management techniques of conventional time-sharing operating systems increasingly inappropriate. For example, many workloads began including real-time applications like multimedia, which required that requests be processed within certain performance bounds. Also, a trend towards distributed client-server architectures increased the importance of fairness, i.e., preventing certain clients from monopolizing system resources.

27. The aforementioned trend towards client-server architectures made it necessary to manage resources hierarchically. For example, web servers and other user-level servers often needed mechanisms for processing client requests with specified QoS and/or fairness bounds. However, conventional time-sharing operating systems did not provide such mechanisms.

28. Then-existing proportional share schedulers did not provide satisfactory solutions

to many problems that arose in their adoption in operating systems. For example, proportional share schedulers were proposed without an application programming interface ("API"), since they were not implemented and were evaluated only analytically or in simulations. As a further example, proportional share schedulers that were implemented used an API limited to a given scheduler and resource. As yet another example, proportional share schedulers that simply added resource reservations to conventional objects such as files or sockets did not provide correct sharing semantics, as such proportional share schedulers allowed those objects to be shared inappropriately by different users. As yet another example, proportional share schedulers did not propose how a parent process running on an operating system could limit the resource reservations used by its children processes. Finally, proportional share schedulers would hold resource reservations for processes that terminated abnormally, causing the reserved resource to become permanently unavailable.

29. The inventors of the '456 patent provided a technical solution for ensuring a desired QoS for an application running on an operating system.

30. Using the techniques for providing a desired QoS claimed by the '456 patent is particularly useful because it allows selected applications to isolate their performance and the performance of their corresponding client(s) from CPU, memory, disk, or network traffic overloads caused by other applications. Such a capability is increasingly important for real-time, multimedia, Web, and distributed client-server applications as demands on network resources grow.

C. The '715 Patent

31. The '715 patent, titled "Shutting Down a Plurality of Software Components in an Ordered Sequence," was duly and properly issued by the USPTO on September 16, 2008. A copy of the '715 patent is attached hereto as Exhibit C.

32. Sound View is the owner and assignee of the ’715 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

33. The ’715 patent generally relates to distributed software applications, and more particularly, to management of distributed software applications.

34. At the time of the invention of the ’715 patent, distributed software applications included software components distributed among a plurality of executables (e.g., software capsules or software entities). Each executable contained one or more software components that performed some portion of the functionality of the distributed software application.

35. These distributed software application systems had disadvantages. For example, if the software components of a distributed software application shut down without a pre-planned shutdown sequence, then the distributed software application could leave system resources in an inconsistent state. As a further example, without a proper shutdown sequence, the distributed software application would not properly store state information, release the allocated system resources, and/or update databases.

36. If, during shutdown of a distributed software application divided into a plurality of executables, the distributed software application were to shut down the executables by following a preplanned shutdown sequence for the executables, it would suffer from other shortcomings. One shortcoming of this approach was that executing the shutdown sequence at the executable level did not serve to fully leave the system resources in a consistent state. Another shortcoming was that the shutdown sequence was unable to fully coordinate a shutdown of the executables and software components of the distributed software application divided across a plurality of processors.

37. Thus, a need existed to shut down a distributed software application in a manner

that stores state information, releases system resources, and/or leaves the system resources in a consistent state.

38. The inventors of the ’715 patent solved these discrete computer-based problems and improved upon distributed software application systems by providing, among other things, computing methods for shutting down software components that avoid the problems associated with doing so in conventional distributed software applications.

39. Shutting down distributed software applications in an ordered sequence, in the manner described and claimed by the ’715 patent, was also particularly useful because it allowed the saving of state information, release of allocated system resources, and updating of databases.

D. The ’062 Patent

40. The ’062 patent, titled “Data Analysis System Using Virtual Databases,” was duly and properly issued by the United States Patent and Trademark Office (“USPTO”) on September 8, 1998. A copy of the ’062 patent is attached hereto as Exhibit D.

41. Sound View is the owner and assignee of the ’062 patent and holds the right to sue for and recover all damages for infringement thereof, including past infringement.

42. The ’062 patent generally relates to customizable data processing applications that rely on a combination of reusable software operators, such as initial operators, query operators, terminal operators, and/or external operators, to process source information from a virtual database in a particular schema, such as HTML or XML, and transform that source information into another virtual database having the same schema.

43. Various types of documents may be stored in a computer system, such as word processing files, computer programs, HTML documents, financial files, employee files, etc. When dealing with large or complex files, it is often desirable to analyze or alter the structure and content of the documents; for example, comparing a first version to a second version, or analyzing

dependency relationships between various sections of computer code.

44. In order to aid such analysis, a database may be constructed which contains information describing the structure of the documents. Various database queries may be performed to extract and process information describing the structure of the source documents. A collection of source documents, along with an associated database that describes the structure of the documents, is called a repository.

45. To analyze source document information, it is necessary to process information contained in the repository. A computer program that extracts or converts information from a repository is called an operator. Thus, an operator receives a source document and/or a database as input, processes the input, and produces some output. A simple example of an operator is a program that takes a source document as input and counts the number of occurrences of a particular word, and outputs a number containing the number of times the particular word occurs. The overall function of the analysis—in the above example, a count of the number of occurrences of a particular word—is called an application.

46. At the time of the invention of the ’062 patent, in existing repository analysis systems, operators were designed for single applications. Thus, the user indicated which operator he/she wished to apply to the repository, and the system processed the repository accordingly. The user was presented with the output when the processing was finished. Different operators processed the repository in different manners, but there was no convenient mechanism for combining the various operators to create new applications. Thus, when a new application was desired, a new operator would need to be designed from scratch.

47. Prior art repository analysis systems generally were closed systems, in that all operators were applied within the confines of the system, and all database accesses were performed

within the system. For example, a repository analysis system operator may have produced as output a file containing information about the structure of a computer program. In conventional closed systems, this output could not be further processed by, for example, an external graphics program that would format the output in a desired manner. Instead, the output could only be formatted according to operators that were internal to the repository system. There was no convenient mechanism to allow the repository analysis system to communicate with operators that were external to the system.

48. The inventors of the ’062 patent solved these discrete computer-based problems by providing an apparatus and method for creating data analysis applications using reusable, compatible and interoperable software operators, configured to receive a particular data structure with a specific format, process data and generate an output with the same data structure and format as the input. For example, query operators receive data in a particular virtual database format, process the data in the virtual database, and output the results of the processing in another virtual database that has the same format as the original virtual database. A plurality of query operators can be combined to customize the processing of the data. In addition, initial operators convert source information into the virtual database format so that the query operators can analyze the source data. External operators take an external format as input and create another external format as output. Also, terminal operators are used to convert a virtual database into an external format. A user can combine initial, query, terminal, and external operators to create customizable data processing applications.

49. The ’062 Patent is directed to a technical improvement in software technology over the rigid general purpose data analysis applications and expensive custom applications that existed in the 1990s. The novel software structure of the claimed inventions enabled users to engineer their

own purpose-built data analysis applications with reusable interoperable software operators.

50. Creating data analysis applications using reusable software operators, as described in the ’062 patent, is particularly useful in that the external format data may be processed in various ways, thus allowing flexible presentation of the analysis results.

E. The ’371 Patent

51. 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 E.

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

53. 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 a current utilization or capacity of memory, a trend analysis of a utilization or capacity of memory over a time period, or any other applied mathematics- or statistics-based analysis, including a comparison of any of the same with a threshold, ceiling/floor, limit, set point, or the like.

54. 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 because of the latencies inherent with non-volatile memory transactions.

55. 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.

56. 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 no 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.

57. In particular, to solve this discrete computer-centric problem, the ’371 patent teaches, among other things, a method that includes assigning a time stamp to transactions to be performed on the database, and may be assigned as a function of a time stamp counter. Time stamping operates to preserve an order of the transactions. The method also includes creating multiple versions of data records of the database that are affected by update transactions. The method further includes monitoring at least one measurable characteristic and deleting prior ones of the multiple data record versions in response to the time stamp and the at least one measurable characteristic to thereby increase the data capacity of the database, thus increasing memory capacity.

58. The ’371 Patent discloses a technical solution to the inefficiencies associated with multi-versioning in computer databases—i.e., the process by which a database creates multiple versions of the same file. Creating multiple versions of the same file increased database speed and integrity, but did so at the expense of memory (in which those multiple file versions had to be stored). The claimed inventions introduced a new scheme for storing and organizing multiple versions of the same file based on timestamps and a measurable characteristic of the memory, thereby leveraging the advantages of multi-versioning while conserving memory resources.

59. The monitoring of memory utilization as embodied in the ’371 patent allows DBMs to avoid continuously expending memory resources collecting and aging older, no longer needed data record versions.

BACKGROUND FACTS

60. On August 13, 2018, Sound View notified Cigna of its infringement of the '062, '371, '133, and '456 patents via a letter. Sound View additionally notified Cigna of representative Cigna systems that infringe those patents and explained its intention to allow Cigna 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.

61. Cigna did not respond to Sound View's August 13, 2018 letter.

62. On April 17, 2019, Sound View notified Cigna of its infringement of the '715 patent via a letter. Sound View also notified Cigna of representative Cigna systems that infringe that patent.

63. Despite Sound View's efforts, Cigna has refused to engage in any discussion about reaching a licensing agreement to end its infringement of Sound View's patents. Instead, Cigna continues to knowingly, intentionally, and willfully infringe Sound View's patents so as to obtain significant benefits without paying any compensation to Sound View. Sound View thus has no other choice but to seek relief through litigation.

COUNT ONE
INFRINGEMENT OF THE '133 PATENT

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

65. The '133 patent is valid and enforceable.

66. On information and belief, Cigna uses and has used a framework known as Apache Spark ("Spark"), a unified analytics engine for large-scale data processing, to perform Fast Batch Processing, Machine Learning, and Stream Processing (the "Cigna Spark Services"). For example, Cigna's use of Spark has been openly advertised by Cigna employees.

67. On information and belief, the Cigna Spark Services streaming workflow has three high-level stages. In the first stage, data is ingested from streaming data sources like Kafka or Flume, or from static data sources like HBase or MySQL. In the second, Spark is used to perform Machine Learning on ingested data through an application program interface ("API"), and Spark SQL is used to perform further operations on this data. In the third, the streaming output can be stored in various data storage systems like HBase, Cassandra, or HDFS.

68. These workflows are integrated with Cigna's infrastructure, such as its database systems, messaging systems, and monitoring/alerting systems. Events are generated by various Cigna system applications, such as discovery, real-time analytics, personalization, search, and revenue optimization. When these system applications generate events, these events are grouped into Streams.

69. Internally, Spark Streaming works by receiving live input data streams and dividing the data into batches, which are then processed by the Spark engine. It supports real-time processing of fast moving, streaming data.

70. Spark offers an abstraction called resilient distributed datasets (RDDs) to support applications efficiently. RDDs can be stored in memory between queries without requiring replication. Instead, they rebuild lost data on failure using lineage: each RDD remembers how it was built from other datasets to rebuild itself.

71. Spark also uses executors, which are worker nodes' processes that run individual tasks in a given Spark job. They are launched at the beginning of a Spark application, and provide in-memory storage for RDDs that are cached by user programs through a Block Manager.

72. On information and belief, Cigna also uses and has used a framework known as Apache Storm ("Storm") to perform stream processing of events in real-time and continuous data

processing, including database updates and processing messages. Cigna's systems based on Storm include, without limitation, those that stream data from Kafka to HDFS, HBase, and Hive by integrating with Storm (the "Cigna Storm Services").

73. The Cigna Storm Services' architectures are composed of three components: (1) "Streams," which are unbounded sequences of tuples that are processed; (2) "Spouts," which are sources of streams, and (3) "Bolts," which are responsible for processing the Streams in real-time.

74. Those services are integrated with Cigna's infrastructure, such as its database systems, messaging systems, and monitoring/alerting systems. Events are generated by various Cigna system applications, such as discovery, real-time analytics, personalization, search, and revenue optimization. When these system applications generate events, these events are grouped into Streams.

75. Spouts emit Streams into the topology, so that they can subsequently be processed.

76. Bolts are real-time analysis engines that process the Streams. Bolts are capable of performing simple stream transformations, and multiple Bolts are used for more complex stream transformations.

77. The Cigna Storm Services systems have the capability to save and retrieve the state of the Bolts within in-memory storage. For example, Storm has a default in-memory based state implementation and also a Redis backed implementation that provides state persistence. This main-memory database within Storm is used for state management, allowing Storm to automatically and periodically take snapshots of the state of the Bolts.

78. On August 13, 2018, Sound View informed Cigna that its systems and applications infringe the '133 patent. However, Cigna did not stop infringing.

79. Cigna infringed one or more claims of the '133 patent under 35 U.S.C. § 271(a),

literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, products and/or methods encompassed by those claims, including for example, by making, using, selling, offering for sale, and/or importing servers and products, such as Cigna's servers used for real-time analytics and real-time processing, that include or use applications based on Apache Spark.

80. For example, Cigna infringed claim 13 by using a method of processing events (such as input data streams) generated by at least one system application, the method comprising the steps of:

a. processing the events in at least one real-time analysis engine (such as a Spark engine); and

b. storing in a main-memory database system (such as an executor) associated with the real-time analysis engine recovery information (such as RDDs) regarding a recovery point for the real-time analysis engine.

81. Cigna also infringed one or more claims of the '133 patent under 35 U.S.C. § 271(a), literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, products and/or methods encompassed by those claims, including for example, by making, using, selling, offering for sale, and/or importing servers and products, such as Cigna's servers used for real-time analytics and real-time processing, that include or use applications based on Apache Storm.

82. For example, Cigna infringed claim 13 by using a method of processing events (such as Streams) generated by at least one system application (such as the Cigna Storm Services), the method comprising the steps of:

a. processing the events in at least one real-time analysis engine (such as a Bolt); and

b. storing in a main-memory database system (such as Storm's default in-memory based state implementation) associated with the real-time analysis engine recovery information regarding a recovery point for the real-time analysis engine (such as the state information relating to the Bolt's state).

83. Sound View has been damaged by Cigna's infringement of the '133 patent and is entitled to recover from Cigna the damages sustained by Sound View as a result of Cigna's wrongful acts in an amount adequate to compensate Sound View for Cigna's infringement subject to proof at trial.

84. In committing these acts of infringement, Cigna committed egregious misconduct including, for example, acting despite knowing that its actions constituted infringement of a valid patent, or recklessly disregarding the fact that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

85. Cigna's infringement of the '133 patent was 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 '456 PATENT

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

87. The '456 patent is valid and enforceable.

88. Cigna has used software known as Apache Hadoop YARN ("Yarn") in its data systems. For example, Cigna's use of Yarn has been openly advertised by Cigna and its

employees, and includes, without limitation, management of cluster resources on HDP 2.0 clusters via Yarn (the "Cigna Yarn Services").

89. Yarn is the architectural center of Hadoop that allows multiple data processing engines such as interactive SQL, real-time streaming, data science and batch processing to handle data stored in a single platform. Yarn provides resource management and a central platform to deliver consistent operations, security, and data governance tools across Hadoop clusters.

90. Yarn splits up the functionalities of resource management and job scheduling into separate daemons, by having a global ResourceManager ("RM") and per-application ApplicationMaster ("AM"). The RM is the ultimate authority that arbitrates resources among all the applications in the system. The per-application AM is, in effect, a framework specific library and is tasked with negotiating resources from the RM and working with the NodeManager(s) to execute and monitor the tasks. Yarn provides the ability to preempt certain applications in order to make room for other more time-sensitive or higher priority applications.

91. Within Yarn, the fundamental unit of scheduling is a queue. The capacity of each queue specifies the percentage of cluster resources that are available for applications submitted to the queue. Yarn uses a hierarchy of queues wherein each leaf (child) queue is tied to a single parent queue. Parent queues contain more parent queues or leaf queues but do not themselves accept any application submissions directly. Child queues live under a parent queue and accept applications.

92. A user may launch an application on Yarn using the YarnClient and ContainerLaunchContext APIs. New clients define all the information needed by the RM to launch the AM, which includes the application ID, name, queue, and priority information. ContainerLaunchContext is used to define the container in which the AM will be launched and

run. It defines all required information needed to run the application, including resources and environmental settings. ContainerLaunchContext includes resource requirements such as memory and vCores. Moreover, helper APIs convert values obtained from the environment into objects.

93. Additionally, Yarn's Cluster Reservation Submit API can be used to submit reservations. When the reservation is made, the user can use the reservation ID used to submit the reservation to get access to the resources by specifying it as part of a Cluster Submit Applications API. The Cluster Submit Applications Object includes a resource object, which includes memory and vCore requirements for each container.

94. Yarn's RM includes a Fair Scheduler and Capacity Scheduler, which allow assigning guaranteed minimum shares to queues. When an API submits a reservation, it is validated by the RM, which returns a reservation ID and creates reservable queues. RM's schedulers then provide containers, giving a user guaranteed access to the required resources, as identified by objects, in accordance with capacity and fairness sharing protocols.

95. Yarn's Fair Scheduler and Capacity Scheduler guarantee minimum resource reservations, e.g., memory and/or vCores, to queues. If a queue's minimum share is not satisfied, it will be offered available resources before any other queue under the same parent. Fair Scheduler uses hierarchical queues, such that queues are sibling queues when they have the same parent. Associated with each queue is a weight, which determines the amount of resources a queue deserves in relation to its sibling queues. This amount is known as Steady FairShare, which is calculated at the queue level. For the root queue, the Steady FairShare is equal to all the cluster's resources. The Steady FairShare is calculated such that the minimum amount of resources associated with the parent queue is at least equal to the sum of the minimum resources associated with each of the parent's children.

96. Cigna infringes and has infringed one or more claims of the ’456 patent under 35 U.S.C. § 271(a), literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, products and/or methods encompassed by those claims, including for example, by making, using, selling, offering for sale, and/or importing systems and platforms that include or use the Cigna Yarn Services.

97. On August 13, 2018, Sound View informed Cigna that its systems and applications infringe the ’456 patent. However, Cigna has not stopped infringing.

98. For example, Cigna infringes at least claim 13 by using a method of ensuring a particular quality of service for an application in a computer system, the method comprising the steps of:

a. utilizing an application programming interface of an operating system to establish one or more quality of service guarantees that correspond to a reference to an object (such as the YarnClient, ContainerLaunchContext, and/or Cluster Reservation Submit APIs) ; and

b. providing a particular quality of service to a request in accordance with the one or more quality of service guarantees that correspond to one or more object references used in the request (such as through use of Yarn’s Fair Scheduler and/or Capacity Scheduler); and

c. wherein the quality of service guarantees comprise resource reservations, each specifying a portion of a resource set aside for exclusive use by one or more processes (such as memory and vCores); and

d. wherein the resource reservations are organized hierarchically such that each resource reservation r may have at most one parent and one or more siblings and children, and associated with r is a weight that specifies how r shares the resources of r’s parent with r’s

siblings (such as the hierarchical queues used by Yarn's Fair Scheduler and Capacity Scheduler); and

e. wherein associated with each resource reservation r is a minimum amount of resources that r receives from its parent p, such that the minimum amount of resources associated with p is at least equal to the sum of the minimum amount of resources associated with each of p's children (such as the Steady FairShare of resources).

99. Sound View has been damaged by Cigna's infringement of the '456 patent and is entitled to recover from Cigna the damages sustained by Sound View as a result of Cigna's wrongful acts in an amount adequate to compensate Sound View for Cigna's infringement subject to proof at trial.

100. In committing these acts of infringement, Cigna committed egregious misconduct including, for example, acting despite knowing that its actions constituted infringement of a valid patent, or recklessly disregarding the fact that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

101. Cigna's infringement of the '456 patent was and is 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 '715 PATENT

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

103. The '715 patent is valid and enforceable.

104. Cigna has used software known as Apache Ambari ("Ambari") in its data systems. For example, Cigna's use of Ambari has been openly advertised by Cigna and its employees, and

includes, without limitation, Hadoop Cluster Management via Ambari (the "Cigna Ambari Services").

105. Ambari is a management platform for provisioning, managing, monitoring, and securing Apache Hadoop Clusters. Ambari allows Cigna to manage service dependencies and shutdown sequences in Hadoop, making it easier to provide ongoing cluster maintenance and management.

106. Ambari managed services include metainfo, as defined in an XML file, which is a declarative definition of an Ambari managed service describing its content. Those services further include additional files that define, for example, dependencies between managed services.

107. Services in Ambari are defined in its stacks folder. A "stack" is a collection of services.

108. Within Ambari, "role" is another name for a component. Each service can define its own role command order by including the Role Command Order file in its service folder. Furthermore, Ambari includes extensions that include role command orders based on default dependencies. On information and belief, Cigna can and has specified the order in which components are run by including the Role Command Order file in the stack version folder.

109. Ambari is responsible for starting and stopping components or services of Hadoop Clusters such as HDFS, Hbase, and Zookeeper. Communications between Hadoop nodes generally occur using Remote Procedure Call ("RPC") as the mechanism. RPC communications between nodes are commonly layered on top of the TCP/IP protocol, as are other protocols for communication between nodes. Using the Role Command Order, Ambari shuts down software components according to an ordered sequence. For example, the "Stop" command ensures that HBase Master Servers and HBase Region Servers are stopped before Zookeeper servers, and that

communication channels between components are torn down.

110. As another example, HDFS establishes communication channels between various software components in a distributed environment by using a NameNode server. In addition, an HDFS cluster includes a number of DataNodes, usually one per node in the cluster, which manage storage attached to the nodes that they run on. When a software component is shutdown, the communication channel to the NameNode is torn down.

111. Cigna infringes and has infringed one or more claims of the ’715 patent under 35 U.S.C. § 271(a), literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, products and/or methods encompassed by those claims, including for example, by making, using, selling, offering for sale, and/or importing systems and platforms that include or use the Cigna Ambari Services.

112. On April 17, 2019, Sound View informed Cigna that its systems and applications infringe the ’715 patent. However, Cigna did not stop infringing.

113. For example, Cigna infringed claim 19 by using a method, comprising the steps of:

a. obtaining one or more dependency relationships among a plurality of software components that run within one or more executables of a distributed software application (such as those dependencies defined in metainfo);

b. establishing an ordered sequence for shutdown of the plurality of Software components based on one or more of the one or more dependency relationships (such as the sequence defined in Role Command Order); and

c. shutting down the plurality of Software components according to the ordered sequence (such as in response to the Stop command using the Role Command Order); and

d. tearing down any communication channels between the plurality of software components upon deactivation of each of the plurality of software components (such as by tearing down communication channels between services such as Zookeeper, HBase, and HDFS).

114. Sound View has been damaged by Cigna's infringement of the '715 patent and is entitled to recover from Cigna the damages sustained by Sound View as a result of Cigna's wrongful acts in an amount adequate to compensate Sound View for Cigna's infringement subject to proof at trial.

115. In committing these acts of infringement, Cigna committed egregious misconduct including, for example, acting despite knowing that its actions constituted infringement of a valid patent, or recklessly disregarding the fact that its actions constituted an unjustifiably high risk of infringement of a valid and enforceable patent.

116. Cigna's infringement of the '715 patent was and is 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 '062 PATENT

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

118. The court in *Sound View Innovations, LLC. v. Hulu*, LLC, No. 2:17-cv-04146 (C.D. Cal.), recently found claim 14 of the '062 patent invalid under 35 U.S.C. § 101. Civil Minutes - General at 12, *Sound View Innovations, LLC. v. Hulu, LLC*, No. 2:17-cv-04146 (C.D. Cal. Apr. 30, 2019), ECF No. 451. That determination is not final, and Sound View continues to assert its rights in the '062 patent. However, to avoid unnecessary expense Sound View is willing to

postpone or stay resolution of its claims with respect to the ’062 patent until that finding is vacated, reversed, or otherwise disposed of.

119. Cigna’s web pages and internet services, including at least cigna.com, and my.cigna.com (the “Cigna DOM Services”), have used the Document Object Model (“DOM”) to create and process customizable data analysis and processing applications. The DOM is an application programming interface (“API”) that allows documents to be modelled using objects of a variety of data formats, including HTML and XML. It defines the logical structure of documents and the way a document is accessed and manipulated.

120. Using the DOM, the nodes (or objects) of every document are organized in a tree structure, called the “DOM tree,” and can be manipulated individually using the DOM methods (or operators). With the DOM, programmers can build documents, navigate their structure, and add, modify, or delete elements and content. Anything found in an HTML or XML document can be manipulated in this way using the DOM, with a few exceptions.

121. As an object model, the DOM identifies: (1) the interfaces and objects used to represent and manipulate a document; (2) the semantics of these interfaces and objects – including both behavior and attributes of the relationships; and (3) collaborations among these interfaces and objects.

122. jQuery is a DOM manipulation library that makes it easier to use JavaScript on a website by taking more complex code needed to manipulate the DOM and wrapping the code into simpler methods that can be called with smaller amounts of JavaScript.

123. On information and belief, Cigna has used jQuery throughout its websites, including at least the Cigna DOM Services.

124. Cigna has infringed one or more claims of the ’062 patent under 35 U.S.C. § 271(a),

literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, products and/or methods encompassed by those claims, including for example, by making, using, selling, offering for sale, and/or importing it's platforms, web pages, and servers, including for example the Cigna DOM Services, which have used jQuery.

125. For example, Cigna has infringed claim 14 by using a method for processing information (such as the Cigna DOM Services) comprising the steps of:

a. providing a plurality of software operators (such as jQuery methods, including, for example, ".toggleClass( )," ".attr( )," ".css( )," ".find( )," ".wrapall( )," ".append( )," ".hide( )," and ".removeClass( )," each configured to receive a virtual database (such as DOM nodes (or objects) or web pages, describing the structure of a document) having a first schema (such as HTML or XML), for processing information contained in said virtual database (such as by applying a jQuery method to a node in the DOM tree), and for outputting a virtual database having said first schema; and

b. combining at least two of said software operators to create an application (such as that used to construct and serve the Cigna DOM Services).

126. Sound View has been damaged by Cigna's infringement of the '062 patent and is entitled to recover from Cigna the damages sustained by Sound View as a result of Cigna's wrongful acts in an amount adequate to compensate Sound View for Cigna's infringement subject to proof at trial.

COUNT FIVE
INFRINGEMENT OF THE '371 PATENT

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

128. The Patent Trial and Appeal Board (the "PTAB") in IPR Nos. 2018-00017, 2018-00096, and 2018-00366 recently found, collectively, claims 1–3 and 8–10 of the '371 patent unpatentable under 35 U.S.C. § 103. *See Hulu, LLC. v. Sound View Innovations, LLC*, Case IPR2018-00017, Paper 48 (Apr. 9, 2019); *Unified Patents Inc. v. Sound View Innovations, LLC*, Case IPR2018-00096, Paper 41 (Apr. 9, 2019); *Hulu, LLC v. Sound View Innovations, LLC*, Case IPR2018-00366, Paper 41 (Apr. 9, 2019). Sound View has filed notices of appeal at the PTAB, and its appeal has been docketed at the Court of Appeals for the Federal Circuit. Given the PTAB's determinations, though it continues to assert its rights in the '371 patent, to avoid unnecessary expense Sound View is willing to postpone or stay resolution of its claims with respect to the '371 patent until the PTAB's decision is vacated, reversed, or otherwise disposed of.

129. On information and belief, Cigna has used the Cassandra database in its data systems, including without limitation, in NoSQL clusters within Cigna's Big Data infrastructure (the "Cigna Cassandra Services"). Additionally, current and former employees of Cigna have openly advertised Cigna's use of the Cassandra database.

130. The Cassandra database is stored in a memory comprising a combination of "memtable" and "SSTable." A memtable is a Cassandra table-specific, in-memory data structure that resembles a write-back cache. A sorted string table (SSTable) is an immutable data file to which Cassandra writes memtables periodically. SSTables are stored on disk sequentially and maintained for each Cassandra table.

131. During a write transaction, a timestamp is assigned to the transaction performed on the Cassandra database.

132. Cassandra databases utilize periodic compaction to manage the accumulation of SSTables.

133. Cassandra databases have configurable parameters (such as min_threshold and max_threshold parameters) that control when a minor compaction occurs.

134. Cigna has used a distributed database known as HBase in its data systems. For example, on information and belief, Cigna platforms, web pages, servers, or products that include or use applications based on the HBase database (the "Cigna HBase Services") include, without limitation, storage within Cigna's Spark & Hadoop ecosystem. Additionally, current and former employees of Cigna have openly advertised Cigna's use of the HBase database.

135. HBase is a column-oriented database management system that runs on top of a Hadoop Distributed File System. Applications store data into HBase 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 Cigna's HBase databases, it is first written to an in-memory store (memstore), and then is flushed into a store file. When Cigna puts data into HBase, a timestamp is required and is generated by HBase. Performing a "put" operation to HBase creates a new version of a cell.

136. Cigna controls the number of versions stored in HBase.

137. During major compaction, excess versions are deleted from the store file. The number of versions to be deleted is determined by comparing the number of versions stored to the MaxVersions. If the number of stored versions of the store files is greater than the MaxVersions, then the excess versions are deleted. The versions that are deleted are selected based on timestamps.

138. During minor compactions, a configurable number of smaller store files are combined into fewer, but larger store files. The store files to be compacted in a minor compaction are determined at least in part by configurable store file number, size, and/or ratio parameters.

During a minor compaction, versions are also deleted based on timestamps.

139. Cigna has infringed one or more claims of the ’371 patent under 35 U.S.C. § 271(a), literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, products and/or methods encompassed by those claims, including for example, by making, using, selling, offering for sale, and/or importing servers and systems that include or use applications based on the Cassandra database, such as the Cigna Cassandra Services.

140. Cigna has also infringed one or more claims of the ’371 patent under 35 U.S.C. § 271(a), literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, products and/or methods encompassed by those claims, including for example, by making, using, selling, offering for sale, and/or importing systems and platforms that include or use applications based on HBase, such as the Cigna HBase Services.

141. On August 13, 2018, Sound View informed Cigna that its systems and applications infringe the ’371 patent. However, Cigna did not stop infringing.

142. For example, Cigna infringes claim 8 by using a method of operating a processing system for use with a database of data records (such as a Cassandra database), said database stored in a memory, comprising the steps of:

a. assigning a time stamp to transactions to be performed on said database (such as assigning a timestamp during a write transaction);

b. creating multiple versions of ones of said data records affected by said transactions that are update transactions (such as creating a new timestamped version of an updated row in the database);

c. monitoring a measurable characteristic of said memory (such as a measurement associated with a min_threshold or max_threshold parameter); and

d. deleting 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 by performing a compaction process in response to the min_threshold parameter being met or exceeded).

143. As another example, Cigna infringes claim 8 by using a method of operating a processing system for use with a database of data records (such as an HBase database), said database stored in a memory, comprising the steps of:

a. assigning a time stamp to transactions to be performed on said database (such as assigning a timestamp during a write transaction);

b. creating multiple versions of ones of said data records affected by said transactions that are update transactions (such as creating a new timestamped version of an updated cell in the database);

c. monitoring a measurable characteristic of said memory (such as the number of versions being stored in the store file, and/or the store file number, size, and/or ratio parameters); and

d. deleting 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 by deleting a version of the cell based on the measurable characteristic and the timestamp of each version).

144. Sound View has been damaged by Cigna's infringement of the '371 patent and is entitled to recover from Cigna the damages sustained by Sound View as a result of Cigna's

wrongful acts in an amount adequate to compensate Sound View for Cigna's infringement subject to proof at trial.

RELIEF REQUESTED

Wherefore, Sound View respectfully requests that this Court enter judgment against Cigna as follows:

a) that Cigna has infringed each of the Patents-in-Suit;

b) that Cigna's infringement of the '133, '456, and '715 patents is and/or has been willful;

c) that Sound View be awarded damages in accordance with 35 U.S.C. § 284, including treble damages and, if necessary to adequately compensate Sound View for Cigna's infringement, an accounting;

d) that this case is exceptional under 35 U.S.C. § 285;

e) that Sound View be awarded the attorney's fees, costs, and expenses that it incurs in prosecuting this action; and

f) that Sound View be awarded further relief at law or in equity as the Court deems just and proper.

DEMAND FOR JURY TRIAL

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

Dated: May 24, 2019

By: */s/ John C. Phillips, Jr.*

Phillips, Goldman, McLaughlin & Hall, P.A.
John C. Phillips, Jr. (#110)
Megan C. Haney (#5016)
1200 N. Broom Street
Wilmington, DE 19806
Tel: (302) 655-4200
Fax: (302) 655-4210

jcp@pgmhlaw.com
mch@pgmhlaw.com

Of Counsel:

DESMARAIS LLP
Alan S. Kellman (*pro hac vice* pending)
Richard M. Cowell (*pro hac vice* pending)
Edward Geist (*pro hac vice* pending)
Kathryn Bi (*pro hac vice* pending)
Carson Olsheski (*pro hac vice* pending)
230 Park Avenue
New York, NY 10169
Tel: (212) 351-3400
Fax: (212) 351-3401
akellman@desmaraisllp.com
rcowell@desmaraisllp.com
egeist@desmaraisllp.com
kbi@desmaraisllp.com
colsheski@desmaraisllp.com

Counsel for Plaintiff Sound View Innovations, LLC