**IN THE UNITED STATES DISTRICT COURT**
**FOR THE EASTERN DISTRICT OF TEXAS**
**TYLER DIVISION**

| | |
|---|---|
| REALTIME DATA LLC d/b/a IXO,<br>Plaintiff,<br>v.<br>ORACLE AMERICA, INC.,<br>Defendant. | Case No. 6:16-cv-88<br><br><br>JURY TRIAL DEMANDED |

**COMPLAINT FOR PATENT INFRINGEMENT AGAINST ORACLE AMERICA, INC.**

This is an action for patent infringement arising under the Patent Laws of the United States of America, 35 U.S.C. § 1 *et seq.* in which Plaintiff Realtime Data LLC d/b/a IXO ("Plaintiff," "Realtime," or "IXO") makes the following allegations against Defendant Oracle America, Inc. ("Oracle"):

**PARTIES**

1. Realtime is a New York limited liability company. Realtime has places of business at 5851 Legacy Circle, Plano, Texas 75024, 1828 E.S.E. Loop 323, Tyler, Texas 75701, and 116 Croton Lake Road, Katonah, New York, 10536, and is organized under the laws of the State of New York. Realtime has been registered to do business in Texas since May 2011. Since the 1990s, Realtime has researched and developed specific solutions for data compression, including, for example, those that increase the speeds at which data can be stored and accessed. As recognition of its innovations rooted in this technological field, Realtime holds over 45 United States patents and has numerous pending patent applications. Realtime has licensed patents in this portfolio to many of the world's leading technology companies. The patents-in-suit relate to Realtime's development of advanced systems and methods for fast and efficient data compression using numerous innovative compression techniques based on, for example, particular

attributes of the data.

2. On information and belief, Defendant Oracle America, Inc. ("Oracle") is a Delaware corporation with its principal office at 500 Oracle Parkway, Redwood City, California 94065. On information and belief, Oracle can be served through its registered agent, Corporation Service Company d/b/a CSC-Lawyers Inco, 211 E. 7th Street Suite 620, Austin, Texas 78701.

## JURISDICTION AND VENUE

3. This action arises under the patent laws of the United States, Title 35 of the United States Code. This Court has original subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

4. This Court has personal jurisdiction over Defendant Oracle in this action because Oracle has committed acts within the Eastern District of Texas giving rise to this action and has established minimum contacts with this forum such that the exercise of jurisdiction over Oracle would not offend traditional notions of fair play and substantial justice. Defendant Oracle, directly and through subsidiaries or intermediaries (including distributors, retailers, and others), has committed and continues to commit acts of infringement in this District by, among other things, offering to sell and selling products and/or services that infringe the asserted patents. Oracle is registered to do business in the State of Texas and has appointed Corporation Service Company d/b/a CSC-Lawyers Inco, 211 E. 7th Street Suite 620, Austin, Texas 78701 as its agent for service of process.

5. Venue is proper in this district under 28 U.S.C. §§ 1391(b), 1391(c) and 1400(b). Oracle is registered to do business in Texas, and upon information and belief, has transacted business in the Eastern District of Texas and has committed acts of direct and indirect infringement in the Eastern District of Texas.

## COUNT I

## INFRINGEMENT OF U.S. PATENT NO. 7,161,506

6. Plaintiff realleges and incorporates by reference paragraphs 1-5 above, as if fully set forth herein.

7. Plaintiff Realtime is the owner by assignment of United States Patent No. 7,161,506 ("the '506 patent") entitled "Systems and methods for data compression such as content dependent data compression." The '506 patent was duly and legally issued by the United States Patent and Trademark Office on January 9, 2007. A true and correct copy of the '506 patent, including its reexamination certificates, is included as Exhibit A.

**Oracle SecureFiles**

8. On information and belief, Oracle has made, used, offered for sale, sold and/or imported into the United States Oracle products that infringe the '506 patent, and continues to do so. By way of illustrative example, these infringing products include, without limitation, Oracle's compression products and services, such as, *e.g.*, Oracle Database 11g Release 2 and Oracle Database 12c, each of which includes Oracle's SecureFiles technology, and all versions and variations thereof since the issuance of the '506 patent ("Accused Instrumentality").

9. On information and belief, Oracle has directly infringed and continues to infringe the '506 patent, for example, through its own use and testing of the Accused Instrumentality to practice compression methods claimed by Claim 104 of the '506 patent, namely, a computer implemented method for compressing data, comprising: analyzing data within a data block of an input data stream to identify one or more data types of the data block, the input data stream comprising a plurality of disparate data types; performing content dependent data compression with a content dependent data compression encoder if a data type of the data block is identified; and performing data compression with a single data compression encoder, if a data type of the data block is not identified, wherein the analyzing of the data within the data block to identify one or

more data types excludes analyzing based only on a descriptor that is indicative of the data type of the data within the data block. Upon information and belief, Oracle uses the Accused Instrumentality to practice infringing methods for its own internal non-testing business purposes, while testing the Accused Instrumentality, and while providing technical support and repair services for the Accused Instrumentality to Oracle's customers.

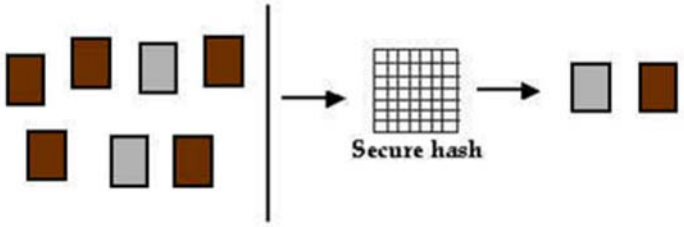
10. The Accused Instrumentality is a computer-implemented method for data compression. This system minimizes the amount of data transmitted over a network and stored on a backup device. The Accused Instrumentality employs several data compression techniques to achieve this goal.

11. The Accused Instrumentality analyzes data within a data block of an input data stream to identify one or more data types of the data block, the input data stream comprising a plurality of disparate data types. *See, e.g.*, http://www.oracle.com/technetwork/database/sfe-092218.html:

ORACLE DATABASE 12c SecureFiles Advanced Features

The advanced features are available only for SecureFiles and do not apply to older LOBs or BasicFiles. The new features in SecureFiles - Deduplication, Compression and Encryption - can be setup independently or as a combination of one or more features. If all three features are turned on, Oracle will perform deduplication first and then compression followed by encryption.

Deduplication



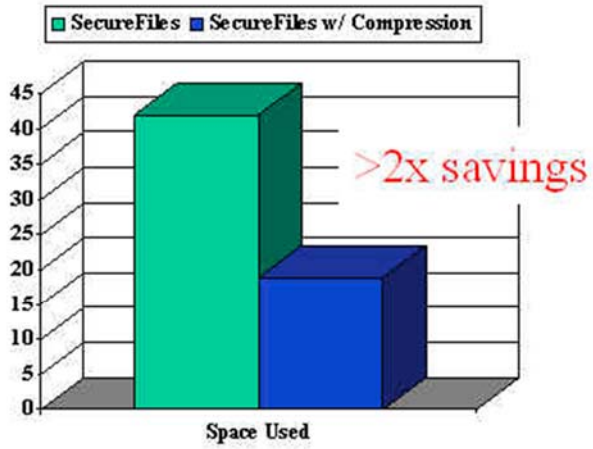

Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. In addition to storing only one copy, SecureFiles maintains 'references' for other duplicates. Deduplication is completely transparent to applications. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment. The lob_storage_clause allows for specifying deduplication at a partition level. Duplicate detection does not span across partitions or subpartitions for partitioned SecureFile columns.

More details on Deduplication can be found in the SecureFiles and Large Object Developer's Guide.

SecureFile Deduplication is part of the Advanced Compression Option.

Compression

SecureFile data is compressed using industry standard compression algorithms. Compresion not only results in significant savings in storage but also improved performance by reducing IO, buffer cache requirements, redo generation and encryption overhead. If the compression does not yield any savings or if the data is already compressed, SecureFiles will automatically turn off compression for such columns. Compression is performed on the server-side and allows for random reads and writes to SecureFile data. SecureFile provides for varying degrees of compression: MEDIUM (default) and HIGH, which represent a tradeoff between storage savings and latency.




Compression tested on the Calgary data set consisting of doc, pdf and text documents. Actual compression might vary depending on the data loaded into SecureFiles. More details on Compression can be found in the SecureFiles and Large Object Developer's Guide

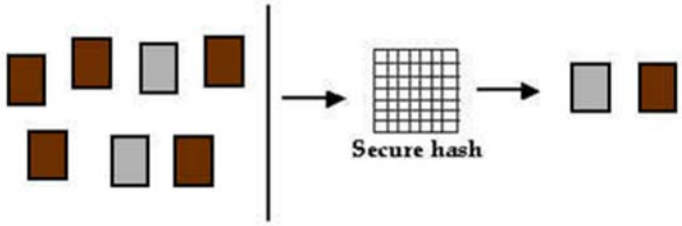
SecureFile Compression is part of the Advanced Compression Option.

12. The Accused Instrumentality performs content dependent data compression with a content dependent data compression encoder if a data type of the data block is identified. *See, e.g.,* http://www.oracle.com/technetwork/database/sfe-092218.html:

ORACLE DATABASE 12c SecureFiles Advanced Features

The advanced features are available only for SecureFiles and do not apply to older LOBs or BasicFiles. The new features in SecureFiles - Deduplication, Compression and Encryption - can be setup independently or as a combination of one or more features. If all three features are turned on, Oracle will perform deduplication first and then compression followed by encryption.

Deduplication




Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. In addition to storing only one copy, SecureFiles maintains 'references' for other duplicates. Deduplication is completely transparent to applications. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment. The lob_storage_clause allows for specifying deduplication at a partition level. Duplicate detection does not span across partitions or subpartitions for partitioned SecureFile columns.

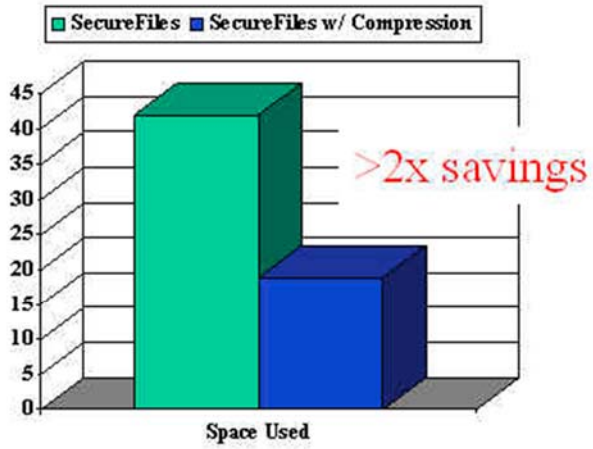
More details on Deduplication can be found in the SecureFiles and Large Object Developer's Guide.

SecureFile Deduplication is part of the Advanced Compression Option.

13. The Accused Instrumentality performs data compression with a single data compression encoder, if a data type of the data block is not identified. *See, e.g.*, http://www.oracle.com/technetwork/database/sfe-092218.html:

Compression

SecureFile data is compressed using industry standard compression algorithms. Compresion not only results in significant savings in storage but also improved performance by reducing IO, buffer cache requirements, redo generation and encryption overhead. If the compression does not yield any savings or if the data is already compressed, SecureFiles will automatically turn off compression for such columns. Compression is performed on the server-side and allows for random reads and writes to SecureFile data. SecureFile provides for varying degrees of compression: MEDIUM (default) and HIGH, which represent a tradeoff between storage savings and latency.




Compression tested on the Calgary data set consisting of doc, pdf and text documents. Actual compression might vary depending on the data loaded into SecureFiles. More details on Compression can be found in the SecureFiles and Large Object Developer's Guide

SecureFile Compression is part of the Advanced Compression Option.

14. In the Accused Instrumentality analyzing of the data within the data block to identify one or more data types excludes analyzing based only on a descriptor that is indicative of the data type of the data within the data block. *See, e.g.*, http://www.oracle.com/technetwork/articles/sql/11g-securefiles-084075.html:

**Deduplication**

Deduplication is likely to be the most popular feature in SecureFiles because it is the most widely sought after benefit of OS files in some high-end filesystems as opposed to database-resident blobs. Suppose a table has five records each with a BLOB. Three of the BLOBs are identical. If it were possible to store the BLOB only once and store only the reference to that copy on other two records, it would reduce the space consumption substantially. This is possible in OS files but would not have been possible in Oracle Database 10*g* LOBs. But with SecureFiles it's actually trivial via a property called deduplication. You can specify it during the table creation or modify it later as:

```
SQL> alter table contracts_sec
  2  modify lob(orig_file)
  3  (deduplicate)
  4  /

Table altered.
```

After the deduplication, the database calculates the hash values of the columns values in each row and compares them to the others. If the hash values match, the hash value is stored&m;dashnot the actual BLOB. When a new record is inserted its hash value is calculated, and if it matches to another value then the hash value is inserted; otherwise the real value is stored.

15. On information and belief, Oracle also directly infringes and continues to

infringe other claims of the '506 patent, for similar reasons as explained above with respect to Claim 104 of the '506 patent.

16. On information and belief, all of the Accused Instrumentalities perform the claimed methods in substantially the same way. In particular, on information and belief, the SecureFiles compression technology used in Oracle Database 12c is similar to the SecureFiles compression technology used in Oracle Database 11g: "SecureFiles is available in Oracle Database 11g and 12c on all supported database platforms. … It is now the default LOB storage in Oracle Database 12c." [1]

17. On information and belief, use of the Accused Instrumentality in its ordinary and customary fashion results in infringement of the methods claimed by the '506 patent.

18. On information and belief, Oracle has had knowledge of the '506 patent since at least August 12, 2011 and/or at least since the filing of this Complaint or shortly thereafter, and on information and belief, Oracle knew of the '506 patent and knew of its infringement, including by way of this lawsuit. In particular, the '506 patent was cited by the Examiner in an Office Action dated August 12, 2011 during prosecution of U.S. Patent Application No. 12/059,393, now U.S. Patent No. 8,208,532, assigned to Oracle.

19. Upon information and belief, Oracle's affirmative acts of making, using, and selling the Accused Instrumentalities, and providing implementation services and technical support to users of the Accused Instrumentalities, have induced and continue to induce users of the Accused Instrumentalities to use them in their normal and customary way to infringe Claim 104 of the '506 patent by practicing a computer implemented method comprising: receiving a data block in an uncompressed form, said data block being included in a data stream; analyzing data within the data block to determine a type of said data block; and compressing said data block to provide a compressed data block,

[1] http://www.oracle.com/technetwork/database/sf-faq-082597.html

wherein if one or more encoders are associated to said type, compressing said data block with at least one of said one or more encoders, otherwise compressing said data block with a default data compression encoder, and wherein the analyzing of the data within the data block to identify one or more data types excludes analyzing based only on a descriptor that is indicative of the data type of the data within the data block. For example, Oracle instructs users of SecureFiles that, "The new features in SecureFiles - Deduplication, Compression and Encryption - can be setup independently or as a combination of one or more features. If all three features are turned on, Oracle will perform deduplication first and then compression followed by encryption. Deduplication: This feature eliminates multiple, redundant copies of SecureFiles data and is completely transparent to applications. Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment. … Compression: Oracle automatically detects if SecureFile data is compressible and will compress using industry standard compression algorithms. If the compression does not yield any savings or if the data is already compressed, SecureFiles will automatically turn off compression for such LOBs. Compression not only results in significant savings in storage but also improved performance by reducing I/O, buffer cache requirements, redo generation and encryption overhead." [2] For similar reasons, Oracle also induces its customers to use the Accused Instrumentalities to infringe other claims of the '506 patent. Oracle specifically intended and was aware that these normal and customary activities would infringe the '506 patent. Oracle performed the acts that constitute induced infringement, and would induce actual infringement, with the

[2] http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf at 7-8.

knowledge of the '506 patent and with the knowledge, or willful blindness to the probability, that the induced acts would constitute infringement. On information and belief, Oracle engaged in such inducement to promote the sales of the Accused Instrumentalities. Accordingly, Oracle has induced and continue to induce users of the accused products to use the accused products in their ordinary and customary way to infringe the '506 patent, knowing that such use constitutes infringement of the '506 patent.

20. By making, using, offering for sale, selling and/or importing into the United States the Accused Instrumentalities, and touting the benefits of using the Accused Instrumentalities' compression features, Oracle has injured Realtime and is liable to Realtime for infringement of the '506 patent pursuant to 35 U.S.C. § 271.

21. As a result Oracle's infringement of the '506 patent, Plaintiff Realtime is entitled to monetary damages in an amount adequate to compensate for Oracle's infringement, but in no event less than a reasonable royalty for the use made of the invention by Oracle, together with interest and costs as fixed by the Court.

## COUNT II

## INFRINGEMENT OF U.S. PATENT NO. 9,054,728

22. Plaintiff Realtime realleges and incorporates by reference paragraphs 1-21 above, as if fully set forth herein.

23. Plaintiff Realtime is the owner by assignment of United States Patent No. 9,054,728 ("the '728 Patent") entitled "Data compression systems and methods." The '728 Patent was duly and legally issued by the United States Patent and Trademark Office on June 9, 2015. A true and correct copy of the '728 Patent is included as Exhibit B.

### Oracle SecureFiles

24. On information and belief, Oracle has made, used, offered for sale, sold and/or imported into the United States Oracle products that infringe the '728 patent, and

continues to do so. By way of illustrative example, these infringing products include, without limitation, Oracle's compression products and services, such as, *e.g.*, Oracle Database 11g Release 2 and Oracle Database 12c, each of which includes Oracle's SecureFiles technology, and all versions and variations thereof since the issuance of the '728 patent ("Accused Instrumentality").

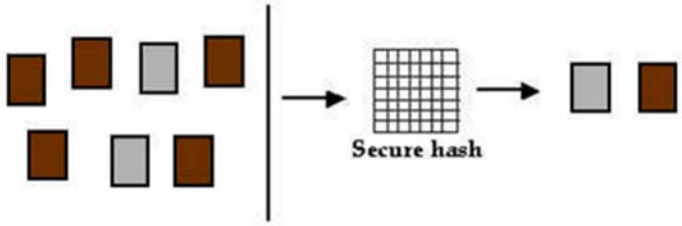
25. On information and belief, Oracle has directly infringed and continues to infringe the '728 patent, for example, through its own use and testing of the Accused Instrumentality, which constitute systems for compressing data claimed by Claim 1 of the '728 patent, comprising a processor; one or more content dependent data compression encoders; and a single data compression encoder; wherein the processor is configured: to analyze data within a data block to identify one or more parameters or attributes of the data wherein the analyzing of the data within the data block to identify the one or more parameters or attributes of the data excludes analyzing based solely on a descriptor that is indicative of the one or more parameters or attributes of the data within the data block; to perform content dependent data compression with the one or more content dependent data compression encoders if the one or more parameters or attributes of the data are identified; and to perform data compression with the single data compression encoder, if the one or more parameters or attributes of the data are not identified. Upon information and belief, Oracle uses the Accused Instrumentality, an infringing system, for its own internal non-testing business purposes, while testing the Accused Instrumentality, and while providing technical support and repair services for the Accused Instrumentality to Oracle's customers.

26. The Accused Instrumentality is a system for compressing data, comprising a processor and one or more content dependent data compression encoders. *See, e.g.*, http://www.oracle.com/technetwork/database/sfe-092218.html:

ORACLE DATABASE 12c SecureFiles Advanced Features

The advanced features are available only for SecureFiles and do not apply to older LOBs or BasicFiles. The new features in SecureFiles - Deduplication, Compression and Encryption - can be setup independently or as a combination of one or more features. If all three features are turned on, Oracle will perform deduplication first and then compression followed by encryption.

Deduplication




Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. In addition to storing only one copy, SecureFiles maintains 'references' for other duplicates. Deduplication is completely transparent to applications. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment. The lob_storage_clause allows for specifying deduplication at a partition level. Duplicate detection does not span across partitions or subpartitions for partitioned SecureFile columns.

More details on Deduplication can be found in the SecureFiles and Large Object Developer's Guide.

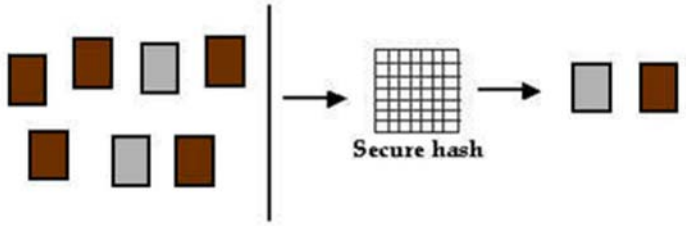
SecureFile Deduplication is part of the Advanced Compression Option.

27. The Accused Instrumentality uses a single data compression encoder. *See*, *e.g.*, http://www.oracle.com/technetwork/database/sfe-092218.html:

ORACLE DATABASE 12c SecureFiles Advanced Features

The advanced features are available only for SecureFiles and do not apply to older LOBs or BasicFiles. The new features in SecureFiles - Deduplication, Compression and Encryption - can be setup independently or as a combination of one or more features. If all three features are turned on, Oracle will perform deduplication first and then compression followed by encryption.

Deduplication



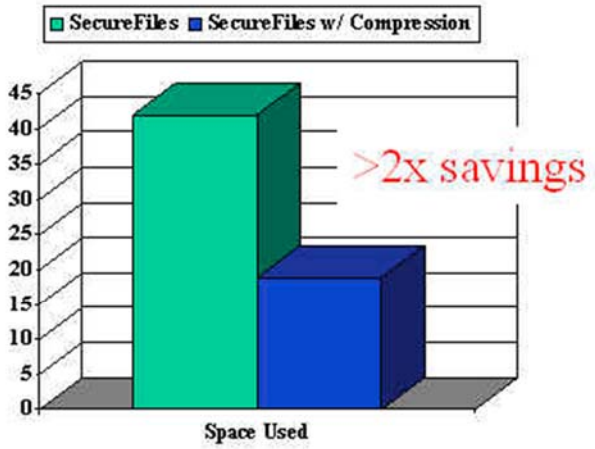

Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. In addition to storing only one copy, SecureFiles maintains 'references' for other duplicates. Deduplication is completely transparent to applications. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment. The lob_storage_clause allows for specifying deduplication at a partition level. Duplicate detection does not span across partitions or subpartitions for partitioned SecureFile columns.

More details on Deduplication can be found in the SecureFiles and Large Object Developer's Guide.

SecureFile Deduplication is part of the Advanced Compression Option.

Compression

SecureFile data is compressed using industry standard compression algorithms. Compresion not only results in significant savings in storage but also improved performance by reducing IO, buffer cache requirements, redo generation and encryption overhead. If the compression does not yield any savings or if the data is already compressed, SecureFiles will automatically turn off compression for such columns. Compression is performed on the server-side and allows for random reads and writes to SecureFile data. SecureFile provides for varying degrees of compression: MEDIUM (default) and HIGH, which represent a tradeoff between storage savings and latency.




Compression tested on the Calgary data set consisting of doc, pdf and text documents. Actual compression might vary depending on the data loaded into SecureFiles. More details on Compression can be found in the SecureFiles and Large Object Developer's Guide

SecureFile Compression is part of the Advanced Compression Option.

28. The Accused Instrumentality analyzes data within a data block to identify one or more parameter of the data, in this case, whether the data has been recognized as having been seen by the system before and where the analysis does not rely only on the descriptor. *See, e.g.,* http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf at 7 ("Deduplication: This feature eliminates multiple, redundant copies of SecureFiles data and is completely transparent to applications. Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment.").

29. The Accused Instrumentality performs content dependent data

compression with the one or more content dependent data compression encoders if the one or more parameters or attributes of the data are identified. *See, e.g.*, http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf at 7 ("Deduplication: This feature eliminates multiple, redundant copies of SecureFiles data and is completely transparent to applications. Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment.").

30. The Accused Instrumentality performs data compression with the single data compression encoder, if the one or more parameters or attributes of the data are not identified. *See, e.g.*, http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf at 8 ("Compression: Oracle automatically detects if SecureFiles data is compressible and will compress using industry standard compression algorithms. If the compression does not yield any savings or if the data is already compressed, SecureFiles will automatically turn off compression for such LOBs. Compression not only results in significant savings in storage but also improved performance by reducing I/O, buffer cache requirements, redo generation and encryption overhead.").

31. On information and belief, Oracle also directly infringes and continues to infringe other claims of the '728 patent, for similar reasons as explained above with respect to Claim 1 of the '728 patent.

32. On information and belief, all of the Accused Instrumentalities operate in substantially the same way. In particular, on information and belief, the SecureFiles compression technology used in Oracle Database 12c is similar to the SecureFiles compression technology used in Oracle Database 11g: "SecureFiles is available in Oracle Database 11g and 12c on all supported database platforms. … It is now the default LOB

storage in Oracle Database 12c." [3]

33. On information and belief, use of the Accused Instrumentality in its ordinary and customary fashion results in infringement of the systems claimed by the '728 patent.

34. On information and belief, Oracle has had knowledge of the '728 patent since at least the filing of the original Complaint or shortly thereafter, and on information and belief, Oracle knew of the '728 patent and knew of its infringement, including by way of this lawsuit.

35. Upon information and belief, Oracle's affirmative acts of making, using, and selling the Accused Instrumentalities, and providing implementation services and technical support to users of the Accused Instrumentalities, have induced and continue to induce users of the Accused Instrumentalities to use them in their normal and customary way to infringe the '728 patent by making or using a system for compressing data comprising a processor; one or more content dependent data compression encoders; and a single data compression encoder; wherein the processor is configured: to analyze data within a data block to identify one or more parameters or attributes of the data wherein the analyzing of the data within the data block to identify the one or more parameters or attributes of the data excludes analyzing based solely on a descriptor that is indicative of the one or more parameters or attributes of the data within the data block; to perform content dependent data compression with the one or more content dependent data compression encoders if the one or more parameters or attributes of the data are identified; and to perform data compression with the single data compression encoder, if the one or more parameters or attributes of the data are not identified. For example, Oracle instructs users of SecureFiles that, "The new features in SecureFiles - Deduplication, Compression and Encryption - can be setup independently or as a

[3] http://www.oracle.com/technetwork/database/sf-faq-082597.html

combination of one or more features. If all three features are turned on, Oracle will perform deduplication first and then compression followed by encryption. Deduplication: This feature eliminates multiple, redundant copies of SecureFiles data and is completely transparent to applications. Oracle automatically detects multiple, identical SecureFiles data and stores only one copy, thereby saving storage space. Deduplication not only simplifies storage management but also results in significantly better performance, especially for copy operations. Duplicate detection happens within a LOB segment. … Compression: Oracle automatically detects if SecureFile data is compressible and will compress using industry standard compression algorithms. If the compression does not yield any savings or if the data is already compressed, SecureFiles will automatically turn off compression for such LOBs. Compression not only results in significant savings in storage but also improved performance by reducing I/O, buffer cache requirements, redo generation and encryption overhead." [4] Oracle specifically intended and was aware that these normal and customary activities would infringe the '728 patent. Oracle performed the acts that constitute induced infringement, and would induce actual infringement, with the knowledge of the '728 patent and with the knowledge, or willful blindness to the probability, that the induced acts would constitute infringement. On information and belief, Oracle engaged in such inducement to promote the sales of the Accused Instrumentalities. Accordingly, Oracle has induced and continues to induce users of the accused products to use the Accused Instrumentalities in their ordinary and customary way to infringe the '728 patent, knowing that such use constitutes infringement of the '728 patent.

36. By making, using, offering for sale, selling and/or importing into the United States the Accused Instrumentalities, and touting the benefits of using the

[4] http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf at 7-8.

Accused Instrumentalities' compression features, Oracle has injured Realtime and is liable to Realtime for infringement of the '728 patent pursuant to 35 U.S.C. § 271.

37. As a result of Oracle's infringement of the '728 patent, Plaintiff Realtime is entitled to monetary damages in an amount adequate to compensate for Oracle's infringement, but in no event less than a reasonable royalty for the use made of the invention by Oracle, together with interest and costs as fixed by the Court.

## COUNT III

## INFRINGEMENT OF U.S. PATENT NO. 6,597,812

38. Plaintiff realleges and incorporates by reference paragraphs 1-37 above, as if fully set forth herein.

39. Plaintiff Realtime is the owner by assignment of United States Patent No. 6,597,812 ("the '812 patent") entitled "System and method for lossless data compression and decompression." The '812 patent was duly and legally issued by the United States Patent and Trademark Office on July 22, 2003. A true and correct copy of the '812 patent is included as Exhibit C.

**Oracle Database 12c with Oracle Database In-Memory**

40. On information and belief, Oracle has made, used, offered for sale, sold and/or imported into the United States Oracle products that infringe the '728 patent, and continues to do so. By way of illustrative example, these infringing products include, without limitation, Oracle's compression products and services, such as, *e.g.*, Oracle Database 12c, which includes Oracle Database In-Memory technology, and all versions and variations thereof since the issuance of the '812 patent ("Accused Instrumentality").

41. On information and belief, Oracle has directly infringed and continues to infringe the '812 patent, for example, through its own use and testing of the Accused Instrumentality to practice compression methods claimed by Claim 1 of the '812 patent, namely, a method for compressing input data comprising a plurality of data blocks, the

method comprising the steps of: detecting if the input data comprises a run-length sequence of data blocks; outputting an encoded run-length sequence, if a run-length sequence of data blocks is detected; maintaining a dictionary comprising a plurality of code words, wherein each code word in the dictionary is associated with a unique data block string; building a data block string from at least one data block in the input data that is not part of a run-length sequence; searching for a code word in the dictionary having a unique data block string associated therewith that matches the built data block string; and outputting the code word representing the built data block string. Upon information and belief, Oracle uses the Accused Instrumentality to practice infringing methods for its own internal non-testing business purposes, while testing the Accused Instrumentality, and while providing technical support and repair services for the Accused Instrumentality to Oracle's customers.

42. The Accused Instrumentality compresses input data comprising a plurality of data blocks, the method comprising the steps of: detecting if the input data comprises a run-length sequence of data blocks; and outputting an encoded run-length sequence, if a run-length sequence of data blocks is detected. *See, e.g.*, http://www.oracle.com/technetwork/database/in-memory/overview/twp-oracle-database-in-memory-2245633.html at 10 ("By default, data is compressed using the FOR QUERY LOW option, which provides the best performance for queries. This option utilizes common compression techniques such as Dictionary Encoding, Run Length Encoding and Bit-Packing."); http://www.oracle.com/technetwork/server-storage/hardware-solutions/oos-for-secure-oracle-database-2736047.pdf at 13 ("By default, compression is optimized for query performance using dictionary encoding, run length encoding, and bit-packing algorithms.").

43. The Accused Instrumentality maintains a dictionary comprising a plurality of code words, wherein each code word in the dictionary is associated with a unique data block string; building a data block string from at least one data block in the input data

that is not part of a run-length sequence; searching for a code word in the dictionary having a unique data block string associated therewith that matches the built data block string; and outputting the code word representing the built data block string. *See, e.g.*, http://www.oracle.com/technetwork/database/in-memory/overview/twp-oracle-database-in-memory-2245633.html at 10 ("By default, data is compressed using the FOR QUERY LOW option, which provides the best performance for queries. This option utilizes common compression techniques such as Dictionary Encoding, Run Length Encoding and Bit-Packing."); http://www.oracle.com/technetwork/server-storage/hardware-solutions/oos-for-secure-oracle-database-2736047.pdf at 13 ("By default, compression is optimized for query performance using dictionary encoding, run length encoding, and bit-packing algorithms."); http://www.bluefinsolutions.com/blogs/john-appleby/october-2014/oracle-database-in-memory-faq ("An initial version of the column store is built when the data is first accessed. This is comprised of multiple In-Memory Compression Units or IMCUs, which contain unsorted data in the same order as the row store. The IMCUs are dictionary encoded and use various compression strategies.").

44. On information and belief, Oracle also directly infringes and continues to infringe other claims of the '812 patent, for similar reasons as explained above with respect to Claim 1 of the '812 patent.

45. On information and belief, all of the Accused Instrumentalities operate in substantially the same way.

46. On information and belief, use of the Accused Instrumentality in its ordinary and customary fashion results in infringement of the methods claimed by the '812 patent.

47. On information and belief, Oracle has had knowledge of the '812 patent since at least the filing of the original Complaint or shortly thereafter, and on information and belief, Oracle knew of the '812 patent and knew of its infringement, including by way of this lawsuit.

48. Upon information and belief, Oracle's affirmative acts of making, using, and selling the Accused Instrumentalities, and providing implementation services and technical support to users of the Accused Instrumentalities, have induced and continue to induce users of the Accused Instrumentalities to use them in their normal and customary way to infringe the '812 patent by practicing compression methods claimed by Claim 1 of the '812 patent, namely, a method for compressing input data comprising a plurality of data blocks, the method comprising the steps of: detecting if the input data comprises a run-length sequence of data blocks; outputting an encoded run-length sequence, if a run-length sequence of data blocks is detected; maintaining a dictionary comprising a plurality of code words, wherein each code word in the dictionary is associated with a unique data block string; building a data block string from at least one data block in the input data that is not part of a run-length sequence; searching for a code word in the dictionary having a unique data block string associated therewith that matches the built data block string; and outputting the code word representing the built data block string. For example, Oracle instructs users of Oracle Database In-Memory technology that is operates by default using a combination of dictionary compression, run-length encoding, and bit packing. *See, e.g.*, http://www.oracle.com/technetwork/database/in-memory/overview/twp-oracle-database-in-memory-2245633.html at 10 ("By default, data is compressed using the FOR QUERY LOW option, which provides the best performance for queries. This option utilizes common compression techniques such as Dictionary Encoding, Run Length Encoding and Bit-Packing."); http://www.oracle.com/technetwork/server-storage/hardware-solutions/oos-for-secure-oracle-database-2736047.pdf at 13 ("By default, compression is optimized for query performance using dictionary encoding, run length encoding, and bit-packing algorithms."). For similar reasons, Oracle also induces its customers to use the Accused Instrumentalities to infringe other claims of the '812 patent. Oracle specifically intended and was aware that these normal and customary activities would infringe the '812 patent.

Oracle performed the acts that constitute induced infringement, and would induce actual infringement, with the knowledge of the '812 patent and with the knowledge, or willful blindness to the probability, that the induced acts would constitute infringement. On information and belief, Oracle engaged in such inducement to promote the sales of the Accused Instrumentalities. Accordingly, Oracle has induced and continue to induce users of the accused products to use the accused products in their ordinary and customary way to infringe the '812 patent, knowing that such use constitutes infringement of the '812 patent.

49. By making, using, offering for sale, selling and/or importing into the United States the Accused Instrumentalities, and touting the benefits of using the Accused Instrumentalities' compression features, Oracle has injured Realtime and is liable to Realtime for infringement of the '812 patent pursuant to 35 U.S.C. § 271.

50. As a result Oracle's infringement of the '812 patent, Plaintiff Realtime is entitled to monetary damages in an amount adequate to compensate for Oracle's infringement, but in no event less than a reasonable royalty for the use made of the invention by Oracle, together with interest and costs as fixed by the Court.

## COUNT IV
## INFRINGEMENT OF U.S. PATENT NO. 7,358,867

51. Plaintiff Realtime realleges and incorporates by reference paragraphs 1-50 above, as if fully set forth herein.

52. Plaintiff Realtime is the owner by assignment of United States Patent No. 7,358,867 ("the '867 Patent") entitled "Content independent data compression method and system." The '867 Patent was duly and legally issued by the United States Patent and Trademark Office on April 15, 2008. A true and correct copy of the '867 Patent is included as Exhibit D.

**Oracle Database 11g / 12c**

53. On information and belief, Oracle has made, used, offered for sale, sold and/or imported into the United States Oracle products that infringe the '867 patent, and continues to do so. By way of illustrative example, these infringing products include, without limitation, Oracle's compression products and services, such as, *e.g.*, Oracle Database 11g Release 2 and Oracle Database 12c, each of which includes Oracle's Hybrid Columnar Compression ("HCC") technology, and all versions and variations thereof since the issuance of the '867 patent ("Accused Instrumentality").

54. On information and belief, Oracle has directly infringed and continues to infringe the '867 patent, for example, through its own use and testing of the Accused Instrumentality to practice compression methods claimed by Claim 16 of the '867 patent, namely, a method comprising: receiving a plurality of data blocks; determining whether or not to compress each one of said plurality of data blocks with a particular one or more of several encoders; if said determination is to compress with said particular one or more of said several encoders for a particular one of said plurality of data blocks; compressing said particular one of said plurality of data blocks with said particular one or more of said several encoders to provide a compressed data block; providing a data compression type descriptor representative of said particular one or more of said several encoders; outputting said data compression type descriptor and said compressed data block; if said determination is to not compress said particular one of said plurality of data blocks; providing a null data compression type descriptor representative of said determination not to compress; and outputting said null data compression type descriptor and said particular one of said plurality of data blocks. Upon information and belief, Oracle uses the Accused Instrumentality, an infringing system, for its own internal non-testing business purposes, while testing the Accused Instrumentality, and while providing technical support and repair services for the Accused Instrumentality to Oracle's customers.

55. The Accused Instrumentality practices a method comprising: receiving a plurality of data blocks; determining whether or not to compress each one of said

plurality of data blocks with a particular one or more of several encoders; if said determination is to compress with said particular one or more of said several encoders for a particular one of said plurality of data blocks; compressing said particular one of said plurality of data blocks with said particular one or more of said several encoders to provide a compressed data block; providing a data compression type descriptor representative of said particular one or more of said several encoders; outputting said data compression type descriptor and said compressed data block. *See, e.g.*, https://docs.oracle.com/cd/B28359_01/server.111/b28314/tdpdw_optimize.htm:

# Optimizing Storage Requirements

You can reduce your storage requirements by compressing data, which is achieved by eliminating duplicate values in a database block. Database objects that can be compressed include tables and materialized views. For partitioned tables, you can choose to compress some or all partitions. Compression attributes can be declared for a tablespace, a table, or a partition of a table. If declared at the tablespace level, then all tables created in that tablespace are compressed by default. You can alter the compression attribute for a table (or a partition or tablespace), and the change only applies only to new data going into that table. As a result, a single table or partition may contain some compressed blocks and some regular blocks. This guarantees that data size will not increase as a result of compression; in cases where compression could increase the size of a block, it is not applied to that block.

## Using Data Compression to Improve Storage

You can compress several partitions or a complete partitioned heap-organized table. You do this either by defining a complete partitioned table as being compressed, or by defining it on a per-partition level. Partitions without a specific declaration inherit the attribute from the table definition or, if nothing is specified on table level, from the tablespace definition.

The decision about whether or not a partition should be compressed or stay uncompressed adheres to the same rules as a nonpartitioned table. However, due to the ability of range and composite partitioning to separate data logically into distinct partitions, such a partitioned table is an ideal candidate for compressing parts of the data (partitions) that are mainly read-only. It is, for example, beneficial in all rolling window operations as a kind of intermediate stage before aging out old data. With data compression, you can keep more old data online, minimizing the burden of additional storage consumption.

https://community.oracle.com/thread/2506070?tstart=0 ("There is overhead associated with the compression because the metadata that is needed to translate any compressed data back into its original state is stored in the block along with the compressed data.").

56. If said determination is to not compress said particular one of said plurality of data blocks, the Accused Instrumentality provides a null data compression type descriptor representative of said determination not to compress; and outputs said null data compression type descriptor and said particular one of said plurality of data blocks. *See*, *e.g.*, https://docs.oracle.com/cd/B28359_01/server.111/b28314/tdpdw_optimize.htm:

## Optimizing Storage Requirements

You can reduce your storage requirements by compressing data, which is achieved by eliminating duplicate values in a database block. Database objects that can be compressed include tables and materialized views. For partitioned tables, you can choose to compress some or all partitions. Compression attributes can be declared for a tablespace, a table, or a partition of a table. If declared at the tablespace level, then all tables created in that tablespace are compressed by default. You can alter the compression attribute for a table (or a partition or tablespace), and the change only applies only to new data going into that table. As a result, a single table or partition may contain some compressed blocks and some regular blocks. This guarantees that data size will not increase as a result of compression; in cases where compression could increase the size of a block, it is not applied to that block.

## Using Data Compression to Improve Storage

You can compress several partitions or a complete partitioned heap-organized table. You do this either by defining a complete partitioned table as being compressed, or by defining it on a per-partition level. Partitions without a specific declaration inherit the attribute from the table definition or, if nothing is specified on table level, from the tablespace definition.

The decision about whether or not a partition should be compressed or stay uncompressed adheres to the same rules as a nonpartitioned table. However, due to the ability of range and composite partitioning to separate data logically into distinct partitions, such a partitioned table is an ideal candidate for compressing parts of the data (partitions) that are mainly read-only. It is, for example, beneficial in all rolling window operations as a kind of intermediate stage before aging out old data. With data compression, you can keep more old data online, minimizing the burden of additional storage consumption.

57. On information and belief, Oracle also directly infringes and continues to infringe other claims of the ‘867 patent, for similar reasons as explained above with respect to Claim 16 of the ‘867 patent.

58. On information and belief, all of the Accused Instrumentalities perform the claimed methods in substantially the same way. In particular, on information and belief, the compression technology used in Oracle Database 12c is similar to the

compression technology used in Oracle Database 11g. *See, e.g.*, https://docs.oracle.com/database/121/DWHSG/schemas.htm#DWHSG8912 ("In cases where compression could increase the size of a block, it is not applied to that block. … The decision about whether or not a partition should be compressed is based on the same rules as a nonpartitioned table.") (from Oracle Database Online Documentation 12c Release 1 (12.1)).

59. On information and belief, use of the Accused Instrumentality in its ordinary and customary fashion results in infringement of the methods claimed by the '867 patent.

60. On information and belief, Oracle has had knowledge of the '867 patent since at least the filing of this Complaint or shortly thereafter, and on information and belief, Oracle knew of the '867 patent and knew of its infringement, including by way of this lawsuit.

61. Upon information and belief, Oracle's affirmative acts of making, using, and selling the Accused Instrumentalities, and providing implementation services and technical support to users of the Accused Instrumentalities, have induced and continue to induce users of the Accused Instrumentalities to use them in their normal and customary way to infringe the '867 patent by practicing a method comprising: receiving a plurality of data blocks; determining whether or not to compress each one of said plurality of data blocks with a particular one or more of several encoders; if said determination is to compress with said particular one or more of said several encoders for a particular one of said plurality of data blocks; compressing said particular one of said plurality of data blocks with said particular one or more of said several encoders to provide a compressed data block; providing a data compression type descriptor representative of said particular one or more of said several encoders; outputting said data compression type descriptor and said compressed data block; if said determination is to not compress said particular one of said plurality of data blocks; providing a null data compression type descriptor

representative of said determination not to compress; and outputting said null data compression type descriptor and said particular one of said plurality of data blocks. For example, Oracle instructs its customers about the benefits of using the compression features of the Accused Instrumentality. *See, e.g.*, https://docs.oracle.com/cd/B28359_01/server.111/b28314/tdpdw_optimize.htm ("You can reduce your storage requirements by compressing data, which is achieved by eliminating duplicate values in a database block. Database objects that can be compressed include tables and materialized views. For partitioned tables, you can choose to compress some or all partitions. … in cases where compression could increase the size of a block, it is not applied to that block. … The decision about whether or not a partition should be compressed or stay uncompressed adheres to the same rules as a nonpartitioned table."); https://community.oracle.com/thread/2506070?tstart=0 ("There is overhead associated with the compression because the metadata that is needed to translate any compressed data back into its original state is stored in the block along with the compressed data."). Oracle specifically intended and was aware that the normal and customary use of compression in the Accused Instrumentalities would infringe the '867 patent. Oracle performed the acts that constitute induced infringement, and would induce actual infringement, with the knowledge of the '867 patent and with the knowledge, or willful blindness to the probability, that the induced acts would constitute infringement. On information and belief, Oracle engaged in such inducement to promote the sales of the Accused Instrumentalities. Accordingly, Oracle has induced and continues to induce users of the accused products to use the accused products in their ordinary and customary way to infringe the '867 patent, knowing that such use constitutes infringement of the '867 patent.

62. By making, using, offering for sale, selling and/or importing into the United States the Accused Instrumentalities, and touting the benefits of using the Accused Instrumentalities' compression features, Oracle has injured Realtime and is

liable to Realtime for infringement of the ‘867 patent pursuant to 35 U.S.C. § 271.

63. As a result of Oracle’s infringement of the ‘867 patent, Plaintiff Realtime is entitled to monetary damages in an amount adequate to compensate for Oracle’s infringement, but in no event less than a reasonable royalty for the use made of the invention by Oracle, together with interest and costs as fixed by the Court.

**Oracle SecureFiles**

64. On information and belief, Oracle has made, used, offered for sale, sold and/or imported into the United States Oracle products that infringe the ‘867 patent, and continues to do so. By way of illustrative example, these infringing products include, without limitation, Oracle’s compression products and services, such as, *e.g.*, Oracle Database 11g Release 2 and Oracle Database 12c, each of which includes Oracle’s SecureFiles technology, and all versions and variations thereof since the issuance of the ‘867 patent (“Accused Instrumentality”).

65. On information and belief, Oracle has directly infringed and continues to infringe the ‘867 patent, for example, through its own use and testing of the Accused Instrumentality to practice compression methods claimed by Claim 16 of the ‘867 patent, namely, a method comprising: receiving a plurality of data blocks; determining whether or not to compress each one of said plurality of data blocks with a particular one or more of several encoders; if said determination is to compress with said particular one or more of said several encoders for a particular one of said plurality of data blocks; compressing said particular one of said plurality of data blocks with said particular one or more of said several encoders to provide a compressed data block; providing a data compression type descriptor representative of said particular one or more of said several encoders; outputting said data compression type descriptor and said compressed data block; if said determination is to not compress said particular one of said plurality of data blocks; providing a null data compression type descriptor representative of said determination not to compress; and outputting said null data compression type descriptor and said particular

one of said plurality of data blocks. Upon information and belief, Oracle uses the Accused Instrumentality, an infringing system, for its own internal non-testing business purposes, while testing the Accused Instrumentality, and while providing technical support and repair services for the Accused Instrumentality to Oracle's customers.

66. The Accused Instrumentality practices a method comprising: receiving a plurality of data blocks; determining whether or not to compress each one of said plurality of data blocks with a particular one or more of several encoders; if said determination is to compress with said particular one or more of said several encoders for a particular one of said plurality of data blocks; compressing said particular one of said plurality of data blocks with said particular one or more of said several encoders to provide a compressed data block; providing a data compression type descriptor representative of said particular one or more of said several encoders; outputting said data compression type descriptor and said compressed data block: "Another feature of SecureFiles is compression. … Compression takes up CPU cycles so depending on how much data is compressible, it may not be worthy of compression. For instance, if you have a lot of JPEG pictures they are compressed already, so further compression will not save any space. On the other hand, if you have an XML document as a CLOB, then compression may produce substantial reduction. SecureFiles compression automatically detects if the data is compressible and only spends CPU cycles if compression yields gains. … In Oracle Database 11g Release 2, there is a third compression option in addition to HIGH and MEDIUM: LOW. As the name suggests, it compresses less but also consumes a lot less CPU and completes faster. This approach uses a block-based lossless compression similar to the fast Lempel–Ziv–Oberhumer (LZO) algorithm."[5]

67. If said determination is to not compress said particular one of said plurality of data blocks, the Accused Instrumentality provides a null data compression

[5] http://www.oracle.com/technetwork/articles/sql/11g-securefiles-084075.html

type descriptor representative of said determination not to compress; and outputs said null data compression type descriptor and said particular one of said plurality of data blocks: "Another feature of SecureFiles is compression. … Compression takes up CPU cycles so depending on how much data is compressible, it may not be worthy of compression. For instance, if you have a lot of JPEG pictures they are compressed already, so further compression will not save any space. On the other hand, if you have an XML document as a CLOB, then compression may produce substantial reduction. SecureFiles compression automatically detects if the data is compressible and only spends CPU cycles if compression yields gains." [6] "Compression: Oracle automatically detects if SecureFile data is compressible and will compress using industry standard compression algorithms. If the compression does not yield any savings or if the data is already compressed, SecureFiles will automatically turn off compression for such LOBs." [7]

68. On information and belief, Oracle also directly infringes and continues to infringe other claims of the '867 patent, for similar reasons as explained above with respect to Claim 16 of the '867 patent.

69. On information and belief, all of the Accused Instrumentalities perform the claimed methods in substantially the same way. In particular, on information and belief, the SecureFiles compression technology used in Oracle Database 12c is similar to the SecureFiles compression technology used in Oracle Database 11g: "SecureFiles is available in Oracle Database 11g and 12c on all supported database platforms. … It is now the default LOB storage in Oracle Database 12c." [8]

70. On information and belief, use of the Accused Instrumentality in its ordinary and customary fashion results in infringement of the methods claimed by the

[6] http://www.oracle.com/technetwork/articles/sql/11g-securefiles-084075.html

[7] http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf at 8.

[8] http://www.oracle.com/technetwork/database/sf-faq-082597.html

‘867 patent.

71. On information and belief, Oracle has had knowledge of the ‘867 patent since at least the filing of this Complaint or shortly thereafter, and on information and belief, Oracle knew of the ‘867 patent and knew of its infringement, including by way of this lawsuit.

72. Upon information and belief, Oracle’s affirmative acts of making, using, and selling the Accused Instrumentalities, and providing implementation services and technical support to users of the Accused Instrumentalities, have induced and continue to induce users of the Accused Instrumentalities to use them in their normal and customary way to infringe the ‘867 patent by practicing a method comprising: receiving a plurality of data blocks; determining whether or not to compress each one of said plurality of data blocks with a particular one or more of several encoders; if said determination is to compress with said particular one or more of said several encoders for a particular one of said plurality of data blocks; compressing said particular one of said plurality of data blocks with said particular one or more of said several encoders to provide a compressed data block; providing a data compression type descriptor representative of said particular one or more of said several encoders; outputting said data compression type descriptor and said compressed data block; if said determination is to not compress said particular one of said plurality of data blocks; providing a null data compression type descriptor representative of said determination not to compress; and outputting said null data compression type descriptor and said particular one of said plurality of data blocks. For example, Oracle instructs its customers that, “Another feature of SecureFiles is compression. … Compression takes up CPU cycles so depending on how much data is compressible, it may not be worthy of compression. For instance, if you have a lot of JPEG pictures they are compressed already, so further compression will not save any space. On the other hand, if you have an XML document as a CLOB, then compression may produce substantial reduction. SecureFiles compression automatically detects if the

data is compressible and only spends CPU cycles if compression yields gains. … In Oracle Database 11g Release 2, there is a third compression option in addition to HIGH and MEDIUM: LOW. As the name suggests, it compresses less but also consumes a lot less CPU and completes faster. This approach uses a block-based lossless compression similar to the fast Lempel–Ziv–Oberhumer (LZO) algorithm." [9] Oracle specifically intended and was aware that the normal and customary use of SecureFiles compression in the Accused Instrumentalities would infringe the '867 patent. Oracle performed the acts that constitute induced infringement, and would induce actual infringement, with the knowledge of the '867 patent and with the knowledge, or willful blindness to the probability, that the induced acts would constitute infringement. On information and belief, Oracle engaged in such inducement to promote the sales of the Accused Instrumentalities. Accordingly, Oracle has induced and continues to induce users of the accused products to use the accused products in their ordinary and customary way to infringe the '867 patent, knowing that such use constitutes infringement of the '867 patent.

73. By making, using, offering for sale, selling and/or importing into the United States the Accused Instrumentalities, and touting the benefits of using the Accused Instrumentalities' compression features, Oracle has injured Realtime and is liable to Realtime for infringement of the '867 patent pursuant to 35 U.S.C. § 271.

74. As a result of Oracle's infringement of the '867 patent, Plaintiff Realtime is entitled to monetary damages in an amount adequate to compensate for Oracle's infringement, but in no event less than a reasonable royalty for the use made of the invention by Oracle, together with interest and costs as fixed by the Court.

## COUNT V
## INFRINGEMENT OF U.S. PATENT NO. 7,395,345

75. Plaintiff Realtime realleges and incorporates by reference paragraphs 1-74

[9] http://www.oracle.com/technetwork/articles/sql/11g-securefiles-084075.html

above, as if fully set forth herein.

76. Plaintiff Realtime is the owner by assignment of United States Patent No. 7,395,345 ("the '345 Patent") entitled "System and methods for accelerated data storage and retrieval." The '345 Patent was duly and legally issued by the United States Patent and Trademark Office on July 1, 2008. A true and correct copy of the '345 Patent is included as Exhibit E.

**Oracle SecureFiles**

77. On information and belief, Oracle has made, used, offered for sale, sold and/or imported into the United States Oracle products that infringe the '345 patent, and continues to do so. By way of illustrative example, these infringing products include, without limitation, Oracle's compression products and services, such as, *e.g.*, Oracle Database 11g Release 2 and Oracle Database 12c, each of which includes Oracle's SecureFiles technology, and all versions and variations thereof since the issuance of the '345 patent ("Accused Instrumentality").

78. On information and belief, Oracle has directly infringed and continues to infringe the '345 patent, for example, through its own use and testing of the Accused Instrumentality to practice compression methods claimed by Claim 3 of the '345 patent, namely, a method comprising: receiving data at an input data transmission rate which is greater than a data storage rate of a target storage device; providing an output data block and a data compression type descriptor, wherein: if said data compression type descriptor is indicative of said data not being compressed then said data was not compressed and said output data block is said data; if said data compression type descriptor is indicative of said data being compressed then said data was compressed based on said compression type descriptor at a compression rate that increased the effective data storage rate of the target storage device to provide a compressed data block, wherein said output data block is said compressed data block; and storing said output data block and said data compression type descriptor in said target storage device, wherein said compressing is

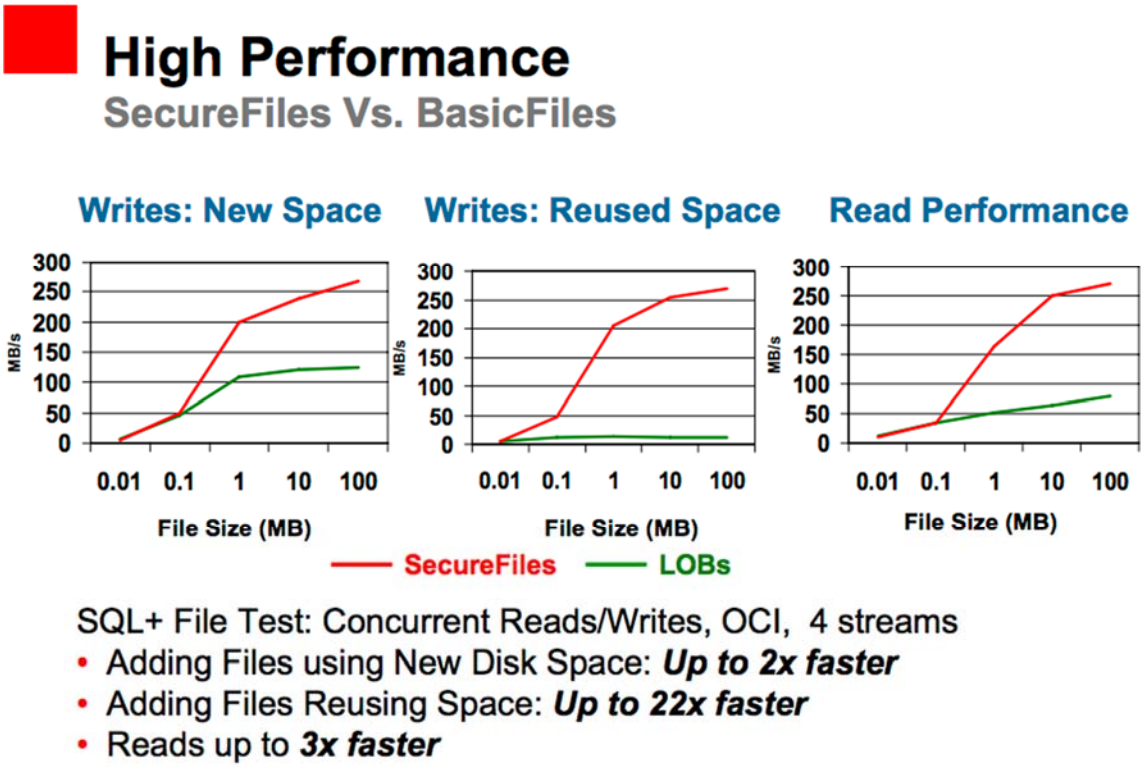
lossless. Upon information and belief, Oracle uses the Accused Instrumentality, an infringing system, for its own internal non-testing business purposes, while testing the Accused Instrumentality, and while providing technical support and repair services for the Accused Instrumentality to Oracle's customers.

79. The Accused Instrumentality practices a method comprising: receiving data at an input data transmission rate which is greater than a data storage rate of a target storage device. *See, e.g.*, http://www.oracle.com/technetwork/database/database-technologies/performance/boost-your-database-performance-10x-130376.pdf at 7 ("SecureFiles is a new database feature designed to break the performance barrier keeping file data out of databases. Similar to LOBs but much faster, and with more capabilities.").

80. The Accused Instrumentality provides an output data block and a data compression type descriptor, wherein: if said data compression type descriptor is indicative of said data not being compressed then said data was not compressed and said output data block is said data. *See, e.g.*, http://www.oracle.com/technetwork/articles/sql/11g-securefiles-084075.html ("Another feature of SecureFiles is compression. … Compression takes up CPU cycles so depending on how much data is compressible, it may not be worthy of compression. For instance, if you have a lot of JPEG pictures they are compressed already, so further compression will not save any space. On the other hand, if you have an XML document as a CLOB, then compression may produce substantial reduction. SecureFiles compression automatically detects if the data is compressible and only spends CPU cycles if compression yields gains."); http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf at 8 ("Compression: Oracle automatically detects if SecureFile data is compressible and will compress using industry standard compression algorithms. If the compression does not yield any savings or if the data is already compressed, SecureFiles

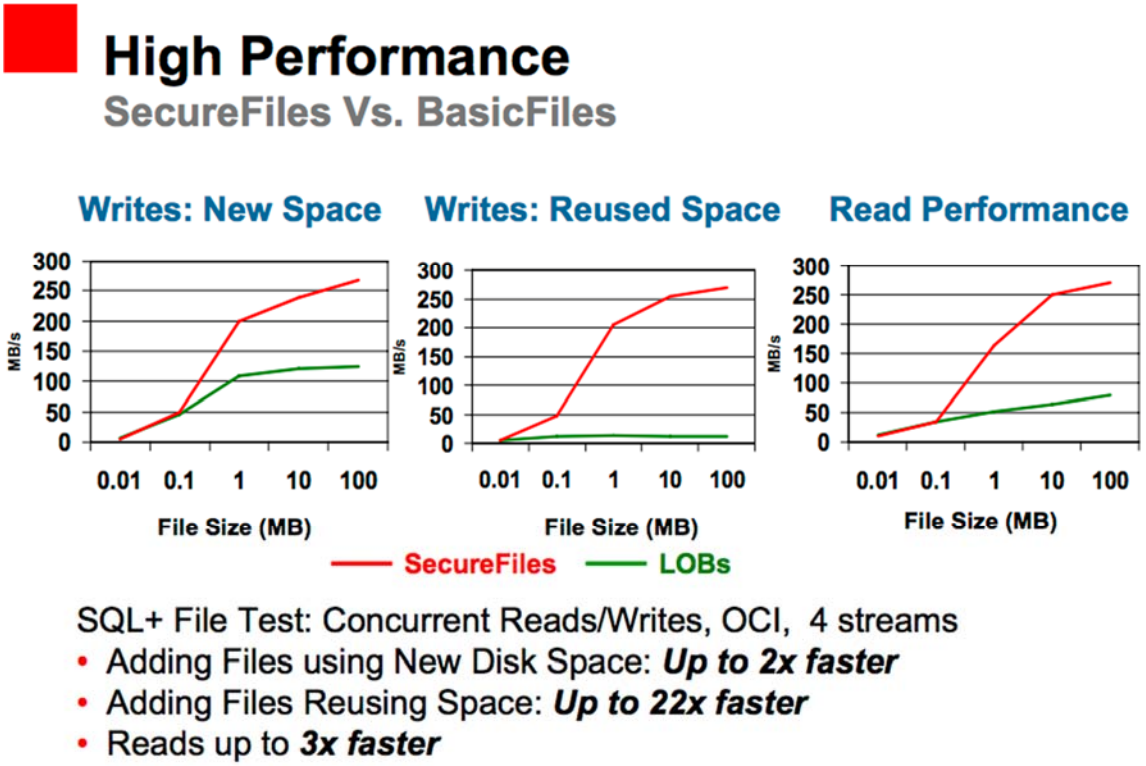
will automatically turn off compression for such LOBs.").

81. If said data compression type descriptor in the Accused Instrumentality is indicative of said data being compressed, then said data was compressed based on said compression type descriptor at a compression rate that increased the effective data storage rate of the target storage device to provide a compressed data block, wherein said output data block is said compressed data block. *See, e.g.*, http://www.oracle.com/technetwork/database/database-technologies/performance/boost-your-database-performance-10x-130376.pdf at 11:




82. The Accused Instrumentality stores said output data block and said data compression type descriptor in said target storage device, wherein said compressing is lossless. *See, e.g.*, http://www.oracle.com/technetwork/articles/sql/11g-securefiles-084075.html ("In Oracle Database 11g Release 2, there is a third compression option in addition to HIGH and MEDIUM: LOW. … This approach uses a block-based lossless compression similar to the fast Lempel–Ziv–Oberhumer (LZO) algorithm.");

http://www.oracle.com/technetwork/database/database-technologies/performance/boost-your-database-performance-10x-130376.pdf at 11:




83. On information and belief, Oracle also directly infringes and continues to infringe other claims of the '345 patent, for similar reasons as explained above with respect to Claim 3 of the '345 patent.

84. On information and belief, all of the Accused Instrumentalities perform the claimed methods in substantially the same way. In particular, on information and belief, the SecureFiles compression technology used in Oracle Database 12c is similar to the SecureFiles compression technology used in Oracle Database 11g: "SecureFiles is available in Oracle Database 11g and 12c on all supported database platforms. … It is now the default LOB storage in Oracle Database 12c." [10]

85. On information and belief, use of the Accused Instrumentality in its ordinary and customary fashion results in infringement of the methods claimed by the

[10] http://www.oracle.com/technetwork/database/sf-faq-082597.html

‘345 patent.

86. On information and belief, Oracle has had knowledge of the ‘345 patent since at least the filing of this Complaint or shortly thereafter, and on information and belief, Oracle knew of the ‘345 patent and knew of its infringement, including by way of this lawsuit.

87. Upon information and belief, Oracle’s affirmative acts of making, using, and selling the Accused Instrumentalities, and providing implementation services and technical support to users of the Accused Instrumentalities, have induced and continue to induce users of the Accused Instrumentalities to use them in their normal and customary way to infringe Claim 3 of the ‘345 patent by practicing a method comprising: receiving data at an input data transmission rate which is greater than a data storage rate of a target storage device; providing an output data block and a data compression type descriptor, wherein: if said data compression type descriptor is indicative of said data not being compressed then said data was not compressed and said output data block is said data; if said data compression type descriptor is indicative of said data being compressed then said data was compressed based on said compression type descriptor at a compression rate that increased the effective data storage rate of the target storage device to provide a compressed data block, wherein said output data block is said compressed data block; and storing said output data block and said data compression type descriptor in said target storage device, wherein said compressing is lossless. For similar reasons, Oracle also induces its customers to use the Accused Instrumentalities to infringe other claims of the ‘345 patent. Oracle specifically intended and was aware that these normal and customary activities would infringe the ‘345 patent. Oracle performed the acts that constitute induced infringement, and would induce actual infringement, with the knowledge of the ‘345 patent and with the knowledge, or willful blindness to the probability, that the induced acts would constitute infringement. On information and belief, Oracle engaged in such inducement to promote the sales of the Accused Instrumentalities. Accordingly,

Oracle has induced and continue to induce users of the accused products to use the accused products in their ordinary and customary way to infringe the '345 patent, knowing that such use constitutes infringement of the '345 patent.

88. By making, using, offering for sale, selling and/or importing into the United States the Accused Instrumentalities, and touting the benefits of using the Accused Instrumentalities' compression features, Oracle has injured Realtime and is liable to Realtime for infringement of the '345 patent pursuant to 35 U.S.C. § 271.

89. As a result Oracle's infringement of the '345 patent, Plaintiff Realtime is entitled to monetary damages in an amount adequate to compensate for Oracle's infringement, but in no event less than a reasonable royalty for the use made of the invention by Oracle, together with interest and costs as fixed by the Court.

**PRAYER FOR RELIEF**

WHEREFORE, Plaintiff Realtime respectfully requests that this Court enter:

a. A judgment in favor of Plaintiff that Oracle has infringed, either literally and/or under the doctrine of equivalents, the '506 patent, the '728 patent, the '812 patent, the '867 patent and the '345 patent;

b. A judgment and order requiring Oracle to pay Plaintiff its damages, costs, expenses, and prejudgment and post-judgment interest for its infringement of the '506 patent, the '728 patent, the '812 patent, the '867 patent and the '345 patent as provided under 35 U.S.C. § 284;

c. A judgment and order requiring Oracle to provide an accounting and to pay supplemental damages to Realtime, including without limitation, prejudgment and post-judgment interest;

d. A judgment and order finding that this is an exceptional case within the meaning of 35 U.S.C. § 285 and awarding to Plaintiff its reasonable attorneys' fees against Oracle; and

e. Any and all other relief as the Court may deem appropriate and just under

the circumstances.

**DEMAND FOR JURY TRIAL**

Plaintiff, under Rule 38 of the Federal Rules of Civil Procedure, requests a trial by jury of any issues so triable by right.

Dated: February 26, 2016

Respectfully submitted,

*/s/ Marc A. Fenster by permission Claire Abernathy Henry*
Marc A. Fenster (CA SBN 181067)
Reza Mirzaie (CA SBN 246953)
Brian D. Ledahl (CA SBN 186579)
Jeffrey Z.Y. Liao (CA SBN 288994)
C. Jay Chung (CA SBN 252794)
**RUSS AUGUST & KABAT**
12424 Wilshire Boulevard, 12th Floor
Los Angeles, CA 90025
(310) 826-7474
mfenster@raklaw.com
rmirzaie@raklaw.com
bledahl@raklaw.com
jliao@raklaw.com
jchung@raklaw.com

T. John Ward, Jr.
Texas State Bar No. 00794818
E-mail: jw@wsfirm.com
Claire Abernathy Henry
Texas State Bar No. 24053063
E-mail: claire@wsfirm.com
**WARD, SMITH & HILL, PLLC**
1127 Judson Road, Ste 220
Longview, Texas 75601
(903) 757-6400 (telephone)
(903) 757-2323 (facsimile)

***Attorneys for Plaintiff***
***Realtime Data LLC d/b/a IXO***