IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
SHERMAN DIVISION

R2 Solutions LLC, Plaintiff, v. American Airlines, Inc., Defendant.	Civil Action No. 4:22-cv-00353-ALM Jury Trial Demanded

PLAINTIFF'S DISCLOSURE OF
ASSERTED CLAIMS AND INFRINGEMENT CONTENTIONS

Pursuant to Patent Rules 3-1 and 3-2 of the Eastern District of Texas, Plaintiff R2 Solutions LLC ("R2" or "Plaintiff") serves its Disclosure of Asserted Claims and Infringement Contentions on Defendant American Airlines, Inc. ("AA" or "Defendant") for infringement of U.S. Patent Nos. 8,190,610 ("the '610 patent"), 8,341,157 ("the '157 patent"), 7,698,329 ("the '329 patent"), and 8,209,317 ("the '317 patent"). The '610 patent, '157 patent, '329 patent, and '317 patent are referred to collectively as the "patents-in-suit."

The following asserted claims and contentions are based upon R2's good faith evaluation of information known to R2 at this time based on publicly-available information and documents. For the avoidance of doubt, the claim elements identified in the attached charts are software limitations or associated with software limitations for the purposes of P.R. 3-1(g) promulgated by this Court (ECF 20 at 4-5), and to the extent that such elements are performed by hardware, firmware, software, and/or any combination thereof, the particular details are in AA's possession, custody, and control. R2 reserves its right to supplement and/or amend these contentions in view

of information and/or documents that may be obtained during discovery, further investigation, the Court's claim construction ruling, applicable case law and authorities, and/or any other reasons permitted under the Court's Scheduling Order, the Local Rules, and the Federal Rules of Civil Procedure.

Additionally, R2 incorporates by reference the allegations made in its live complaint in this case, including all identification of asserted claims, accused instrumentalities, and factual allegations related thereto.

I. P. R. 3-1: Disclosure of Asserted Claims and Infringement Contentions

(a) Each claim of each patent in suit that is allegedly infringed by each opposing party

AA directly and/or indirectly infringes the following claims (collectively, the "asserted claims"), as indicated:

1. Directly and indirectly infringes Claims 1–5 and 17–21 of the '610 patent;
2. Directly and indirectly infringes Claims 1–2, 6, 11–14, 16, and 22 of the '157 patent;
3. Directly and indirectly infringes Claims 1, 4–5, 8, and 11–12 of the '329 patent; and
4. Directly and indirectly infringes Claims 1–5 and 8–14 of the '317 patent.

R2 reserves the right to augment or supplement its contentions to identify additional claims infringed by AA after discovery, in response to AA's production pursuant to P.R. 3-4, or as permitted under the Court's Scheduling Order, the Local Rules, and the Federal Rules of Civil Procedure.

(b) Separately for each asserted claim, each accused apparatus, product, device, process, method, act, or other instrumentality ("Accused Instrumentality") of each opposing party of which the party is aware. This identification shall be as specific as possible. Each product, device, and apparatus must be identified by name or model number, if known. Each method or process must be identified by

American Airlines/Hilton, Ex. 1018, p.2
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

name, if known, or by any product, device, or apparatus which, when used, allegedly results in the practice of the claimed method or process

As to the ’610 patent, the Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/or software), and/or other instrumentalities that process big data and/or data analytics. The “Accused Instrumentalities” include, without limitation, all instrumentalities associated with mapping and/or reducing functionalities, merging and/or joining data having different schemas, and any other similar functionality and all instrumentalities related to how data is received, input, stored, organized, partitioned, accessed, analyzed, delivered, processed, and transformed into outputs. This includes the data analytics systems utilizing, at least in part, Apache Hadoop, Hive, Spark, Teradata Vantage, and/or similar functionality. The Accused Instrumentalities that perform and/or embody the functionalities and features discussed and particularly described and claimed in the ’610 patent and Exhibit A hereto are referred to as the “Accused AA Data Analytics Instrumentalities.”

As to the ’157 patent, the ’329 patent, and the ’317 patent, the Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA’s web platform(s) and mobile application(s). “Accused Instrumentalities” include, without limitation, all instrumentalities associated with receiving, handling, processing, responding to, or otherwise interacting with queries (including user queries and back-end queries) and all instrumentalities related to the location, identification, delivery, display, rendering, ranking, and communication of information relating to such queries. The “Accused Instrumentalities” include, without limitation, the front-end(s) and back-end(s) of AA’s online travel agency and search engine services for hotels, vacation rentals, flights, and airport transfer incorporated into the AA web platform and/or AA mobile application. The Accused Instrumentalities that perform and/or embody the

American Airlines/Hilton, Ex. 1018, p.3
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

functionalities and features discussed and particularly described and claimed in the ’157 patent, the ’329 patent, the ’317 patent, and Exhibits B, C, and D hereto are referred to as the “Accused AA Search Instrumentalities.” The Accused AA Search Instrumentalities and Accused AA Data Analytics Instrumentalities are referred to collectively as the “Accused Instrumentalities.”

The Accused Instrumentalities infringe the asserted claims as set forth in Exhibits A, B, C, and D hereto. R2 reserves the right, in response to discovery or as otherwise permitted, to supplement the identification of Accused Instrumentalities to include additional instrumentalities made, used, sold, offered for sale, or imported by AA. By making, using, testing, offering for sale, selling, and/or importing any of the Accused Instrumentalities identified above, AA directly infringes the asserted claims.

As to the asserted claims of the ’610 patent, AA indirectly infringes the ’610 asserted claims in violation of 35 U.S.C. § 271(b) because AA induces direct infringement by its vendors, customers, end users and/or others who make, use, and/or sell the Accused Instrumentalities. AA has been on notice of the ’610 patent as early as October 20, 2021, when R2 explicitly informed AA of resolution of a lawsuit which involved the ’610 patent. Since receiving notice (and thus having knowledge of the claims of the ’610 patent), AA has specifically intended, and continues to specifically intend, for persons who make, use, and/or sell the Accused Instrumentalities, including AA’s vendors, customers, end users, and others, to make, use, and/or sell such Accused Instrumentalities in a manner that infringes the asserted claims of the ’610 patent. AA encourages, directs, and instructs vendors, customers, end users and/or others in the infringing use and operation of the Accused Instrumentalities, via advertisement, technical material, instructional material, instructional videos, technical support, or otherwise.

American Airlines/Hilton, Ex. 1018, p.4
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

As to the asserted claims of the ’157 patent, AA indirectly infringes the ’157 asserted claims in violation of 35 U.S.C. § 271(b) because AA induces direct infringement by its vendors, customers, end users and/or others who make, use, and/or sell the Accused Instrumentalities. AA has been on notice of the ’157 patent as early as October 20, 2021, when R2 explicitly informed AA of resolution of a lawsuit which involved the ’157 patent. Since receiving notice (and thus having knowledge of the claims of the ’157 patent), AA has specifically intended, and continues to specifically intend, for persons who make, use, and/or sell the Accused Instrumentalities, including AA’s vendors, customers, end users and/or others, to make, use, and/or sell such Accused Instrumentalities in a manner that infringes the asserted claims of the ’157 patent. AA encourages, directs, and instructs vendors, customers, end users and/or others in the infringing use and operation of the Accused Instrumentalities, via advertisement, technical material, instructional material, instructional videos, technical support, or otherwise.

As to the asserted claims of the ’329 patent, AA indirectly infringes the ’329 asserted claims in violation of 35 U.S.C. § 271(b) because AA induces direct infringement by its vendors, customers, end users and/or others who make, use, and/or sell the Accused Instrumentalities. AA has been on notice of the ’329 patent as early as November 23, 2021, when R2 explicitly informed AA of resolution of a lawsuit which involved the ’329 patent. Since receiving notice (and thus having knowledge of the claims of the ’329 patent), AA has specifically intended, and continues to specifically intend, for persons who make, use, and/or sell the Accused Instrumentalities, including AA’s vendors, customers, end users and/or others, to make, use, and/or sell such Accused Instrumentalities in a manner that infringes the asserted claims of the ’329 patent. AA encourages, directs, and instructs vendors, customers, end users and/or others in the infringing use

American Airlines/Hilton, Ex. 1018, p.5
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

and operation of the Accused Instrumentalities, via advertisement, technical material, instructional material, instructional videos, technical support, or otherwise.

As to the asserted claims of the ’317 patent, AA indirectly infringes the ’317 asserted claims in violation of 35 U.S.C. § 271(b) because AA induces direct infringement by its vendors, customers, end users and/or others who make, use, and/or sell the Accused Instrumentalities. AA has been on notice of the ’317 patent as early as October 20, 2021, when R2 explicitly informed AA of resolution of a lawsuit which involved the ’317 patent. Since receiving notice (and thus having knowledge of the claims of the ’317 patent), AA has specifically intended, and continues to specifically intend, for persons who make, use, and/or sell the Accused Instrumentalities, including AA’s vendors, customers, end users and/or others, to make, use, and/or sell such Accused Instrumentalities in a manner that infringes the asserted claims of the ’157 patent. AA encourages, directs, and instructs vendors, customers, end users and/or others in the infringing use and operation of the Accused Instrumentalities, via advertisement, technical material, instructional material, instructional videos, technical support, or otherwise

(c) A chart identifying specifically where each element of each asserted claim is found within each Accused Instrumentality, including for each element that such party contends is governed by 35 U.S.C. § 112(6), the identity of the structure(s), act(s), or materials(s) in the Accused Instrumentality that performs the claimed functions

R2 attaches the following charts setting forth where in the Accused Instrumentalities each element of the asserted claims is found:

Exhibit A: Claim chart for the asserted claims of the ’610 patent;

Exhibit B: Claim chart for the asserted claims of the ’157 patent;

Exhibit C: Claim chart for the asserted claims of the ’329 patent; and

Exhibit D: Claim chart for the asserted claims of the ’317 patent.

American Airlines/Hilton, Ex. 1018, p.6
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

R2 reserves the right to modify or seek leave of Court to augment or supplement these contentions after discovery from AA, in response to AA's production pursuant to P.R. 3-4, or as permitted under the Court's Scheduling Order, Local Rules, and the Federal Rules of Civil Procedure.

(d) Whether each element of each asserted claim is claimed to be literally present or present under the doctrine of equivalents in the Accused Instrumentality

Based on its current understanding of the claim language and publicly-available information pertaining to the Accused Instrumentalities, and without notice of any claim construction or non-infringement position from AA, R2 asserts that the Accused Instrumentalities literally infringe each element or step of the asserted claims. However, any claim element or step not literally present in or performed by the Accused Instrumentalities is satisfied under the doctrine of equivalents because any difference between such claim element or step and the accused element or step is insubstantial. In other words, the accused element or step performs substantially the same function, in substantially the same way, to achieve substantially the same result.

(e) For any patent that claims priority to an earlier application, the priority date to which each asserted claim allegedly is entitled

The patents-in-suit are entitled to a priority date no later than the filing date of the earliest application to which it claims priority. As such, the claims of the '610 patent are entitled to a priority date at least as early as October 5, 2006; the claims of the '157 patent are entitled to a priority date at least as early as July 31, 2009; the claims of the '329 patent are entitled to a priority date at least as early as January 10, 2007; and the claims of the '317 patent are entitled to a priority date at least as early as August 10, 2006.

(f) If a party claiming patent infringement wishes to preserve the right to rely, for any purpose, on the assertion that its own apparatus, product, device, process, method, act, or other instrumentality practices the claimed invention, the party must identify, separately for each asserted claim, each such apparatus, product,

American Airlines/Hilton, Ex. 1018, p.7
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

device, process, method, act, or other instrumentality that incorporates or reflects that particular claim.

R2 is not aware of any apparatus, product, device, process, method, act, or other instrumentality of its own that practices the claimed inventions of the patents-in-suit.

II. P. R. 3-2: Document Production Accompanying Disclosure

(a) R2 is not aware at this time of any non-privileged documents sufficient to evidence discussion with, disclosure to, or other manner of providing to a third party, or sale of or offer to sell, the claimed inventions prior to the date of application for the patents-in-suit, as called for by P. R. 3-2(a);

(b) R2 is not presently aware of any non-privileged documents that evidence conception, reduction to practice, design, and development of each claimed invention that were created on or before the above-referenced dates for each of the patents-in-suit. However, R2 believes that documents responsive to P. R. 3-2(b) may be in the possession of the inventors and/or original assignees of the patents-in-suit. In the event that discovery from such parties leads to documents that evidence conception, reduction to practice, design, and development of any claimed invention that were created on or before the date of application for the patents-in-suit, R2 reserves the right to claim such date.

(c) R2 has produced a copy of the file history for each of the patents-in-suit within the production number range of *WAVE4COMMON000001 - WAVE4COMMON001052*.

Dated: October 13, 2022

Respectfully submitted,

/s/ Edward R. Nelson III
EDWARD R. NELSON III
State Bar No. 00797142
ed@nelbum.com
BRENT N. BUMGARDNER
State Bar No. 00795272
brent@nelbum.com

American Airlines/Hilton, Ex. 1018, p.8
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

CHRISTOPHER G. GRANAGHAN
State Bar No. 24078585
chris@nelbum.com
JOHN P. MURPHY
State Bar No. 24056024
murphy@nelbum.com
CARDER W. BROOKS
State Bar No. 24105536
carder@nelbum.com
NELSON BUMGARDNER CONROY PC
3131 West 7th Street, Suite 300
Fort Worth, Texas 76107
817.377.9111

COUNSEL FOR
PLAINTIFF R2 SOLUTIONS LLC

CERTIFICATE OF SERVICE

I hereby certify that the foregoing document was served on all counsel of record for this matter by electronic means on October 13, 2022.

/s/ Edward R. Nelson III

American Airlines/Hilton, Ex. 1018, p.9
American Airlines/Hilton v. R2 Solutions
IPR2023-00887

Exhibit A

Preliminary Infringement Contentions

R2 Solutions LLC v. American Airlines, Inc.
Civil Action No. 4:22-cv-00353-ALM

nelson bumgardner conroy

U.S. Pat. No.: 8,190,610

Claims: 1-5, 17-21

Preliminary Infringement Contentions

R2 Solutions LLC ("R2" or "Plaintiff") contends that American Airlines, Inc. ("AA" or "Defendant") has infringed and continues to infringe U.S. Pat. No. 8,190,610 (the "'610 Patent") by, among other things, making, having made, using, offering for sale, and/or selling within this judicial district and elsewhere in the United States and/or importing into this judicial district and into the United States – without license or authority – the "Accused Instrumentalities," which include, without limitation, all instrumentalities associated with mapping and/or reducing functionalities, merging and/or joining data having different schemas, and any other similar functionality and all instrumentalities related to how data is received, input, stored, organized, partitioned, accessed, analyzed, delivered, processed, and transformed into outputs. This includes the data analytics systems utilizing, at least in part, Apache Hadoop, Hive, Spark, Teradata Vantage, and/or similar functionality. The Accused Instrumentalities practice the steps of the asserted claims or are capable of performing the claim elements of the asserted claims, or otherwise satisfy all elements, described or set forth in one or more claims of the '610 Patent in violation of at least 35 U.S.C. § 271(a). AA's infringements also include, and R2 is entitled to recover damages for, sales of infringing products and components sold and/or shipped abroad that are subsequently imported into the United States.

Preliminary Infringement Contentions

R2 contends that each and every limitation of the asserted claims of the ’610 Patent is literally present or performed and infringed by each and every Accused Instrumentality. In the alternative, R2 contends that all limitations of the asserted claims are present under the doctrine of equivalents because (1) to the extent there are any differences between the Accused Instrumentalities and those claim limitations (which R2 denies), such differences are insubstantial. Equivalency may also be shown by the fact that (2) the Accused Instrumentalities perform substantially the same function, in substantially the same way, to achieve substantially the same result as the functions recited in the limitations of the asserted claims.

R2 reserves the right to supplement or amend its claim charts, preliminary infringement contentions, and expert reports as may be appropriate or permissible under the applicable rules and Scheduling Order, including the Court’s modifications to P.R. 3-1 available in the Scheduling Order. Any reference or citation to the specification is for illustrative purposes only and is not intended to suggest that the scope of the claimed invention is limited to any particular preferred embodiment. R2 does not hereby waive any applicable work product protections.

For the avoidance of doubt, the claim elements charted herein are software limitations and/or associated with software limitations for the purposes of P.R. 3-1(g), and to the extent that such elements are performed by hardware, firmware, software, and/or any combination thereof, the particular details are in AA’s possession, custody, and control.

References Cited

The following chart presents R2 Solutions' analysis of the Accused Instrumentalities based on publicly available information. The citations in the chart refer to the following publicly available documents, which are incorporated by reference as if fully set forth herein. This infringement analysis is preliminary, and R2's investigation into AA's infringement is ongoing. R2 reserves the right to provide additional theories under which AA products/systems infringe this patent:

[1] Sachin P. Bappalige, "An introduction to Apache Hadoop for big data," opensource.com (Aug. 26, 2014), *available at* https://opensource.com/life/14/8/intro-apache-hadoop-big-data *as of 03/15/2022* ("**An introduction to Apache Hadoop for big data**")

[2] MapReduce Tutorial, *available at* http://hadoop.apache.org/docs/current/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html *as of 03/15/2022* ("**MapReduce Tutorial**")

[3] Interface Reducer<K2,V2,K3,V3>, *available at* https://hadoop.apache.org/docs/r2.8.0/api/org/apache/hadoop/mapred/Reducer.html *as of 03/15/2022* ("**Interface Reducer**")

[4] ASF Infrabot, "HadoopMapReduce," Dashboard (July 9, 2019), *available at* https://cwiki.apache.org/confluence/display/HADOOP2/HadoopMapReduce *as of 03/15/2022* ("**HadoopMapReduce**")

[5] AA Associate Developer, IT Applications, *available at* https://www.salary.com/job/american-airlines/associate-developer-it-applications/j202202092136250030017 *as of 03/15/2022* ("**AA Associate Developer**")

[6] AA Ft. Worth Associate Developer, *available at* https://jobs.aa.com/job/Ft-Worth-Associate-DeveloperDeveloper%2C-IT-Applications-TX-76101/801819300/ *as of 03/15/2022* ("**AA Ft. Worth Associate Developer**") (this position description mirrored the position description of [5] above, but the description is now unavailable because the position has been filled)

References Cited (continued)

[7] Enum Schema._Fields, *available at* https://docs.cloudera.com/HDPDocuments/HDP1/HDP-Win-1.1.0/ds_Hive/api/index.html?org/apache/hadoop/hive/metastore/api/Schema._Fields.html *as of 03/23/2022* ("**Class Schema**")

[8] Confluence Administrator, "Hive Tutorial," Dashboard (updated by Owen O'Malley, Mar. 5, 2019), *available at* https://cwiki.apache.org/confluence/display/Hive/Tutorial *as of 03/15/2022* ("**Hive Tutorial**")

[9] Confluence Administrator, "LanguageManual Joins," Dashboard (updated by Dudu Markovitz, June 5, 2017), *available at* https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Joins *as of 03/15/2022* ("**LanguageManual Joins**")

[10] Tanel Poder, "Speed Up Your Queries with Hive LLAP Engine on Hadoop or in the Cloud," Gluent, *available at* https://www.slideshare.net/gluent/speed-up-your-queries-with-hive-llap-engine-on-hadoop-or-in-the-cloud *as of 03/15/2022* ("**Speed Up Your Queries with Hive LLAP Engine**")

[11] "Apache Hive," Wikipedia, *available at* https://en.wikipedia.org/wiki/Apache_Hive *as of 03/15/2022* ("**Apache Hive**")

[12] Petar Zecevic & Marko Bonaci, "Spark in Action" (Manning Publications Nov. 2016), *available at* https://learning.oreilly.com/library/view/spark-in-action/9781617292606/kindle_split_014.html *as of 03/15/2022* ("**Spark in Action**")

[13] "Apache Spark FAQ," *available at* https://spark.apache.org/faq.html *as of 03/15/2022* ("**Apache Spark FAQ**")

[14] "Milind R," AA Engineer LinkedIn Profile, *available at* https://www.linkedin.com/in/milind-r-719395206/ *as of 03/15/2022* ("**AA Big Data Engineer**")

[15] "Rajat Gupta," AA Engineer LinkedIn Profile, *available at* https://www.linkedin.com/in/rajat-gupta-100a83209/ *as of 03/15/2022* ("**AA Data Engineer**")

[16] "AA Privacy Notice," *available at* https://www.aa.com/i18n/customer-service/support/privacy-policy.jsp ("**AA Privacy Policy**")

[17] Apache Spark 3.0.1, "RDD Programming Guide," *available at* http://spark.apache.org/docs/latest/rdd-programming-guide.html *as of 03/15/2022* ("**RDD Programming Guide**")

[18] "Learning Spark," O'Reilly Media, Inc. (February 2015), *available at* https://learning.oreilly.com/library/view/learning-spark/9781449359034/ch04.html#chap-pair-RDDS *as of 03/15/2022* ("**Learning Spark**")

References Cited (continued)

[19] "Spark RDD Transformations with examples," *available at* https://sparkbyexamples.com/apache-spark-rdd/spark-rdd-transformations/ *as of 03/15/2022* ("**Spark RDD Transformations**")

[20] Shrey Mehrotra & Akash Grade, "Apache Quick Start Guide" (Packt Publishing, Jan. 2019), *available at* https://learning.oreilly.com/library/view/apache-spark-quick/9781789349108/0ee1a5e2-09d0-49f0-99f5-9dee8336258d.xhtml *as of 03/15/2022* ("**Apache Spark Quick Start Guide**")

[21] "Spark SQL Shuffle Partitions," *available at* https://sparkbyexamples.com/spark/spark-shuffle-partitions/ *as of 10/10/2022* ("**Spark SQL Shuffle Partitions**")

[22] "Apache Spark Map vs. FlatMap Operation," DataFlair, *available at* https://data-flair.training/blogs/apache-spark-map-vs-flatmap/ *as of 03/15/2022* ("**Apache Spark Map vs. FlatMap**")

[23] Confluence Administrator, "LanguageManual Transform,"Dashboard (updated by Sushanth Sowmyan on Dec. 9, 2015), *available at* https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Transform *as of 03/15/2022* ("**LanguageManual Transform**")

[24] System Properties Comparison Hive vs. Spark SQL vs. Teradata, *available at* https://db-engines.com/en/system/Hive%3BSpark+SQL%3BTeradata *as of 10/12/2022* ("**System Properties Comparison**")

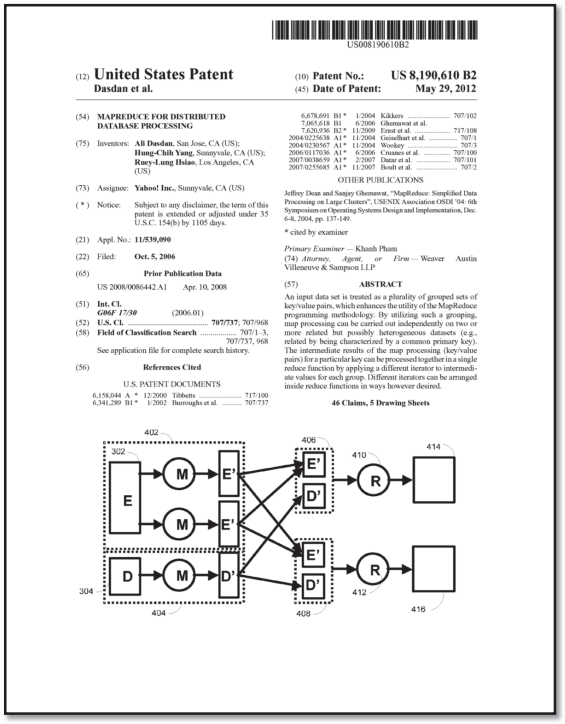
[25] Hadoop vs Teradata, *available at* https://www.educba.com/hadoop-vs-teradata/ *as of 10/12/2022* ("**Hadoop vs Teradata**")

[26] Teradata Vantage User Guide, *available at* https://docs.teradata.com/r/Teradata-Vantage-User-Guide/January-2019/Access-to-Commands-and-Functions/Teradata-ML-Engine-Analytic-Functions/Teradata-ML-Engine-Analytic-Function-Execution-SQL-Syntax *as of 10/12/2022* ("**Teradata Vantage User Guide**")

[27] AA Using Teradata, *available at* https://www.teradata.com/Customers/American-Airlines *as of 10/12/2022* ("**AA Using Teradata**")

[28] Hadoop Migration Program from Teradata, *available at* https://www.teradata.com/Hadoop-Migration-Program *as of 10/12/2022* ("Hadoop Migration Program")

U.S. Patent No. 8,190,610



US008190610B2

(12) **United States Patent**
Dasdan et al.

(10) **Patent No.: US 8,190,610 B2**
(45) **Date of Patent: May 29, 2012**

(54) **MAPREDUCE FOR DISTRIBUTED DATABASE PROCESSING**

(75) Inventors: **Ali Dasdan**, San Jose, CA (US); **Hung-Chih Yang**, Sunnyvale, CA (US); **Ruey-Lung Hsiao**, Los Angeles, CA (US)

(73) Assignee: **Yahoo! Inc.**, Sunnyvale, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 1105 days.

(21) Appl. No.: **11/539,090**

(22) Filed: **Oct. 5, 2006**

(65) **Prior Publication Data**

US 2008/0086442 A1 Apr. 10, 2008

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.** **707/737**; 707/968

(58) **Field of Classification Search** 707/1–3, 707/737, 968

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,158,044 A * 12/2000 Tibbetts 717/100
6,341,289 B1 * 1/2002 Burroughs et al. 707/737
6,678,691 B1 * 1/2004 Kikkers 707/102
7,065,618 B1 6/2006 Ghemawat et al.
7,620,936 B2 * 11/2009 Ernst et al. 717/108
2004/0225638 A1 * 11/2004 Geiselhart et al. 707/1
2004/0230567 A1 * 11/2004 Wookey 707/3
2006/0117036 A1 * 6/2006 Cruanes et al. 707/100
2007/0038659 A1 * 2/2007 Datar et al. 707/101
2007/0255685 A1 * 11/2007 Boult et al. 707/2

OTHER PUBLICATIONS

Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", USENIX Association OSDI '04: 6th Symposium on Operating Systems Design and Implementation, Dec. 6-8, 2004, pp. 137-149.

* cited by examiner

Primary Examiner — Khanh Pham
(74) *Attorney, Agent, or Firm* — Weaver Austin Villeneuve & Sampson LLP

(57) **ABSTRACT**

An input data set is treated as a plurality of grouped sets of key/value pairs, which enhances the utility of the MapReduce programming methodology. By utilizing such a grouping, map processing can be carried out independently on two or more related but possibly heterogeneous datasets (e.g., related by being characterized by a common primary key). The intermediate results of the map processing (key/value pairs) for a particular key can be processed together in a single reduce function by applying a different iterator to intermediate values for each group. Different iterators can be arranged inside reduce functions in ways however desired.

46 Claims, 5 Drawing Sheets

Title:	**MAPREDUCE FOR DISTRIBUTED DATABASE PROCESSING**
Priority Date:	**October 5, 2006**
Filing Date:	**October 5, 2006**
Issue Date:	**May 29, 2012**
Expiration Date:	**October 14, 2029**

(57) **ABSTRACT**

An input data set is treated as a plurality of grouped sets of key/value pairs, which enhances the utility of the MapReduce programming methodology. By utilizing such a grouping, map processing can be carried out independently on two or more related but possibly heterogeneous datasets (e.g., related by being characterized by a common primary key). The intermediate results of the map processing (key/value pairs) for a particular key can be processed together in a single reduce function by applying a different iterator to intermediate values for each group. Different iterators can be arranged inside reduce functions in ways however desired.

Claim 1

1[pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:

1[a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,

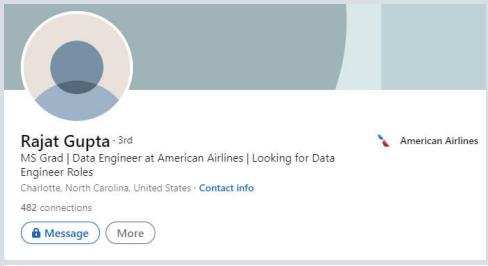
1[b] wherein the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and

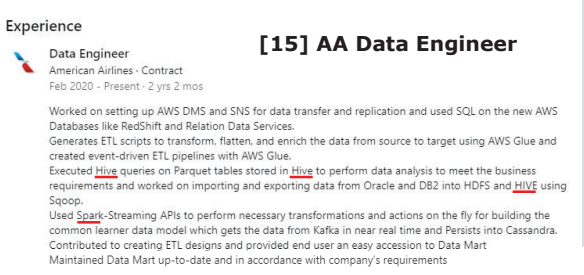
1[c] reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a merging of the corresponding different intermediate data based on the key in common,

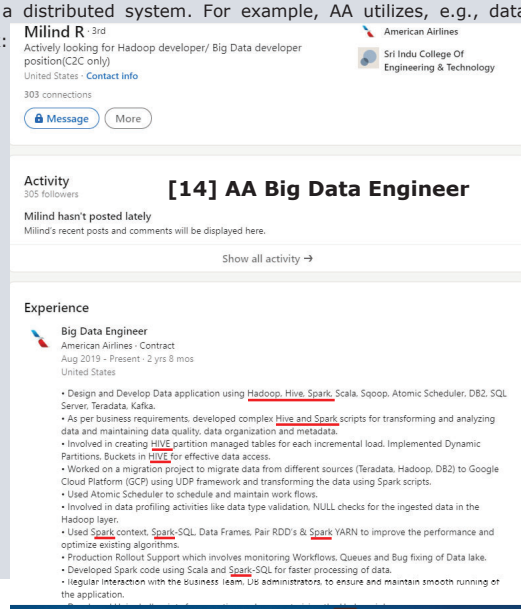
1[d] wherein the mapping and reducing operations are performed by a distributed system.

Claim Chart – Claim 1

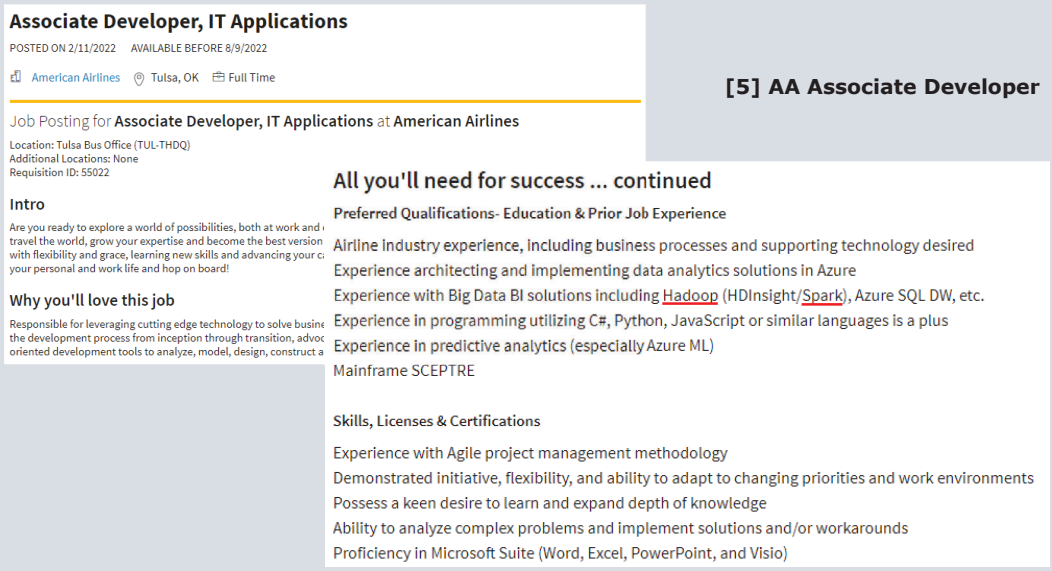
Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	AA infringes Claim 1 of the '610 Patent. Each of the Accused Instrumentalities perform a method of processing data of a data se over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising the steps discussed herein. The Accused Instrumentalities are based, at least in part, on Hadoop, Hive, Spark, Teradata Vantage, and/or othe functionality, which enables processing of data of a data set over a distributed system. For example, AA utilizes, e.g., data engineers whose duties specifically include Hadoop, Hive, and/or Spark:







Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	In another example, AA employs, e.g., developers qualified in big data solutions such as Hadoop and Spark:  **[5] AA Associate Developer**

Associate Developer, IT Applications

POSTED ON 2/11/2022 AVAILABLE BEFORE 8/9/2022

American Airlines Tulsa, OK Full Time

Job Posting for **Associate Developer, IT Applications** at **American Airlines**

Location: Tulsa Bus Office (TUL-THDQ)
Additional Locations: None
Requisition ID: 55022

Intro

Are you ready to explore a world of possibilities, both at work and
travel the world, grow your expertise and become the best version
with flexibility and grace, learning new skills and advancing your c
your personal and work life and hop on board!

Why you'll love this job

Responsible for leveraging cutting edge technology to solve busine
the development process from inception through transition, advoc
oriented development tools to analyze, model, design, construct a

All you'll need for success ... continued

Preferred Qualifications- Education & Prior Job Experience

Airline industry experience, including business processes and supporting technology desired
Experience architecting and implementing data analytics solutions in Azure
Experience with Big Data BI solutions including Hadoop (HDInsight/Spark), Azure SQL DW, etc.
Experience in programming utilizing C#, Python, JavaScript or similar languages is a plus
Experience in predictive analytics (especially Azure ML)
Mainframe SCEPTRE

Skills, Licenses & Certifications

Experience with Agile project management methodology
Demonstrated initiative, flexibility, and ability to adapt to changing priorities and work environments
Possess a keen desire to learn and expand depth of knowledge
Ability to analyze complex problems and implement solutions and/or workarounds
Proficiency in Microsoft Suite (Word, Excel, PowerPoint, and Visio)

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	In another example, AA utilizes Teradata Vantage, which employs similar functionality to Hadoop, Hive, and/or Spark:

American Airlines

American Airlines is the world's largest airline when measured by fleet size, scheduled passengers carried, and number of destinations served. Its mission is to "care for people on life's journey."

The airline averages 6,800 flights per day to more than 365 destinations in 61 countries and in 2021, safely transported 164M passengers. This impressive airline fuels business-critical operations across multiple business departments performing analytics for revenue management, technical operations, network planning, marketing, cargo, passenger sales, and more, all using Teradata Vantage as its connected cloud data platform. Data is the fuel for American Airlines and analytics are the propulsion that lifts American Airlines above the everyday turbulence encountered by airline operators.

[27] AA Using Teradata

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	Teradata Vantage employs similar functionality to Hadoop, Hive, and/or Spark:

System Properties Comparison Hive vs. Spark SQL vs. Teradata

Please select another system to include it in the comparison.

Editorial information provided by DB-Engines

Name	Hive X	Spark SQL X	Teradata X
Description	data warehouse software for querying and managing large distributed datasets, built on Hadoop	Spark SQL is a component on top of 'Spark Core' for structured data processing	A hybrid cloud data analytics software platform (Teradata Vantage)
Primary database model	Relational DBMS	Relational DBMS	Relational DBMS
Secondary database models			Document store Graph DBMS Spatial DBMS Time Series DBMS
DB-Engines Ranking Trend Chart	Score 80.60 Rank #17 Overall #11 Relational DBMS	Score 22.58 Rank #36 Overall #22 Relational DBMS	Score 66.07 Rank #18 Overall #12 Relational DBMS
Website	hive.apache.org	spark.apache.org/sql	www.teradata.com
Technical documentation	cwiki.apache.org/confluence/display/Hive/Home	spark.apache.org/docs/latest/sql-programming-guide.html	docs.teradata.com
Developer	Apache Software Foundation	Apache Software Foundation	Teradata
Initial release	2012	2014	1984
Current release	3.1.3, April 2022	3.3.0 (2.13), June 2022	Teradata Vantage 1.0 MU2, January 2019
License	Open Source	Open Source	commercial
Cloud-based only	no	no	no
DBaaS offerings (sponsored links)			
Implementation language	Java	Scala	
Server operating systems	All OS with a Java VM	Linux OS X Windows	hosted Linux

[24] System Properties Comparison

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	Teradata Vantage employs similar functionality to Hadoop, Hive, and/or Spark:

Comparison Table Between Hadoop and Teradata

Below are the lists of points, describe the Differences between Hadoop and Teradata :

Basis of Comparison	Teradata	Hadoop
Parallel Processing	Workload is divided across the system and evenly among the processors in the system.	Workload is divided among the different nodes on which relevant data is present and each node processes the task individually in parallel which reduces the overall time taken to complete the task.
Share-nothing Architecture	Teradata task executing in a virtual processor is independent of the tasks in other virtual processors.	Task execution on any node of the Hadoop is independent to tasks executing on other nodes.
Highly Scalable	More nodes/disks can be added but will increase the licensing cost.	More number of nodes/disks can be added as and when required to increase the processing and storage power.

[25] Hadoop vs Teradata

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	Teradata Vantage employs similar functionality to Hadoop, Hive, and/or Spark: **The Best Alternative to Hadoop** The Teradata Hadoop Migration Program is a lifeline for companies who need an alternative to Hadoop due to the high cost, production difficulty, infrastructure complexity, lack of skills, and limited insight that comes with the best-intentioned Hadoop deployments. Teradata Vantage is the Hadoop replacement proven to drive maximum business value through analytical insight. This program quickly, easily migrates existing Hadoop data and workloads to Vantage and cloud low-cost object storage using robust and proven migration methodologies and tools. **[28] Hadoop Migration Program**

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	The Accused Instrumentalities are based on Hadoop, Hive, Spark, Teradata Vantage, and/or other functionality, which enables processing of data of a data set over a distributed system. **[10] Speed Up Your Queries with Hive LLAP Engine at 12:**

Introducing Hive LLAP

- "Live Long and Process" or "Low Latency Analytical Processing"
  - Not an execution engine (like Tez), LLAP simply enhances the Hive execution model
  - Built for fast query response time against smaller data volumes
  - Allows concurrent execution of analytical workloads
- Intelligent memory caching for quick startup and data sharing
  - Caches most active data in RAM
  - Shared cache across clients
- Persistent server used to instantly execute queries
  - LLAP daemons are "always on"
  - Data passed to execution as it becomes ready

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a **distributed system**, wherein the data set comprises a plurality of data groups, the method comprising:	Each of the Accused Instrumentalities are implemented over a distributed system: **[1] An introduction to Apache Hadoop for big data at 2:** **The Apache Hadoop framework is composed of the following modules** 1. Hadoop Common: contains libraries and utilities needed by other Hadoop modules 2. Hadoop Distributed File System (HDFS): a distributed file-system that stores data on the commodity machines, providing very high aggregate bandwidth across the cluster 3. Hadoop YARN: a resource-management platform responsible for managing compute resources in clusters and using them for scheduling of users' applications 4. Hadoop MapReduce: a programming model for large scale data processing All the modules in Hadoop are designed with a fundamental assumption that hardware failures (of individual machines, or racks of machines) are common and thus should be automatically handled in software by the framework. Apache Hadoop's MapReduce

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a **distributed system**, wherein the data set comprises a plurality of data groups, the method comprising:	Each of the Accused Instrumentalities are implemented over a distributed system: **[11] Apache Hive:** **Apache Hive** is a data warehouse software project built on top of Apache Hadoop for providing data query and analysis.[3] Hive gives an SQL-like interface to query data stored in various databases and file systems that integrate with Hadoop. Traditional SQL queries must be implemented in the MapReduce Java API to execute SQL applications and queries over distributed data. Hive provides the necessary SQL abstraction to integrate SQL-like queries (HiveQL) into the underlying Java without the need to implement queries in the low-level Java API. Since most data warehousing applications work with SQL-based querying languages, Hive aids portability of SQL-based applications to Hadoop.[4] While initially developed by Facebook, Apache Hive is used and developed by other companies such as Netflix and the Financial Industry Regulatory Authority (FINRA).[5][6] Amazon maintains a software fork of Apache Hive included in Amazon Elastic MapReduce on Amazon Web Services.[7]

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] **A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising**:	The Accused Instrumentalities include a method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups:

Comparison Table Between Hadoop and Teradata

Below are the lists of points, describe the Differences between Hadoop and Teradata :

Basis of Comparison	Teradata	Hadoop
Parallel Processing	Workload is divided across the system and evenly among the processors in the system.	Workload is divided among the different nodes on which relevant data is present and each node processes the task individually in parallel which reduces the overall time taken to complete the task.
Share-nothing Architecture	Teradata task executing in a virtual processor is independent of the tasks in other virtual processors.	Task execution on any node of the Hadoop is independent to tasks executing on other nodes.
Highly Scalable	More nodes/disks can be added but will increase the licensing cost.	More number of nodes/disks can be added as and when required to increase the processing and storage power.

[25] Hadoop vs Teradata

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, **wherein the data set comprises a plurality of data groups**, the method comprising:	The Accused Instrumentalities include a data set comprising a plurality of data groups. For example, AA collects a myriad of data from a variety of sources (e.g., websites, customers, vendors, etc.):

[16] AA Privacy Policy

Information we collect and how we collect it

We collect and maintain personal information about you from many sources to understand and meet your needs, facilitate your travel, manage our business, and for other purposes disclosed to you. For example, we collect personal information about you from:

- You, when you voluntarily provide us with information
- Your transactions with American
- Other third-party sources, such as travel agents, other airlines (e.g., code share and alliance partners) and travel and hospitality service providers (e.g., transportation or tour operators).

If the information is to be collected directly from you, you may in some cases have the option to decline providing that information. However, your choice to not provide information may impact your use of certain features or services.

The personal information we collect about you through these various sources may include, but is not limited to:

- Name, addresses and telephone numbers
- Date of birth and gender
- Email addresses, fax numbers and pager numbers
- Social media handles, including when you communicate with us through social media
- AAdvantage® account number
- Credit/debit card payment information, including card number(s), associated billing address(es), and expiration date(s)
- Information required to facilitate travel or other services, including travel companion(s) names, emergency contacts, and photographs
- Information about your travel, including booking and itinerary information, requests for assistance, interactions with staff and cabin crew, and dietary restrictions
- Digital identity credentials (such as credentials linked to facial or fingerprint verification, vaccination status, or negative test results) that you provide for specific purposes, such as compliance with customs and immigration requirements during international travel or services in which you choose to participate
- Details of your previous travel, such as previous flights and travel-related issues
- Driver's license number
- Passport number, nationality and country of residence
- Technical information such as your browser type, IP address, type of operating system you use, your geolocation, the name of your internet service provider, mobile advertising identifiers, and pages visited on our Services
- Information provided via survey, focus group or other marketing research efforts
- When you send us an email or online customer service request, we will retain the content of the message, your email address or any contact information you provide, and our response in order to handle any follow-up questions you may have. We also use this information to measure how effectively we address your concerns.
- Corporate-contract, employer and/or other corporate affiliation (e.g., employer name, title, work address and contact information), including Business Extra® account information (e.g., tax ID, business type, number of employees, number of business travelers, and travel manager information)

Claim Chart – Claim 1

Claim Feature	Evidence
1 [pre] A method of processing data of a data set over a distributed system, wherein the data set comprises a plurality of data groups, the method comprising:	In another example, Hadoop runs in clusters, indicating that the data set comprises a plurality of data groups. Additionally, Spark revolves around RDDs which are a plurality of elements (e.g., data groups) that can be operated in parallel. **[13] Apache Spark FAQ at 1:** How does Spark relate to Apache Hadoop? Spark is a fast and general processing engine compatible with Hadoop data. It can run in Hadoop clusters through YARN or Spark's standalone mode, and it can process data in HDFS, HBase, Cassandra, Hive, and any Hadoop InputFormat. It is designed to perform both batch processing (similar to MapReduce) and new workloads like streaming, interactive queries, and machine learning. **[17] RDD Programming Guide at 4, 2:** **Resilient Distributed Datasets (RDDs)** Spark revolves around the concept of a *resilient distributed dataset* (RDD), which is a fault-tolerant collection of elements that can be operated on in parallel. There are two ways to create RDDs: *parallelizing* an existing collection in your driver program, or referencing a dataset in an external storage system, such as a shared filesystem, HDFS, HBase, or any data source offering a Hadoop InputFormat. **Overview** At a high level, every Spark application consists of a *driver program* that runs the user's `main` function and executes various *parallel operations* on a cluster. The main abstraction Spark provides is a *resilient distributed dataset* (RDD), which is a collection of elements partitioned across the nodes of the cluster that can be operated on in parallel. RDDs are created by starting with a file in the Hadoop file system (or any other Hadoop-supported file system), or an existing Scala collection in the driver program, and transforming it. Users may also ask Spark to *persist* an RDD in memory, allowing it to be reused efficiently across parallel operations. Finally, RDDs automatically recover from node failures.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] **partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable** to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	The method performed by the Accused Instrumentalities include partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group. For example, in the Accused Instrumentalities' Hadoop MapReduce framework, the input data is partitioned into independent chunks (data partitions), which are processed by Mapper functions: **[2] MapReduce Tutorial at 2:** A MapReduce *job* usually splits the input data-set into independent chunks which are processed by the *map tasks* in a completely parallel manner. The framework sorts the outputs of the maps, which are then input to the *reduce tasks*. Typically both the input and the output of the job are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] **partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable** to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	The data partitions with key-value pairs are passed to user-configurable Mapper functions for further processing: **[4] HadoopMapReduce at 1:** As key-value pairs are read from the RecordReader they are passed to the configured Mapper. The user supplied Mapper does whatever it wants with the input pair and calls OutputCollector.collect with key-value pairs of its own choosing. The output it generates must use one key class and one value class. This is because the Map output will be written into a SequenceFile which has per-file type information and all the records must have the same type (use subclassing if you want to output different data structures). The Map input and output key-value pairs are not necessarily related typewise or in cardinality.

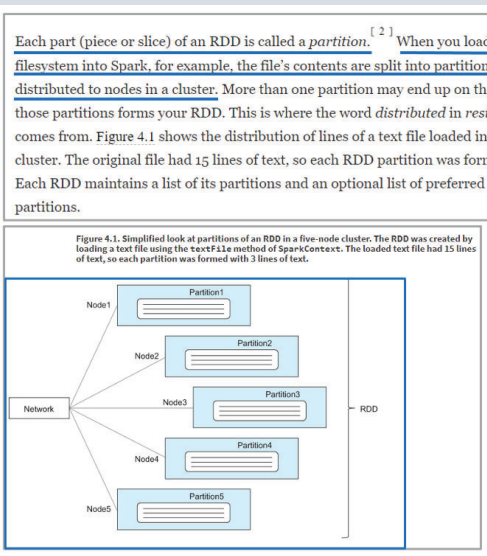
Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition **to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group**,	The Mapper function maps the key-value pairs to form intermediate key-value pairs: **[2] MapReduce Tutorial at 7:** **Mapper** Mapper maps input key/value pairs to a set of intermediate key/value pairs. Maps are the individual tasks that transform input records into intermediate records. The transformed intermediate records do not need to be of the same type as the input records. A given input pair may map to zero or many output pairs.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] **partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs** and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	In another example, with Spark, a data set is first partitioned into a plurality of elements and distributed across nodes This distribution of elements is called a resilient distributed dataset (RDD): **[17] RDD Programming Guide at 2, 4, 5:** **Overview** At a high level, every Spark application consists of a *driver program* that runs the user's `main` function and executes various *parallel operations* on a cluster. The main abstraction Spark provides is a *resilient distributed dataset* (RDD), which is a collection of elements partitioned across the nodes of the cluster that can be operated on in parallel. RDDs are created by starting with a file in the Hadoop file system (or any other Hadoop-supported file system), or an existing Scala collection in the driver program, and transforming it. Users may also ask Spark to *persist* an RDD in memory, allowing it to be reused efficiently across parallel operations. Finally, RDDs automatically recover from node failures. **Resilient Distributed Datasets (RDDs)** Spark revolves around the concept of a *resilient distributed dataset* (RDD), which is a fault-tolerant collection of elements that can be operated on in parallel. There are two ways to create RDDs: *parallelizing* an existing collection in your driver program, or referencing a dataset in an external storage system, such as a shared filesystem, HDFS, HBase, or any data source offering a Hadoop InputFormat. One important parameter for parallel collections is the number of *partitions* to cut the dataset into. Spark will run one task for each partition of the cluster. Typically you want 2-4 partitions for each CPU in your cluster. Normally, Spark tries to set the number of partitions automatically based on your cluster. However, you can also set it manually by passing it as a second parameter to `parallelize` (e.g. `sc.parallelize(data, 10)`). Note: some places in the code use the term slices (a synonym for partitions) to maintain backward compatibility.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] **partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions** that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	**[12]** **Spark in Action:** Each part (piece or slice) of an RDD is called a *partition*.[2] When you load a text file from your local filesystem into Spark, for example, the file's contents are split into partitions, which are evenly distributed to nodes in a cluster. More than one partition may end up on the same node. The sum of all those partitions forms your RDD. This is where the word *distributed* in *resilient distributed dataset* comes from. Figure 4.1 shows the distribution of lines of a text file loaded into an RDD in a five-node cluster. The original file had 15 lines of text, so each RDD partition was formed with 3 lines of text. Each RDD maintains a list of its partitions and an optional list of preferred locations for computing the partitions. Figure 4.1. Simplified look at partitions of an RDD in a five-node cluster. The RDD was created by loading a text file using the `textFile` method of `SparkContext`. The loaded text file had 15 lines of text, so each partition was formed with 3 lines of text. 

Claim Chart – Claim 1

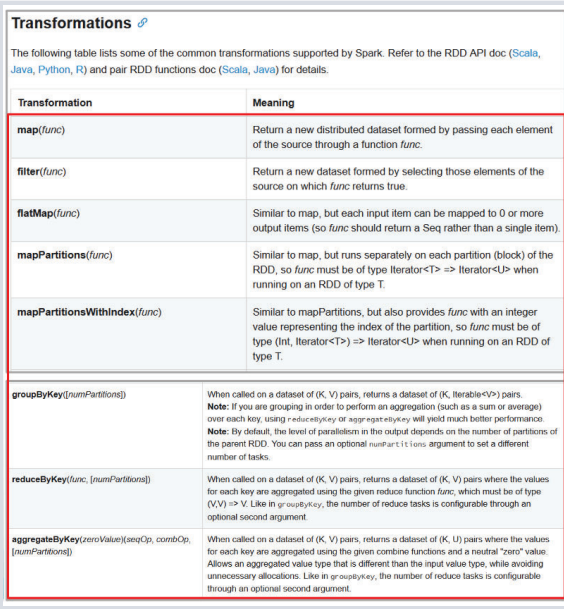
Claim Feature	Evidence
1 [a] **partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs** and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	In the Accused Instrumentalities, the data partitions have a plurality of key-value pairs. For example, one form of the RDDs is a pair RDD, which includes key/value pairs: **[18] Learning Spark:** Spark provides special operations on RDDs containing key/value pairs. These RDDs are called pair RDDs. Pair RDDs are a useful building block in many programs, as they expose operations that allow you to act on each key in parallel or regroup data across the network. For example, pair RDDs have a `reduceByKey()` method that can aggregate data separately for each key, and a `join()` method that can merge two RDDs together by grouping elements with the same key. It is common to extract fields from an RDD (representing, for instance, an event time, customer ID, or other identifier) and use those fields as keys in pair RDD operations. **[12] Spark in Action:** **4.1. WORKING WITH PAIR RDDS** Storing data as key-value pairs offers a simple, general, and extensible data model because each key-value pair can be stored independently and it's easy to add new types of keys and new types of values. This extensibility and simplicity have made this practice fundamental to several frameworks and applications. For example, many popular caching systems and NoSQL databases, such as memcached and Redis, are key-value stores. Hadoop's MapReduce also operates on key-value pairs (as you can see in appendix A). Keys and values can be simple types such as integers and strings, as well as more-complex data structures. Data structures traditionally used to represent key-value pairs are *associative arrays*, also called *dictionaries* in Python and *maps* in Scala and Java. In Spark, RDDs containing key-value tuples are called *pair RDDs*. Although you don't have to use data in Spark in the form of key-value pairs (as you've seen in the previous chapters), pair RDDs are well suited (and indispensable) for many use cases. Having keys along with the data enables you to aggregate, sort, and join the data, as you'll soon see. But before doing any of that, the first step is to create a pair RDD, of course.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and **providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable** to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition **to form corresponding intermediate data for that data group and identifiable to that data group**,	The Accused Instrumentalities provide each data partition to one of a plurality of mapping functions. For example, the RDDs support two types of data operations: transformations and actions. **[17] RDD Programming Guide at 6:** **RDD Operations** RDDs support two types of operations: *transformations*, which create a new dataset from an existing one, and *actions*, which return a value to the driver program after running a computation on the dataset. For example, `map` is a transformation that passes each dataset element through a function and returns a new RDD representing the results. On the other hand, `reduce` is an action that aggregates all the elements of the RDD using some function and returns the final result to the driver program (although there is also a parallel `reduceByKey` that returns a distributed dataset). All transformations in Spark are *lazy*, in that they do not compute their results right away. Instead, they just remember the transformations applied to some base dataset (e.g. a file). The transformations are only computed when an action requires a result to be returned to the driver program. This design enables Spark to run more efficiently. For example, we can realize that a dataset created through `map` will be used in a `reduce` and return only the result of the `reduce` to the driver, rather than the larger mapped dataset.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and **providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable** to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition **to form corresponding intermediate data for that data group and identifiable to that data group**,	Examples of transformation functions: **[17] RDD Programming Guide at 11–12:**

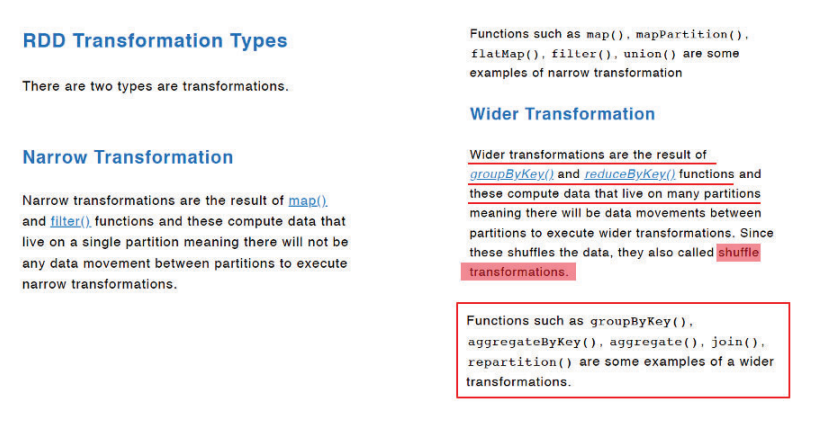


Transformations

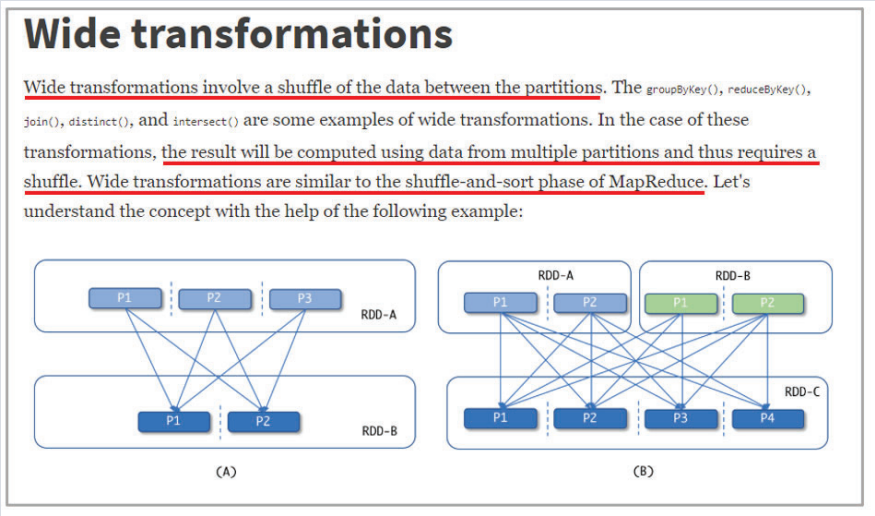
The following table lists some of the common transformations supported by Spark. Refer to the RDD API doc (Scala, Java, Python, R) and pair RDD functions doc (Scala, Java) for details.

Transformation	Meaning
map(*func*)	Return a new distributed dataset formed by passing each element of the source through a function *func*.
filter(*func*)	Return a new dataset formed by selecting those elements of the source on which *func* returns true.
flatMap(*func*)	Similar to map, but each input item can be mapped to 0 or more output items (so *func* should return a Seq rather than a single item).
mapPartitions(*func*)	Similar to map, but runs separately on each partition (block) of the RDD, so *func* must be of type Iterator<T> => Iterator<U> when running on an RDD of type T.
mapPartitionsWithIndex(*func*)	Similar to mapPartitions, but also provides *func* with an integer value representing the index of the partition, so *func* must be of type (Int, Iterator<T>) => Iterator<U> when running on an RDD of type T.
groupByKey([*numPartitions*])	When called on a dataset of (K, V) pairs, returns a dataset of (K, Iterable<V>) pairs. **Note:** If you are grouping in order to perform an aggregation (such as a sum or average) over each key, using `reduceByKey` or `aggregateByKey` will yield much better performance. **Note:** By default, the level of parallelism in the output depends on the number of partitions of the parent RDD. You can pass an optional `numPartitions` argument to set a different number of tasks.
reduceByKey(*func*, [*numPartitions*])	When called on a dataset of (K, V) pairs, returns a dataset of (K, V) pairs where the values for each key are aggregated using the given reduce function *func*, which must be of type (V,V) => V. Like in `groupByKey`, the number of reduce tasks is configurable through an optional second argument.
aggregateByKey(*zeroValue*)(*seqOp*, *combOp*, [*numPartitions*])	When called on a dataset of (K, V) pairs, returns a dataset of (K, U) pairs where the values for each key are aggregated using the given combine functions and a neutral "zero" value. Allows an aggregated value type that is different than the input value type, while avoiding unnecessary allocations. Like in `groupByKey`, the number of reduce tasks is configurable through an optional second argument.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and **providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable** to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition **to form corresponding intermediate data for that data group and identifiable to that data group**,	Exemplary transformation operations include narrow and wide transformations. Wide transformations (also called "shuffle") operate on data from multiple partitions. Wide transformations include aggregateByKey, reduceByKey, etc. **[19] Spark RDD Transformations at 2–3:**  **RDD Transformation Types** There are two types are transformations. **Narrow Transformation** Narrow transformations are the result of map() and filter() functions and these compute data that live on a single partition meaning there will not be any data movement between partitions to execute narrow transformations. Functions such as `map()`, `mapPartition()`, `flatMap()`, `filter()`, `union()` are some examples of narrow transformation **Wider Transformation** Wider transformations are the result of groupByKey() and reduceByKey() functions and these compute data that live on many partitions meaning there will be data movements between partitions to execute wider transformations. Since these shuffles the data, they also called shuffle transformations. Functions such as `groupByKey()`, `aggregateByKey()`, `aggregate()`, `join()`, `repartition()` are some examples of a wider transformations.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and **providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable** to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition **to form corresponding intermediate data for that data group and identifiable to that data group**,	**[20]** **Apache Spark Quick Start Guide:** 

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and **providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group**,	Another example of a mapping function is a shuffle transformation, which, when executed, runs a map task on all data partitions first. **[21] Spark SQL Shuffle Partitions:** **What is Spark Shuffle?** Shuffling is a mechanism Spark uses to redistribute the data across different executors and even across machines. Spark shuffling triggers for transformation operations like `gropByKey()`, `reducebyKey()`, `join()`, `groupBy()` e.t.c Spark Shuffle is an expensive operation since it involves the following • Disk I/O • Involves data serialization and deserialization • Network I/O When creating an RDD, Spark doesn't necessarily store the data for all keys in a partition since at the time of creation there is no way we can set the key for the data set. Hence, when we run the reduceByKey() operation to aggregate the data on keys, Spark does the following. • Spark first runs map tasks on all partitions which groups all values for a single key. • The results of the map tasks are kept in memory. • When results do not fit in memory, Spark stores the data on a disk. • Spark shuffles the mapped data across partitions, some times it also stores the shuffled data into a disk for reuse when it needs to recalculate. • Run the garbage collection • Finally runs reduce tasks on each partition based on key.

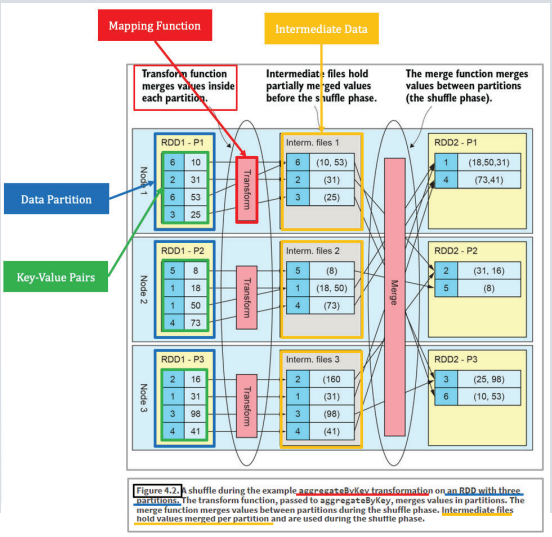
Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of **a plurality of mapping functions that are each user-configurable** to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	The Accused Instrumentalities include a plurality of mapping functions that are each user-configurable. **[22] Apache Spark Map vs. FlatMap at 3–4:** i. Spark Map Transformation A **map** is a transformation operation in **Apache Spark.** It applies to each element of **RDD** and it returns the result as new RDD. In the Map, operation developer can define his own custom business logic. The same logic will be applied to all the elements of RDD. Spark Map function takes one element as input process it according to custom code (specified by the developer) and returns one element at a time. Map transforms an **RDD** of length N into another RDD of length N. The input and output RDDs will typically have the same number of records.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	When a mapping function is applied, intermediate files are created for that partition. For example, as seen below, Spark takes the key values and writes that data to intermediate files on each node of the distributed system. **[12] Spark in Action:** **4.2.2. Understanding and avoiding unnecessary shuffling** Physical movement of data between partitions is called *shuffling*. It occurs when data from multiple partitions needs to be combined in order to build partitions for a new RDD. When grouping elements by key, for example, Spark needs to examine all of the RDD's partitions, find elements with the same key, and then physically group them, thus forming new partitions. To visualize what happens with partitions during a shuffle, we'll use the previous example with the `aggregateByKey` transformation (from section 4.1.2). The shuffle that occurs during that transformation is illustrated in figure 4.2. The transform function puts all values of each key in a single partition (partitions P1 to P3) into a list. Spark then writes this data to intermediate files on each node. In the next phase, the merge function is called to merge lists from different partitions, but of the same key, into a single list for each key. The default partitioner (`HashPartitioner`) then kicks in and puts each key in its proper partition. Tasks that immediately precede and follow the shuffle are called *map* and *reduce* tasks, respectively. The results of map tasks are written to intermediate files (often to the OS's filesystem cache only) and read by reduce tasks. In addition to being written to disk, the data is sent over the network, so it's important to try to minimize the number of shuffles during Spark jobs.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	In the below example, each of the plurality of data partitions has a plurality of key-value pairs. Each data partition (e.g., RDD1 – P1 on Node 1) is provided to mapping functions (e.g., the pictured transform function). The transform function is user-configurable. Each transform function outputs a plurality of list files (e.g., files 6, 2, 3, in Interim files 1). This intermediate data each correspond to a set of keys in the data partition (see the arrows between each key in RDD1 – P1 and Interim files 1). These values are identifiable by the data group. **[12] Spark in Action:** 

Claim Chart – Claim 1

Claim Feature	Evidence
1 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	In another example, Teradata Vantage includes partitioning the data of each one of the data groups into a plurality o data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values fo each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group. The particular details are in AA's possession, custody, and control. American Airlines **American Airlines is the world's largest airline when measured by fleet size, scheduled passengers carried, and number of destinations served. Its mission is to "care for people on life's journey."** The airline averages 6,800 flights per day to more than 365 destinations in 61 countries and in 2021, safely transported 164M passengers. This impressive airline fuels business-critical operations across multiple business departments performing analytics for revenue management, technical operations, network planning, marketing, cargo, passenger sales, and more, all using Teradata Vantage as its connected cloud data platform. Data is the fuel for American Airlines and analytics are the propulsion that lifts American Airlines above the everyday turbulence encountered by airline operators. **[27] AA Using Teradata**

Claim Chart – Claim 1

Claim Feature	Evidence

1 [b] wherein **the data of a first data group has a different schema than the data of a second data group** and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and	In the Accused Instrumentalities, the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common. For example, the Schema class in Hive is defined for each data set. Thus, the data of a first data group can have a different schema than the data of a second data group.
	
	[7] Class Schema:
	
	`public static enum Schema._Fields`
	`extends Enum<Schema._Fields>`
	`implements org.apache.thrift.TFieldIdEnum`
	
	The set of fields this struct contains, along with convenience methods for finding and manipulating them.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [b] wherein the data of a first data group has a different schema than the data of a second data group and **the data of the first data group is mapped differently than the data of the second data group** so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and	In the Accused Instrumentalities, the data of the first data group if mapped differently than the data of the second data group. For example, data partitions can be processed by using different user-configured Mappers: **[8] Hive Tutorial at 15:** Users can also plug in their own custom mappers and reducers in the data stream by using features natively supported in the Hive language, for example, in order to run a custom mapper script - map_script - and a custom reducer script - reduce_script - the user can issue the following command which uses the TRANSFORM clause to embed the mapper and the reducer scripts. **[4] HadoopMapReduce:** As key-value pairs are read from the RecordReader they are passed to the configured Mapper. The user supplied Mapper does whatever it wants with the input pair and calls OutputCollector.collect with key-value pairs of its own choosing. The output it generates must use one key class and one value class. This is because the Map output will be written into a SequenceFile which has per-file type information and all the records must have the same type (use subclassing if you want to output different data structures). The Map input and output key-value pairs are not necessarily related typewise or in cardinality.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [b] wherein the data of a first data group has a different schema than the data of a second data group and the data of the first data group is **mapped differently** than the data of the second data group **so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common**; and	The Mapper function maps the key-value pairs to form intermediate key-value pairs. Similarly, different Mappers generate different intermediate results: **[2] MapReduce Tutorial:** Mapper maps input key/value pairs to a set of intermediate key/value pairs. Maps are the individual tasks that transform input records into intermediate records. The transformed intermediate records do not need to be of the same type as the input records. A given input pair may map to zero or many output pairs.

Claim Chart – Claim 1

Claim Feature	Evidence

1 [b] wherein the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that **different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common**; and	A Hive JOIN query may be used to join two datasets that have a key in common:
	
	[9] LanguageManual Joins:
	
	Examples
	
	Some salient points to consider when writing join queries are as follows:
	
	• Complex join expressions are allowed e.g.
	
	`SELECT a.* FROM a JOIN b ON (a.id = b.id)`
	
	`SELECT a.* FROM a JOIN b ON (a.id = b.id AND a.department = b.department)`
	
	`SELECT a.* FROM a LEFT OUTER JOIN b ON (a.id <> b.id)`

Claim Chart – Claim 1

Claim Feature	Evidence
1 [b] **wherein the data of a first data group has a different schema than the data of a second data group** and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and	Spark supports multiple types of structure data, each with its own schema. **[18] Learning Spark:** **Structured Data with Spark SQL** Spark SQL is a component added in Spark 1.0 that is quickly becoming Spark's preferred way to work with structured and semistructured data. By structured data, we mean data that has a *schema*—that is, a consistent set of fields across data records. Spark SQL supports multiple structured data sources as input, and because it understands their schema, it can efficiently read only the fields you require from these data sources. We will cover Spark SQL in more detail in Chapter 9, but for now, we show how to use it to load data from a few common sources. **Apache Hive** One common structured data source on Hadoop is Apache Hive. Hive can store tables in a variety of formats, from plain text to column-oriented formats, inside HDFS or other storage systems. Spark SQL can load any table supported by Hive. **JSON** If you have JSON data with a consistent schema across records, Spark SQL can infer their schema and load this data as rows as well, making it very simple to pull out the fields you need. To load JSON data, first create a `HiveContext` as when using Hive. (No installation of Hive is needed in this case, though—that is, you don't need a *hive-site.xml* file.) Then use the `HiveContext.jsonFile` method to get an RDD of Row objects for the whole file. Apart from using the whole Row object, you can also register this RDD as a table and select specific fields from it. For example, suppose that we had a JSON file containing tweets in the format shown in Example 5-33, one per line.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [b] **wherein the data of a first data group has a different schema than the data of a second data group** and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and	**[18] Learning Spark:**

File Formats

Spark makes it very simple to load and save data in a large number of file formats. Formats range from unstructured, like text, to semistructured, like JSON, to structured, like SequenceFiles (see Table 5-1). The input formats that Spark wraps all transparently handle compressed formats based on the file extension.

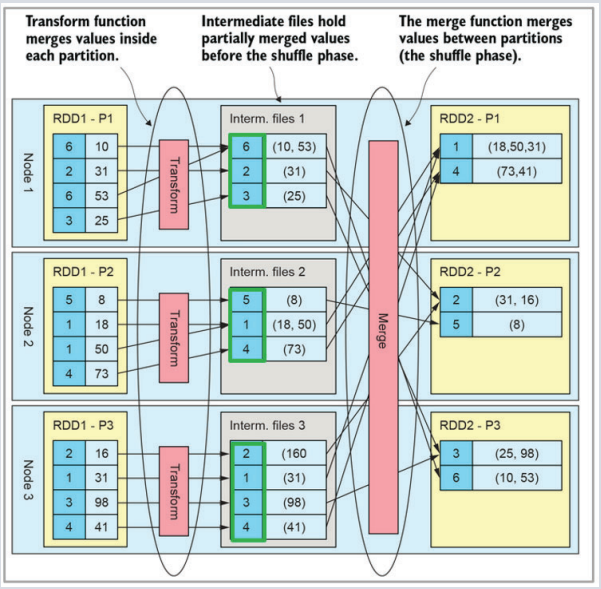
Table 5-1. Common supported file formats

Format name	Structured	Comments
Text files	No	Plain old text files. Records are assumed to be one per line.
JSON	Semi	Common text-based format, semistructured; most libraries require one record per line.
CSV	Yes	Very common text-based format, often used with spreadsheet applications.
SequenceFiles	Yes	A common Hadoop file format used for key/value data.
Protocol buffers	Yes	A fast, space-efficient multilanguage format.
Object files	Yes	Useful for saving data from a Spark job to be consumed by shared code. Breaks if you change your classes, as it relies on Java Serialization.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [b] wherein the data of a first data group has a different schema than the data of a second data group and **the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data,** wherein the different schema and corresponding different intermediate data have a key in common; and	**[13] Apache Spark FAQ at 1:** How does Spark relate to Apache Hadoop? Spark is a fast and general processing engine compatible with Hadoop data. It can run in Hadoop clusters through YARN or Spark's standalone mode, and it can process data in HDFS, HBase, Cassandra, Hive, and any Hadoop InputFormat. It is designed to perform both batch processing (similar to MapReduce) and new workloads like streaming, interactive queries, and machine learning. **[23] LanguageManual Transform at 1:** Transform/Map-Reduce Syntax Users can also plug in their own custom mappers and reducers in the data stream by using features natively supported in the Hive language. e.g. in order to run a custom mapper script - map_script - and a custom reducer script - reduce_script - the user can issue the following command which uses the TRANSFORM clause to embed the mapper and the reducer scripts.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [b] wherein the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, **wherein the different schema and corresponding different intermediate data have a key in common**; and	In the Accused Instrumentalities, the different schema and corresponding different intermediate data have a key in common. **[12] Spark in Action:**  Figure 4.2. A shuffle during the example `aggregateByKey` transformation on an RDD with three partitions. The transform function, passed to `aggregateByKey`, merges values in partitions. The merge function merges values between partitions during the shuffle phase. Intermediate files hold values merged per partition and are used during the shuffle phase.

Claim Chart – Claim 1

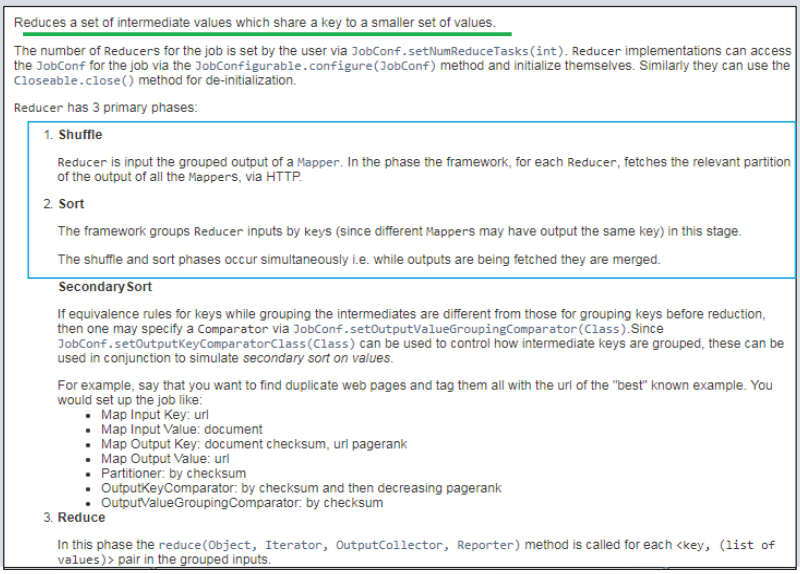
Claim Feature	Evidence

1 [b] wherein the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and	In another example, in Teradata Vantage, the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common. For example, Teradata Vantage enables merging of data of different schema. The particular details are in AA's possession, custody, and control.
	
	[26] Teradata Vantage User Guide
	
	`SELECT select_options`
	`FROM {`
	`  table_name [ [ AS ] correlation_name ] |`
	`  join |`
	`  ( subquery ) [ AS ] derived_table_name [ ( column_name_list ) ] |`
	`  table_function |`
	`  table_operator |`
	`  analytic_function [ AS ] correlation_name`
	`} [, ...]`
	`other_options`
	
	Syntax Elements
	
	select_options
	For information on the select options, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.
	table_name
	For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.
	correlation_name
	For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.
	subquery
	For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.
	derived_table_name
	For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.
	column_name_list
	For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.
	analytic_function
	The name of the Teradata ML Engine function to be run. For information on Teradata ML Engine functions, see *Teradata® Vantage Machine Learning Engine Analytic Function Reference*, B700-4003.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [c] **reducing the intermediate data for the data groups to at least one output data group**, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a merging of the corresponding different intermediate data based on the key in common,	The method performed by the Accused Instrumentalities include reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a merging of the corresponding different intermediate data based on the key in common. For example, the Reducer function reduces the intermediate values into a smaller set of values. This function operates in three phases: (1) Shuffle; (2) Sort; and (3) Reduce. The Shuffle and Sort phases operate on the output of various Mappers and group data before the final Reduce phase based on a common key. **[2] MapReduce Tutorial:** **Reducer** Reducer reduces a set of intermediate values which share a key to a smaller set of values. The number of reduces for the job is set by the user via Job.setNumReduceTasks(int). Overall, `Reducer` implementations are passed the `Job` for the job via the Job.setReducerClass(Class) method and can override it to initialize themselves. The framework then calls reduce(WritableComparable, Iterable<Writable>, Context) method for each `<key, (list of values)>` pair in the grouped inputs. Applications can then override the `cleanup(Context)` method to perform any required cleanup.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [c] reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a merging of the corresponding different intermediate data based on the key in common,	For example, the intermediate data is processed such that the corresponding different intermediate data is merged based on keys that are common to individual elements of the intermediate data. **[3] Interface Reducer:** 

Reduces a set of intermediate values which share a key to a smaller set of values.

The number of `Reducers` for the job is set by the user via `JobConf.setNumReduceTasks(int)`. `Reducer` implementations can access the `JobConf` for the job via the `JobConfigurable.configure(JobConf)` method and initialize themselves. Similarly they can use the `Closeable.close()` method for de-initialization.

`Reducer` has 3 primary phases:

1. **Shuffle**

   `Reducer` is input the grouped output of a `Mapper`. In the phase the framework, for each `Reducer`, fetches the relevant partition of the output of all the `Mappers`, via HTTP.

2. **Sort**

   The framework groups `Reducer` inputs by `keys` (since different `Mappers` may have output the same key) in this stage.

   The shuffle and sort phases occur simultaneously i.e. while outputs are being fetched they are merged.

   SecondarySort

   If equivalence rules for keys while grouping the intermediates are different from those for grouping keys before reduction, then one may specify a `Comparator` via `JobConf.setOutputValueGroupingComparator(Class)`.Since `JobConf.setOutputKeyComparatorClass(Class)` can be used to control how intermediate keys are grouped, these can be used in conjunction to simulate *secondary sort on values*.

   For example, say that you want to find duplicate web pages and tag them all with the url of the "best" known example. You would set up the job like:
   - Map Input Key: url
   - Map Input Value: document
   - Map Output Key: document checksum, url pagerank
   - Map Output Value: url
   - Partitioner: by checksum
   - OutputKeyComparator: by checksum and then decreasing pagerank
   - OutputValueGroupingComparator: by checksum

3. **Reduce**

   In this phase the `reduce(Object, Iterator, OutputCollector, Reporter)` method is called for each `<key, (list of values)>` pair in the grouped inputs.

Claim Chart – Claim 1

Claim Feature	Evidence

1 [c] reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a **merging of the corresponding different intermediate data based on the key in common,**	The Reduce function operates on different sets of input data (intermediate data) that have a key in common. From this processing, the Reduce function populates an output file.
	
	[3] Interface Reducer:
	
	reduce
	`void reduce(K2 key, Iterator<V2> values, OutputCollector<K3,V3> output, Reporter reporter) throws IOException`
	Reduces values for a given key.
	The framework calls this method for each `<key, (list of values)>` pair in the grouped inputs. Output values must be of the same type as input values. Input keys must not be altered. The framework will **reuse** the key and value objects that are passed into the reduce, therefore the application should clone the objects they want to keep a copy of. In many cases, all values are combined into zero or one value.
	Output pairs are collected with calls to `OutputCollector.collect(Object,Object)`.
	Applications can use the `Reporter` provided to report progress or just indicate that they are alive. In scenarios where the application takes a significant amount of time to process individual key/value pairs, this is crucial since the framework might assume that the task has timed-out and kill that task. The other way of avoiding this is to set mapreduce.task.timeout to a high-enough value (or even zero for no time-outs).
	Parameters:
	`key` - the key.
	`values` - the list of values to reduce.
	`output` - to collect keys and combined values.
	`reporter` - facility to report progress.
	Throws:
	`IOException`

Claim Chart – Claim 1

Claim Feature	Evidence
1 [c] reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a merging of the corresponding different intermediate data based on the key in common,	Further, in Spark, the intermediate data is reduced to an output data group. **[21] Spark SQL Shuffle Partitions:** **What is Spark Shuffle?** Shuffling is a mechanism Spark uses to redistribute the data across different executors and even across machines. Spark shuffling triggers for transformation operations like `gropByKey()`, `reducebyKey()`, `join()`, `groupBy()` e.t.c Spark Shuffle is an expensive operation since it involves the following • Disk I/O • Involves data serialization and deserialization • Network I/O When creating an RDD, Spark doesn't necessarily store the data for all keys in a partition since at the time of creation there is no way we can set the key for the data set. Hence, when we run the reduceByKey() operation to aggregate the data on keys, Spark does the following. • Spark first runs map tasks on all partitions which groups all values for a single key. • The results of the map tasks are kept in memory. • When results do not fit in memory, Spark stores the data on a disk. • Spark shuffles the mapped data across partitions, some times it also stores the shuffled data into a disk for reuse when it needs to recalculate. • Run the garbage collection • Finally runs reduce tasks on each partition based on key.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [c] **reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group,** so as to result in a merging of the corresponding different intermediate data based on the key in common,	The Accused Instrumentalities include reducing the intermediate data for the data groups to at least one output data group (e.g., a partition 1 in an RDD2). The intermediate data is processed in a manner that is defined to correspond to that data group (e.g., aggregateByKey transformation). The result is a merging of the corresponding intermediate data based on the key in common. **[12] Spark in Action:**

4.2.2. Understanding and avoiding unnecessary shuffling

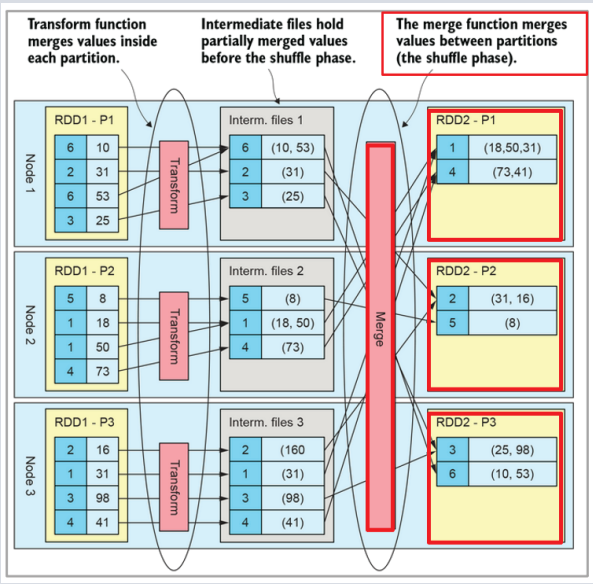
Physical movement of data between partitions is called *shuffling*. It occurs when data from multiple partitions needs to be combined in order to build partitions for a new RDD. When grouping elements by key, for example, Spark needs to examine all of the RDD's partitions, find elements with the same key, and then physically group them, thus forming new partitions.

To visualize what happens with partitions during a shuffle, we'll use the previous example with the `aggregateByKey` transformation (from section 4.1.2). The shuffle that occurs during that transformation is illustrated in figure 4.2.

The transform function puts all values of each key in a single partition (partitions P1 to P3) into a list. Spark then writes this data to intermediate files on each node. In the next phase, the merge function is called to merge lists from different partitions, but of the same key, into a single list for each key. The default partitioner (`HashPartitioner`) then kicks in and puts each key in its proper partition.

Tasks that immediately precede and follow the shuffle are called *map* and *reduce* tasks, respectively. The results of map tasks are written to intermediate files (often to the OS's filesystem cache only) and read by reduce tasks. In addition to being written to disk, the data is sent over the network, so it's important to try to minimize the number of shuffles during Spark jobs.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [c] **reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group**, so as to result in a merging of the corresponding different intermediate data based on the key in common,	[12] **Spark in Action:**  Figure 4.2. A shuffle during the example `aggregateByKey` transformation on an RDD with three partitions. The transform function, passed to `aggregateByKey`, merges values in partitions. The merge function merges values between partitions during the shuffle phase. Intermediate files hold values merged per partition and are used during the shuffle phase.

Claim Chart – Claim 1

Claim Feature	Evidence
1 [c] reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a merging of the corresponding different intermediate data based on the key in common,	In another example, Teradata Vantage includes reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group, so as to result in a merging of the corresponding different intermediate data based on the key in common. For example, Teradata Vantage enables merging of data of different schema. The particular details are in AA's possession, custody, and control. **[26] Teradata Vantage User Guide** **Usage** Syntax: kmeans@coprocessor(**ON**(...) **as** InputTable [**ON**(...) **as** CentroidsTable] [MaxIterNum('INTEGER')] [**Seed**('INTEGER')] [UnpackColumns('BOOLEAN')] [InitialSeeds('STRING')] [NumClusters('INTEGER')] [Threshold('DOUBLE')] [ClusteredOutput('TABLE_NAME')] OutputTable('TABLE_NAME')) **Input Columns:** <**Input Table**> should be **in** the **following format**< **id** ,point1 real ...pointn real> **Output Columns:** <**Output Table** will be **in** the **following format** <clusterid, packedColumn, **size**, withinss> **Function** Owner: db_superuser **Creation** Time: 2017-08-29 16:02:06.230091 **Function Version:** 7.0_rel_1.7_r65218 Interfaces Implemented: **Partition function**

Claim Chart – Claim 1

Claim Feature	Evidence
1 [d] wherein the mapping and reducing operations are performed by a **distributed system.**	The mapping and reducing operations are performed by each of the Accused Instrumentalities, which are a distributed system. **[1] An introduction to Apache Hadoop for big data:** **The Apache Hadoop framework is composed of the following modules** 1. Hadoop Common: contains libraries and utilities needed by other Hadoop modules 2. Hadoop Distributed File System (HDFS): a distributed file-system that stores data on the commodity machines, providing very high aggregate bandwidth across the cluster 3. Hadoop YARN: a resource-management platform responsible for managing compute resources in clusters and using them for scheduling of users' applications 4. Hadoop MapReduce: a programming model for large scale data processing All the modules in Hadoop are designed with a fundamental assumption that hardware failures (of individual machines, or racks of machines) are common and thus should be automatically handled in software by the framework. Apache Hadoop's MapReduce

Claim Chart – Claim 1

Claim Feature	Evidence
1 [d] wherein the mapping and reducing operations are performed by a **distributed system.**	The mapping and reducing operations are performed by each of the Accused Instrumentalities, which are a distributed system.

Comparison Table Between Hadoop and Teradata

Below are the lists of points, describe the Differences between Hadoop and Teradata :

Basis of Comparison	Teradata	Hadoop
Parallel Processing	Workload is divided across the system and evenly among the processors in the system.	Workload is divided among the different nodes on which relevant data is present and each node processes the task individually in parallel which reduces the overall time taken to complete the task.
Share-nothing Architecture	Teradata task executing in a virtual processor is independent of the tasks in other virtual processors.	Task execution on any node of the Hadoop is independent to tasks executing on other nodes.
Highly Scalable	More nodes/disks can be added but will increase the licensing cost.	More number of nodes/disks can be added as and when required to increase the processing and storage power.

[25] Hadoop vs Teradata

Claim 2

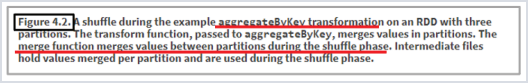
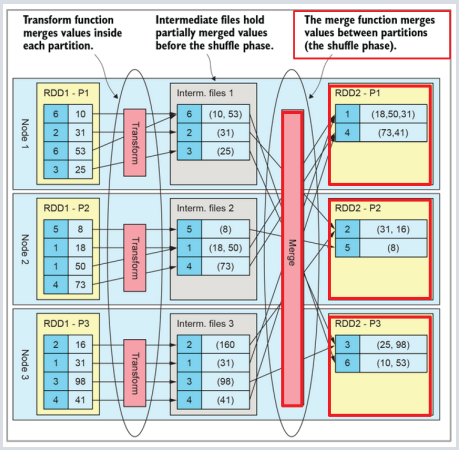
2[pre] The method of claim 1, wherein:

2[a] the at least one output data group is a plurality of output data groups.

Claim Chart – Claim 2

Claim Feature	Evidence
2 [pre] The method of claim 1, wherein:	AA infringes Claim 2 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 1.

Claim Chart – Claim 2

Claim Feature	Evidence
2 [a] the at least one output data group is a plurality of output data groups.	Additionally, in the method performed by the Accused Instrumentality, the at least one output data group is a plurality of output data groups. For example, in the illustration below, RDD2 – P1 is one of the at least one output data groups. In turn, RDD2 – P1 comprises a plurality of output data groups, 1 and 4. **[12] Spark Data Groups:**  Figure 4.2. A shuffle during the example aggregateByKey transformation on an RDD with three partitions. The transform function, passed to aggregateByKey, merges values in partitions. The merge function merges values between partitions during the shuffle phase. Intermediate files hold values merged per partition and are used during the shuffle phase. 

Claim Chart – Claim 2

Claim Feature	Evidence
2 [a] the at least one output data group is a plurality of output data groups.	In another example, in Teradata Vantage, the at least one output data group is a plurality of output data groups. The particular details are in AA's possession, custody, and control.

[26] Teradata Vantage User Guide

```
SELECT select_options
FROM {
   table_name [ [ AS ] correlation_name ] |
   join |
   ( subquery ) [ AS ] derived_table_name [ ( column_name_list ) ] |
   table_function |
   table_operator |
   analytic_function [ AS ] correlation_name
} [, ...]
other_options
```

Syntax Elements

select_options
For information on the select options, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

table_name
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

correlation_name
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

subquery
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

derived_table_name
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

column_name_list
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

analytic_function
The name of the Teradata ML Engine function to be run. For information on Teradata ML Engine functions, see *Teradata® Vantage Machine Learning Engine Analytic Function Reference*, B700-4003.

Claim 3

3[pre] The method of claim 1, wherein:

3[a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with a manner of storage is identifiable to the data group.

Claim Chart – Claim 3

Claim Feature	Evidence
3 [pre] The method of claim 1, wherein:	AA infringes Claim 3 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 1.

Claim Chart – Claim 3

Claim Feature	Evidence
3 [a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with a manner of storage is identifiable to the data group.	Additionally, in the method performed by the Accused Instrumentalities, corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with a manner of storage is identifiable to the data group. For example, intermediate data (e.g. records emitted from map functions) are stored such that metadata in association with the data that is identifiable to the data group. **[2] MapReduce Tutorial** **Mapper** Mapper maps input key/value pairs to a set of intermediate key/value pairs. Maps are the individual tasks that transform input records into intermediate records. The transformed intermediate records do not need to be of the same type as the input records. A given input pair may map to zero or many output pairs. **Map Parameters** A record emitted from a map will be serialized into a buffer and metadata will be stored into accounting buffers.

Claim Chart – Claim 3

Claim Feature	Evidence
3 [a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with a manner of storage is identifiable to the data group.	In another example, in Teradata Vantage, corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with a manner o storage is identifiable to the data group. The particular details are in AA's possession, custody, and control. 1. Contract Phase: Teradata SQL Engine receives the SQL request In the SQL Engine, the Parsing Engine (PE) performs the SQL parsing, and sends a metadata message through QueryGrid™ fabric to the Planner Parser in the ML Engine to discover metadata for function execution. In the metadata message, in addition to the basic information about function name, function version, function type and function descriptions, and so on, it contains parameters specifying the input and output tables, and the input columns. **[26] Teradata Vantage User Guide**

Claim 4

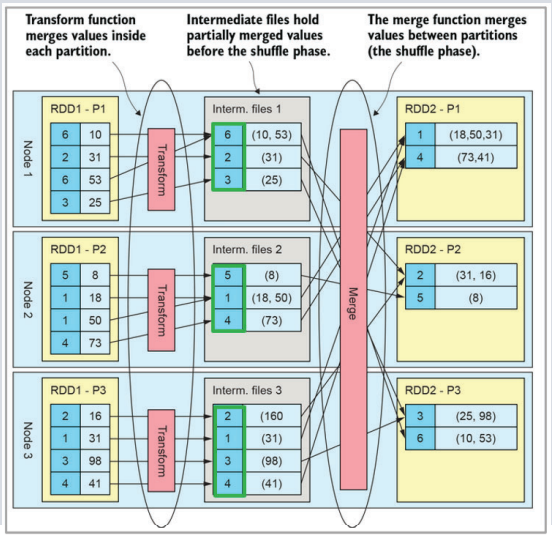
4[pre] The method of claim 1, wherein:

4[a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored in association with data that is identifiable to the data group.

Claim Chart – Claim 4

Claim Feature	Evidence
4 [pre] The method of claim 1, wherein:	AA infringes Claim 4 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 1.

Claim Chart – Claim 4

Claim Feature	Evidence
4 [a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored in association with data that is identifiable to the data group.	Additionally, in the method performed by the Accused Instrumentality, corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored in association with data that is identifiable to the data group. The intermediate data (e.g., intermediate files) are stored in association with the data that is identifiable to the data group (e.g., group identifier) **[12] Spark in Action** 

Claim Chart – Claim 4

Claim Feature	Evidence
4 **[a]** corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored in association with data that is identifiable to the data group.	**[18]** **Learning Spark** Spark provides special operations on RDDs containing key/value pairs. These RDDs are called pair RDDs. Pair RDDs are a useful building block in many programs, as they expose operations that allow you to act on each key in parallel or regroup data across the network. For example, pair RDDs have a `reduceByKey()` method that can aggregate data separately for each key, and a `join()` method that can merge two RDDs together by grouping elements with the same key. It is common to extract fields from an RDD (representing, for instance, an event time, customer ID, or other identifier) and use those fields as keys in pair RDD operations. **[12]** **Spark in Action** **4.1. WORKING WITH PAIR RDDS** Storing data as key-value pairs offers a simple, general, and extensible data model because each key-value pair can be stored independently and it's easy to add new types of keys and new types of values. This extensibility and simplicity have made this practice fundamental to several frameworks and applications. For example, many popular caching systems and NoSQL databases, such as memcached and Redis, are key-value stores. Hadoop's MapReduce also operates on key-value pairs (as you can see in appendix A). Keys and values can be simple types such as integers and strings, as well as more-complex data structures. Data structures traditionally used to represent key-value pairs are *associative arrays*, also called *dictionaries* in Python and *maps* in Scala and Java. In Spark, RDDs containing key-value tuples are called *pair RDDs*. Although you don't have to use data in Spark in the form of key-value pairs (as you've seen in the previous chapters), pair RDDs are well suited (and indispensable) for many use cases. Having keys along with the data enables you to aggregate, sort, and join the data, as you'll soon see. But before doing any of that, the first step is to create a pair RDD, of course.

Claim Chart – Claim 4

Claim Feature	Evidence
4 [a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored in association with data that is identifiable to the data group.	In another example, in Teradata Vantage, corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored in association with data that is identifiable to the data group. The particular details are in AA's possession, custody, and control.

[26] Teradata Vantage User Guide

```
SELECT select_options
FROM {
  table_name [ [ AS ] correlation_name ] |
  join |
  ( subquery ) [ AS ] derived_table_name [ ( column_name_list ) ] |
  table_function |
  table_operator |
  analytic_function [ AS ] correlation_name
} [, ...]
other_options
```

Syntax Elements

select_options
For information on the select options, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

table_name
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

correlation_name
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

subquery
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

derived_table_name
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

column_name_list
For information, see the description of SELECT statements in *Teradata® Database SQL Data Manipulation Language*.

analytic_function
The name of the Teradata ML Engine function to be run. For information on Teradata ML Engine functions, see *Teradata® Vantage Machine Learning Engine Analytic Function Reference*, B700-4003.

Claim 5

5[pre] The method of claim 1, wherein:

5[a] processing the intermediate data for each data group in a manner that is defined to correspond to that data group includes, for each data group, employing an iterator that corresponds to that data group.

Claim Chart – Claim 5

Claim Feature	Evidence
5 [pre] The method of claim 1, wherein:	AA infringes Claim 5 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 1.

Claim Chart – Claim 5

Claim Feature	Evidence
5 **[a]** processing the intermediate data for each data group in a manner that is defined to correspond to that data group includes, for each data group, employing an iterator that corresponds to that data group.	Additionally, in the method performed by the Accused Instrumentality, processing the intermediate data for each data group in a manner that is defined to correspond to the data group includes, for each data group, employing an iterator that corresponds to that data group. For example, the Accused Instrumentalities employs iterators that correspond to e.g., a particular RDD. The particular details are in AA's possession, custody, and control. **[12] Spark in Action** `zipPartitions` accepts two sets of arguments. In the first set, you give it RDDs; and in the second, a function that takes a matching number of `Iterator` objects used for accessing elements in each partition. The function must return a new `Iterator`, which can be a different type (matching the resulting RDD). This function has to take into account that RDDs may have different numbers of elements in partitions and guard against iterating beyond `Iterator` lengths.

Claim 17

17[pre] A computer system including a plurality of computing devices, the computer system configured to process data of a data set, wherein the data set comprises a plurality of data groups, the computer system comprises at least one processor and memory that are operable to perform the following operations:

17[a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,

17[b] wherein the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and

17[c] reduce the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group so as to result in a merging of the corresponding different intermediate data based on the key in common.

Claim Chart – Claim 17

Claim Feature	Evidence
17 [pre] A computer system including a plurality of computing devices, the computer system configured to process data of a data set, wherein the data set comprises a plurality of data groups, the computer system comprises at least one processor and memory that are operable to perform the following operations:	AA infringes Claim 17 of the '610 patent. The Accused Instrumentalities include a plurality of computing devices, the system configured to process data of a data set, wherein the data set comprises a plurality of data groups, the computer system comprises at least one processor and memory that are operable to perform operations described below and on the following slides. As discussed for Claim 1, the Accused Instrumentality is a distributed system (e.g. Hadoop Distributed File System, Teradata Vantage, etc.) that comprises multiple clusters (i.e., a plurality of computing devices). The Accused Instrumentality comprises a data set having a plurality of data groups. **[1] An introduction to Apache Hadoop for big data**

The Apache Hadoop framework is composed of the following modules

1. Hadoop Common: contains libraries and utilities needed by other Hadoop modules
2. Hadoop Distributed File System (HDFS): a distributed file-system that stores data on the commodity machines, providing very high aggregate bandwidth across the cluster
3. Hadoop YARN: a resource-management platform responsible for managing compute resources in clusters and using them for scheduling of users' applications
4. Hadoop MapReduce: a programming model for large scale data processing

All the modules in Hadoop are designed with a fundamental assumption that hardware failures (of individual machines, or racks of machines) are common and thus should be automatically handled in software by the framework. Apache Hadoop's MapReduce

Claim Chart – Claim 17

Claim Feature	Evidence
17 **[pre]** A computer system including a plurality of computing devices, the computer system configured to process data of a data set, wherein the data set comprises a plurality of data groups, the computer system comprises at least one processor and memory that are operable to perform the following operations:	For example, an RDD can be one of a plurality of groups within a data set. **[17] RDD Programming Guide** **Resilient Distributed Datasets (RDDs)** Spark revolves around the concept of a *resilient distributed dataset* (RDD), which is a fault-tolerant collection of elements that can be operated on in parallel. There are two ways to create RDDs: *parallelizing* an existing collection in your driver program, or referencing a dataset in an external storage system, such as a shared filesystem, HDFS, HBase, or any data source offering a Hadoop InputFormat. **Overview** At a high level, every Spark application consists of a *driver program* that runs the user's `main` function and executes various *parallel operations* on a cluster. The main abstraction Spark provides is a *resilient distributed dataset* (RDD), which is a collection of elements partitioned across the nodes of the cluster that can be operated on in parallel. RDDs are created by starting with a file in the Hadoop file system (or any other Hadoop-supported file system), or an existing Scala collection in the driver program, and transforming it. Users may also ask Spark to *persist* an RDD in memory, allowing it to be reused efficiently across parallel operations. Finally, RDDs automatically recover from node failures.

Claim Chart – Claim 17

Claim Feature	Evidence
17 [a] partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key-value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user-configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group,	The at least one processor and memory of the Accused Instrumentality are configured to perform the steps o partitioning the data of each one of the data groups into a plurality of data partitions that each have a plurality of key- value pairs and providing each data partition to a selected one of a plurality of mapping functions that are each user- configurable to independently output a plurality of lists of values for each of a set of keys found in such map function's corresponding data partition to form corresponding intermediate data for that data group and identifiable to that data group. The discussion for Claim element 1[a] is incorporated herein by reference.

Claim Chart – Claim 17

Claim Feature	Evidence
17 [b] wherein the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common; and	In the Accused Instrumentalities, the data of a first data group has a different schema than the data of a second data group and the data of the first data group is mapped differently than the data of the second data group so that different lists of values are output for the corresponding different intermediate data, wherein the different schema and corresponding different intermediate data have a key in common. The discussion for Claim element 1[b] is incorporated herein by reference.

Claim Chart – Claim 17

Claim Feature	Evidence
17 [c] reduce the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group so as to result in a merging of the corresponding different intermediate data based on the key in common.	The at least one processor and memory of the Accused Instrumentality are further configured to perform the steps o reducing the intermediate data for the data groups to at least one output data group, including processing the intermediate data for each data group in a manner that is defined to correspond to that data group so as to result in a merging of the corresponding different intermediate data based on the key in common. The discussion for Claim element 1[c] is incorporated herein by reference.

Claim 18

18[pre] The computer system of claim 17, wherein:

18[a] the at least one output data group is a plurality of output data groups.

Claim Chart – Claim 18

Claim Feature	Evidence
18 [pre] The computer system of claim 17, wherein:	AA infringes Claim 18 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 17.

Claim Chart – Claim 18

Claim Feature	Evidence
18 [a] the at least one output data group is a plurality of output data groups.	Additionally, in the Accused Instrumentality, the at least one output data group is a plurality of output data groups. The discussion for Claim 2 is incorporated herein by reference.

Claim 19

19[pre] The computer system of claim 17, wherein:

19[a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with the manner of storage is identifiable to the data group.

Claim Chart – Claim 19

Claim Feature	Evidence
19 [pre] The computer system of claim 17, wherein:	AA infringes Claim 19 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 17.

Claim Chart – Claim 19

Claim Feature	Evidence
19 [a] corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with the manner of storage is identifiable to the data group.	Additionally, in the Accused Instrumentality, corresponding intermediate data for a data group being identifiable to the data group includes the corresponding intermediate data being stored such that metadata associated with a manner o storage is identifiable to the data group. The discussion for Claim 3 is incorporated herein by reference.

Claim 20

20[pre] The computer system of claim 17, wherein:

20[a] corresponding intermediate data for a data group being identifiable to the data group includes the computer system being configured to store the corresponding intermediate data in association with data that is identifiable to the data group.

Claim Chart – Claim 20

Claim Feature	Evidence
20 [pre] The computer system of claim 19, wherein:	AA infringes Claim 20 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 17.

Claim Chart – Claim 20

Claim Feature	Evidence
20 [a] corresponding intermediate data for a data group being identifiable to the data group includes the computer system being configured to store the corresponding intermediate data in association with data that is identifiable to the data group	Additionally, in the Accused Instrumentality, corresponding intermediate data for a data group being identifiable to the data group includes the computer system being configured to store the corresponding intermediate data in association with data that is identifiable to the data group. The discussion for Claim 4 is incorporated herein by reference.

Claim 21

21[pre] The computer system of claim 17, wherein:

21[a] the at least one processor and memory being configured to process the intermediate data for each data group in a manner that is defined to correspond to that data group includes, for each data group, the at least one processor and memory being configured to employ an iterator that corresponds to that data group.

Claim Chart – Claim 21

Claim Feature	Evidence
21 [pre] The computer system of claim 17, wherein:	AA infringes Claim 21 of the '610 patent. R2 incorporates herein by reference its discussion above of how AA infringes each and every element of Claim 17.

Claim Chart – Claim 21

Claim Feature	Evidence
21 [a] the at least one processor and memory being configured to process the intermediate data for each data group in a manner that is defined to correspond to that data group includes, for each data group, the at least one processor and memory being configured to employ an iterator that corresponds to that data group.	Additionally, in the Accused Instrumentality, the at least one processor and memory being configured to process the intermediate data for each data group in a manner that is defined to correspond to the data group includes, for each data group, the at least one processor and memory being configured to employ an iterator that corresponds to that data group. The discussion for Claim 5 is incorporated herein by reference.

END

Exhibit B

Preliminary Infringement Contentions

R2 Solutions LLC v. American Airlines, Inc.
Civil Action No. 4:22-cv-00353-ALM

U.S. Pat. No.: 8,341,157

Claims: 1-2, 6, 11-14, 16, and 22

Preliminary Infringement Contentions

R2 Solutions LLC ("R2" or "Plaintiff") contends that American Airlines, Inc. ("AA" or "Defendant") has infringed and continues to infringe U.S. Pat. No. 8,341,157 (the "'157 Patent") by, among other things, making, having made, using, offering for sale, and/or selling within this judicial district and elsewhere in the United States and/or importing into this judicial district and into the United States – without license or authority – the "Accused Instrumentalities," which include systems, services, processes, methods, acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s). "Accused Instrumentalities" include, without limitation, all instrumentalities associated with receiving, handling, processing, responding to, or otherwise interacting with queries (including user queries and back-end queries) and all instrumentalities related to the location, identification, delivery, display, rendering, ranking, and communication of information relating to such queries. The "Accused Instrumentalities" include, without limitation, the front-end(s) and back-end(s) of AA's online travel agency and search engine services for hotels, vacation rentals, flights, and airport transfer incorporated into the AA web platform and/or AA mobile application. The Accused Instrumentalities practice the steps of the asserted claims or are capable of performing the claim elements of the asserted claims, or otherwise satisfy all elements, described or set forth in one or more claims of the '157 Patent in violation of at least 35 U.S.C. § 271(a). AA's infringements also include, and R2 is entitled to recover damages for, sales of infringing products and components sold and/or shipped abroad that are subsequently imported into the United States.

Preliminary Infringement Contentions

R2 contends that each and every limitation of the asserted claims of the '157 Patent is literally present or performed and infringed by each and every Accused Instrumentality. In the alternative, R2 contends that all limitations of the asserted claims are present under the doctrine of equivalents because (1) to the extent there are any differences between the Accused Instrumentalities and those claim limitations (which R2 denies), such differences are insubstantial. Equivalency may also be shown by the fact that (2) the Accused Instrumentalities perform substantially the same function, in substantially the same way, to achieve substantially the same result as the functions recited in the limitations of the asserted claims.

R2 reserves the right to supplement or amend its claim charts, preliminary infringement contentions, and expert reports as may be appropriate or permissible under the applicable rules and Scheduling Order, including the Court's modifications to P.R. 3-1 available in the Scheduling Order. Any reference or citation to the specification is for illustrative purposes only and is not intended to suggest that the scope of the claimed invention is limited to any particular preferred embodiment. R2 does not hereby waive any applicable work product protections.

For the avoidance of doubt, the claim elements charted herein are software limitations and/or associated with software limitations for the purposes of P.R. 3-1(g), and to the extent that such elements are performed by hardware, firmware, software, and/or any combination thereof, the particular details are in AA's possession, custody, and control.

References Cited

The following chart presents R2 Solutions' analysis of the Accused Instrumentalities based on publicly available information. The citations in the chart refer to the following publicly available documents, which are incorporated by reference as if fully set forth herein. This infringement analysis is preliminary, and R2's investigation into AA's infringement is ongoing. R2 reserves the right to provide additional theories under which AA products/systems infringe this patent:

[1] "**AA Vacations Homepage**," https://www.aavacations.com/en/.

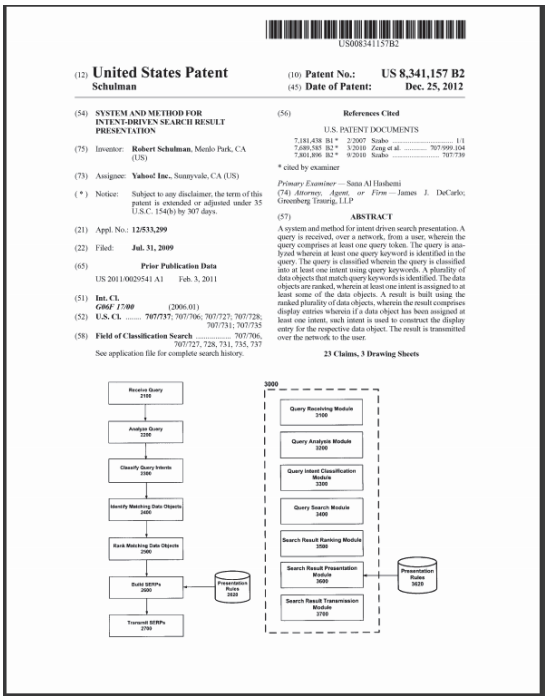
[2] "**DFW to LGA Search**," https://book.aavacations.com/apps/shopping/#/search-results/hotels/1riinvhawu3s7llzd6p/1.

[3] **"AA Vacations,"** https://www.aavacations.com/.

[4] "**AA Flights**," https://www.aa.com/homePage.do.

[5] "**DFW to LGA Flights**," https://www.aa.com/booking/flights/choose-flights/flight1?bookingPathStateId=1665678472499-826&redirectSearchToLegacyAACom=false.

U.S. Patent No. 8,341,157



US008341157B2

(12) **United States Patent**
Schulman

(10) **Patent No.: US 8,341,157 B2**
(45) **Date of Patent: Dec. 25, 2012**

(54) **SYSTEM AND METHOD FOR INTENT-DRIVEN SEARCH RESULT PRESENTATION**

(75) Inventor: **Robert Schulman**, Menlo Park, CA (US)

(73) Assignee: **Yahoo! Inc.**, Sunnyvale, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 307 days.

(21) Appl. No.: **12/533,299**

(22) Filed: **Jul. 31, 2009**

(65) **Prior Publication Data**

US 2011/0029541 A1 Feb. 3, 2011

(51) **Int. Cl.**
G06F 17/00 (2006.01)

(52) **U.S. Cl.** **707/737**; 707/706; 707/727; 707/728; 707/731; 707/735

(58) **Field of Classification Search** 707/706, 707/727, 728, 731, 735, 737
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

7,181,438 B1 * 2/2007 Szabo 1/1
7,689,585 B2 * 3/2010 Zeng et al. 707/999.104
7,801,896 B2 * 9/2010 Szabo 707/730

* cited by examiner

Primary Examiner — Sana Al Hashemi
(74) *Attorney, Agent, or Firm* — James J. DeCarlo; Greenberg Traurig, LLP

(57) **ABSTRACT**

A system and method for intent driven search presentation. A query is received, over a network, from a user, wherein the query comprises at least one query token. The query is analyzed wherein at least one query keyword is identified in the query. The query is classified wherein the query is classified into at least one intent using query keywords. A plurality of data objects that match query keywords is identified. The data objects are ranked, wherein at least one intent is assigned to at least some of the data objects. A result is built using the ranked plurality of data objects, wherein the result comprises display entries wherein if a data object has been assigned at least one intent, such intent is used to construct the display entry for the respective data object. The result is transmitted over the network to the user.

23 Claims, 3 Drawing Sheets

Bibliographic Details	
Title	SYSTEM AND METHOD FOR INTENT-DRIVEN SEARCH RESULT PRESENTATION
Priority Date	July 31, 2009
Filing Date	July 31, 2009
Issue Date	December 25, 2012
Expiry Date	June 3, 2030

(57) **ABSTRACT**

A system and method for intent driven search presentation. A query is received, over a network, from a user, wherein the query comprises at least one query token. The query is analyzed wherein at least one query keyword is identified in the query. The query is classified wherein the query is classified into at least one intent using query keywords. A plurality of data objects that match query keywords is identified. The data objects are ranked, wherein at least one intent is assigned to at least some of the data objects. A result is built using the ranked plurality of data objects, wherein the result comprises display entries wherein if a data object has been assigned at least one intent, such intent is used to construct the display entry for the respective data object. The result is transmitted over the network to the user.

Claim 1

1[pre] A method comprising:

1[a] receiving, over a network, a query from a user, the query comprising at least one query token;

1[b] analyzing the query, using at least one computing device, to identify at least one query keyword;

1[c] determining, at least the one computing device, a plurality of intents from the at least one keyword, each of the plurality of intents indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query;

1[d] classifying the query, using the at least one computing device, into at least one of the plurality of intents;

1[e] identifying, using the at least one computing device, a plurality of data objects available over the network that match the at least one query keyword;

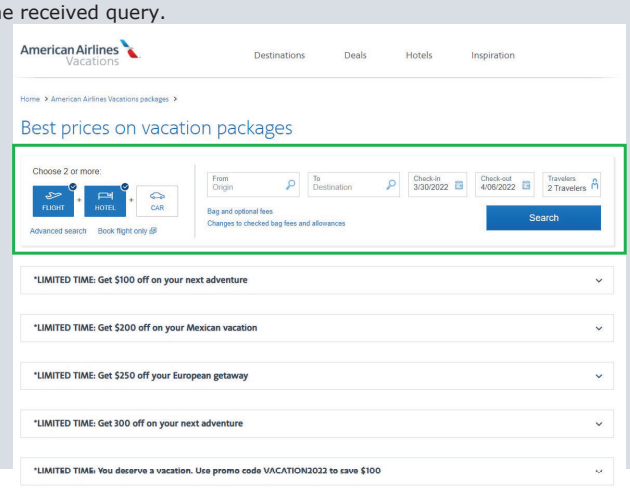
1[f] assigning, using the at least one computing device, at least one of the plurality of intents to at least some of the plurality of data objects;

1[g] ranking, using the at least one computing device, the plurality of data objects;

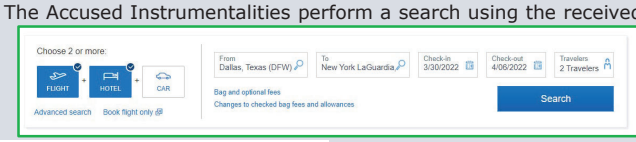
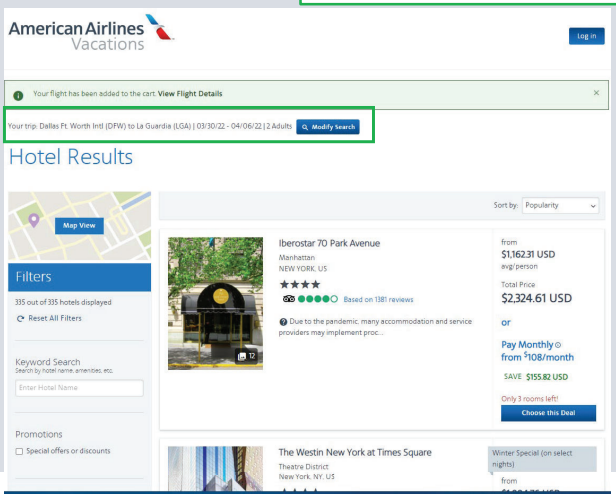
1[h] building a result, using the at least one computing device, using the ranked plurality of data objects, the result comprises a plurality of display entries, at least one display entry customized to a respective assigned intent is constructed for each of the ranked plurality of data objects; and

1[i] transmitting the result, over the network, to the user.

Claim Chart – Claim 1

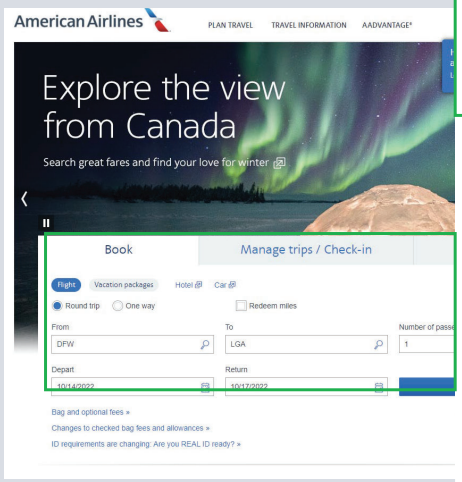
Claim Feature	Evidence
1[pre] A method comprising: **1[a]** receiving, over a network, a query from a user, the query comprising at least one query token;	AA infringes Claim 1 of the '157 Patent. The Accused Instrumentalities include systems, services, processes, methods acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s). The Accused Instrumentalities receive, over a network, a query from a user, the query comprising at least one query token. For example, the Accused Instrumentalities provide a search engine that allow users to submit search queries over a network (e.g., the Internet). The query can be generated by a user by typing, site visitation, sharing location information, etc. The Accused Instrumentalities perform a search using the received query. **[1] AA Vacations Homepage** 

Claim Chart – Claim 1

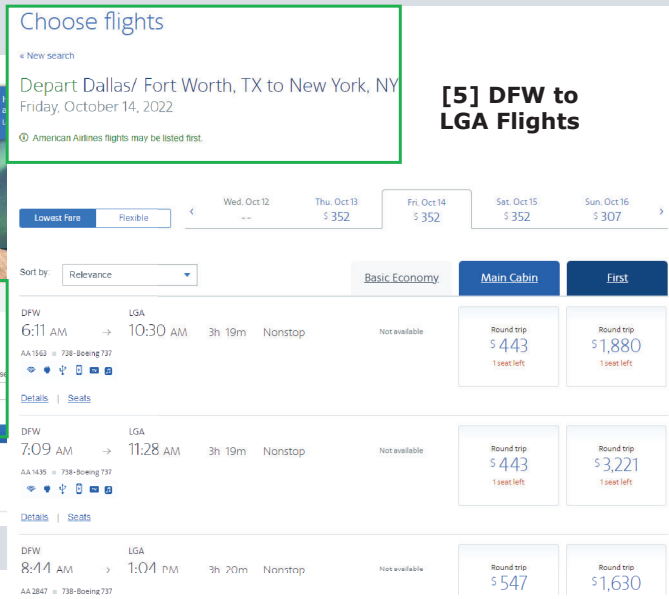
Claim Feature	Evidence
1[a] receiving, over a network, a query from a user, the query comprising at least one query token;	For example, the Accused Instrumentalities provide a search engine that allow users to submit search queries over a network (e.g., the Internet). The query can be generated by a user by typing, site visitation, sharing location information, clicking a link, selecting an option, etc. The Accused Instrumentalities perform a search using the received query. **[1] AA Vacations Homepage**  **[3] DFW to LGA Search** 

Claim Chart – Claim 1

Claim Feature	Evidence
1[a] receiving, over a network, a query from a user, the query comprising at least one query token;	For example, the Accused Instrumentalities provide a search engine that allow users to submit search queries over a network (e.g., the Internet). The query can be generated by a user by typing, site visitation, sharing location information, clicking a link, selecting an option, etc. The Accused Instrumentalities perform a search using the received query.

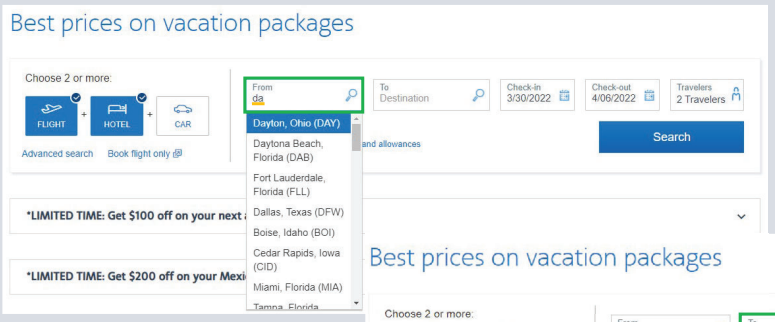
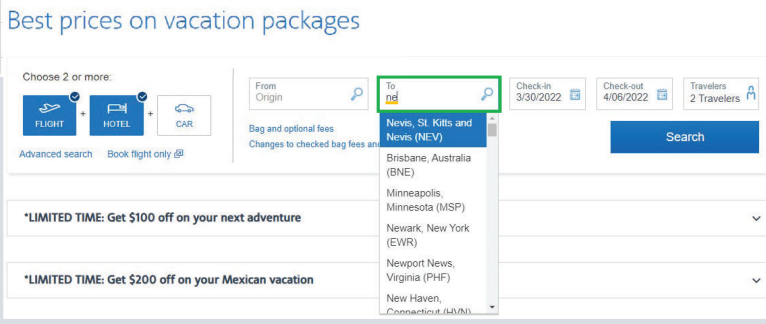


[4] AA Flights

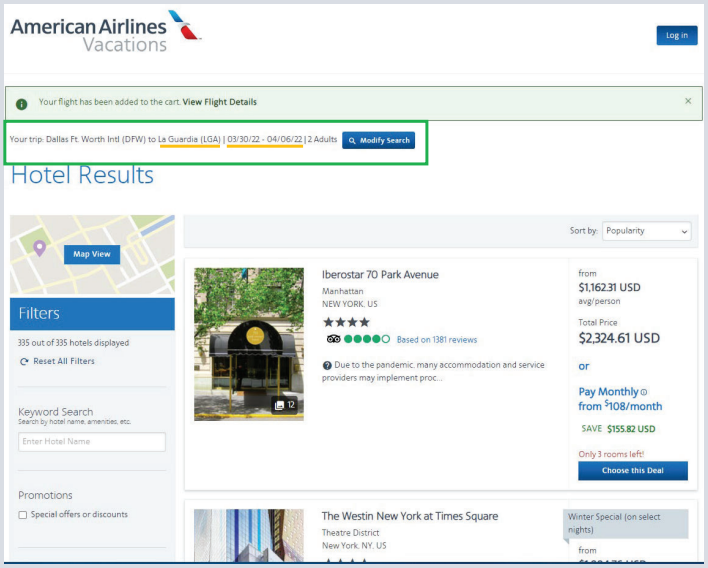


[5] DFW to LGA Flights

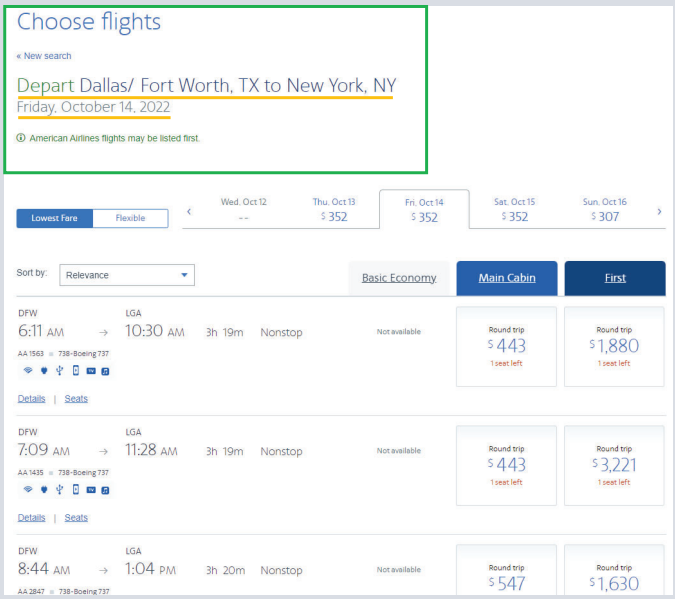
Claim Chart – Claim 1

Claim Feature	Evidence
1[a] receiving, over a network, a query from a user, the query comprising at least one query token;	The Accused Instrumentalities receive, over a network, a query from a user, the query comprising at least one query token. For example, a query can include a destination, location, property, etc., and/or a word and/or character related thereto:  **[1] AA Vacations Homepage** 

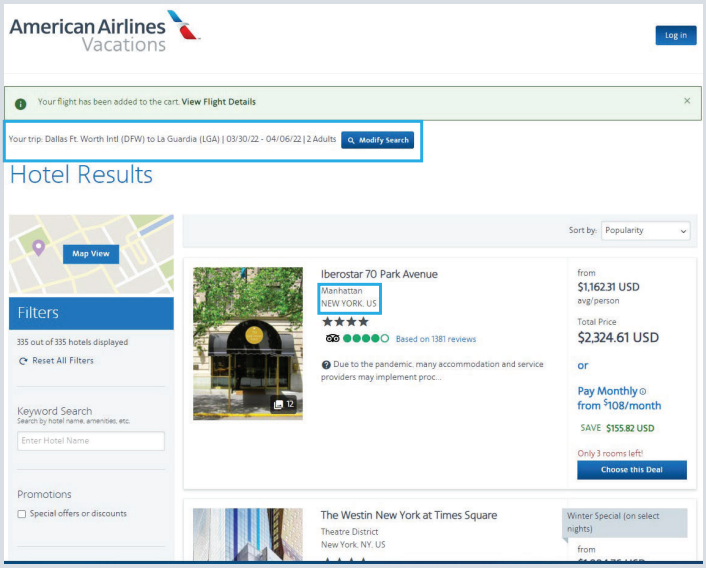
Claim Chart – Claim 1

Claim Feature	Evidence
1[a] receiving, over a network, a query from a user, the query comprising at least one query token;	The Accused Instrumentalities receive, over a network, a query from a user, the query comprising at least one query token. For example, a query can include a destination, location, property, etc., and/or a word and/or character related thereto: **[3] DFW to LGA Search** 

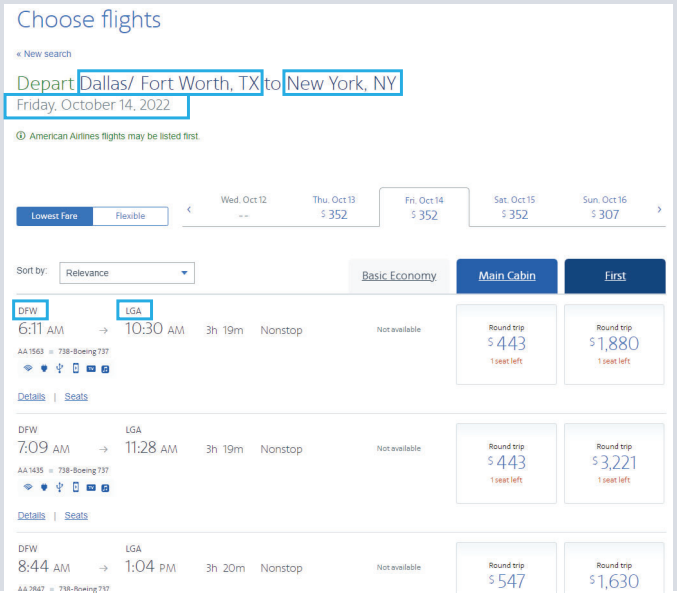
Claim Chart – Claim 1

Claim Feature	Evidence
1[a] receiving, over a network, a query from a user, the query comprising at least one query token;	For example, a query can include a starting point, destination, and/or date, and/or a word and/or character related thereto:  **[5] DFW to LGA Flights**

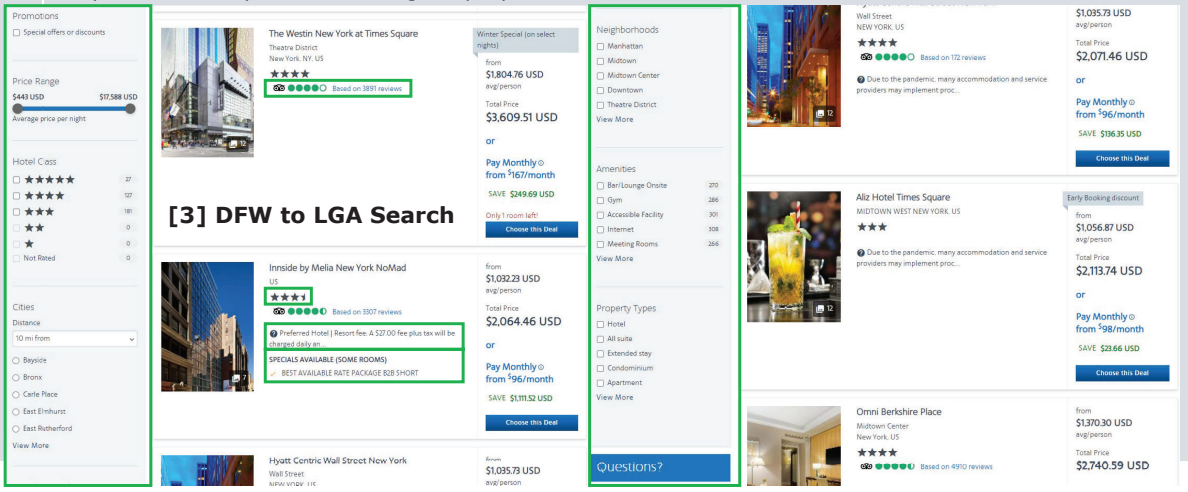
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] analyzing the query, using at least one computing device, to identify at least one query keyword;	The Accused Instrumentalities analyze the query, using at least one computing device, to identify at least one query keyword. For example, the Accused Instrumentalities can identify a query keyword (e.g., "New," "York," etc.) from the query (e.g., "LGA," etc.). The Accused Instrumentalities use at least one computing device to analyze the query: **[3] DFW to LGA Search** 

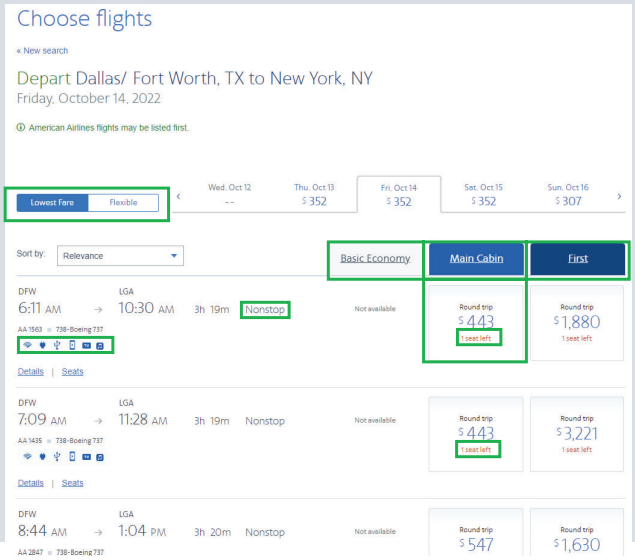
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] analyzing the query, using at least one computing device, to identify at least one query keyword;	For example, the Accused Instrumentalities can identify a query keyword (e.g., "New," "York," etc.) from the query (e.g., "LGA," etc.). The Accused Instrumentalities use at least one computing device to analyze the query:  **[5] DFW to LGA Flights**

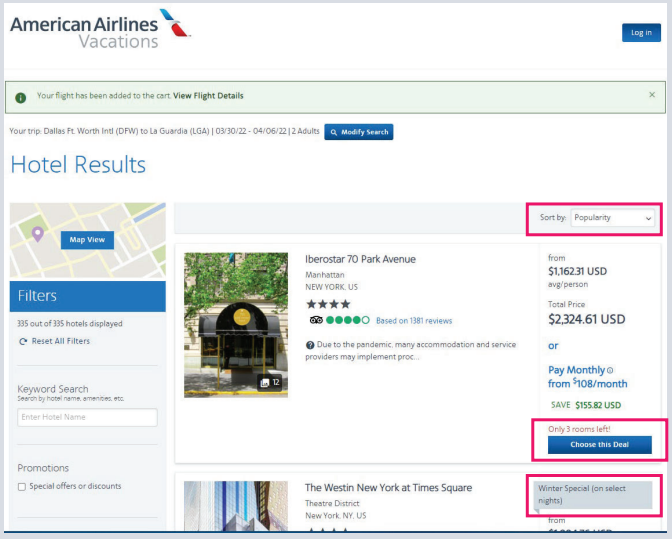
Claim Chart – Claim 1

Claim Feature	Evidence
1[c] determining, at least one computing device, a plurality of intents from the at least one keyword, each of the plurality of intents indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query;	The Accused Instrumentalities determine, using at least one computing device, a plurality of intents from the at least one keyword, wherein each of the plurality of intents indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query. For example, the Accused Instrumentalities determine multiple intents corresponding to the query keyword—a user may submit a query, and Accused Instrumentalities can determine that the user likely desires to see location-specific material, flights, hotels in a particular area, recommendations, deals, popular results, amenities, availability, etc. Each of the intents determined indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query: 

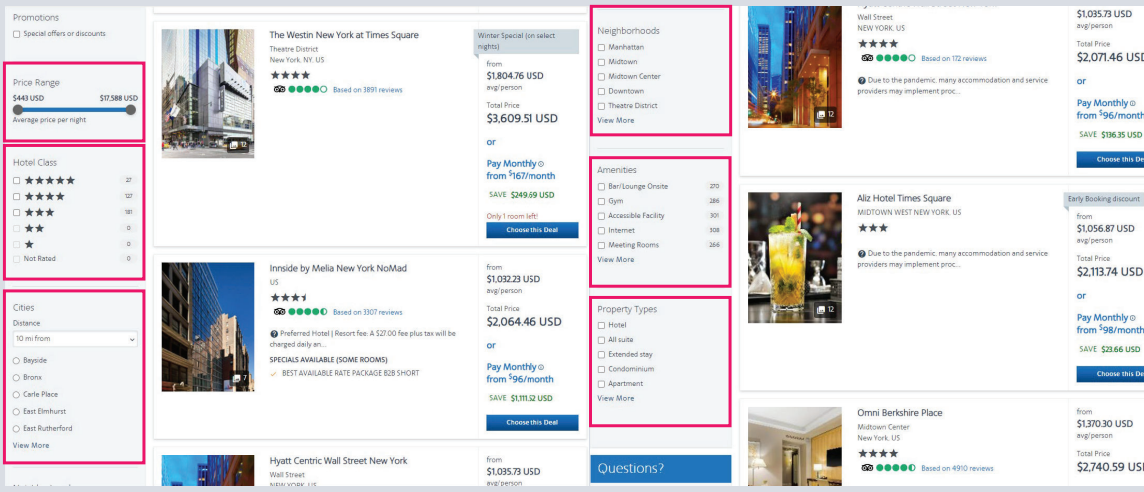
Claim Chart – Claim 1

Claim Feature	Evidence
1[c] determining, at least one computing device, a plurality of intents from the at least one keyword, each of the plurality of intents indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query;	For example, the Accused Instrumentalities determine multiple intents corresponding to the query keyword—a user may submit a query, and Accused Instrumentalities can determine that the user likely desires to see availability, lowest fare nonstop, cabin type, amenities, flexible, etc. Each of the intents determined indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query:  Choose flights « New search Depart Dallas/ Fort Worth, TX to New York, NY Friday, October 14, 2022 American Airlines flights may be listed first. Lowest Fare | Flexible Wed. Oct 12 -- | Thu. Oct 13 $352 | Fri. Oct 14 $352 | Sat. Oct 15 $352 | Sun. Oct 16 $307 Sort by: Relevance Basic Economy | Main Cabin | First DFW 6:11 AM → LGA 10:30 AM 3h 19m Nonstop Not available | Round trip $443 1 seat left | Round trip $1,880 1 seat left Details | Seats DFW 7:09 AM → LGA 11:28 AM 3h 19m Nonstop Not available | Round trip $443 1 seat left | Round trip $3,221 1 seat left Details | Seats DFW 8:44 AM → LGA 1:04 PM 3h 20m Nonstop Not available | Round trip $547 | Round trip $1,630 **[5] DFW to LGA Flights**

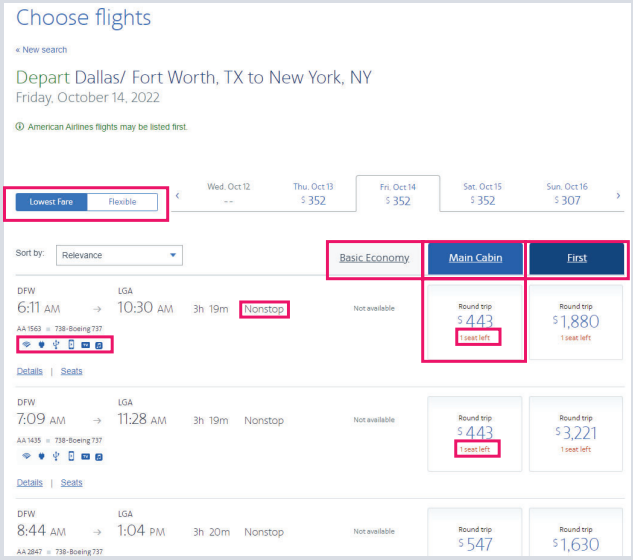
Claim Chart – Claim 1

Claim Feature	Evidence
1[d] classifying the query, using the at least one computing device, into at least one of the plurality of intents;	The Accused Instrumentalities classify the query, using the at least one computing device, into at least one of the plurality of intents. For example, when a query is submitted, Accused Instrumentalities classify the query into at least one intent from a plurality of intents. For example, a query can be classified into a "Popularity" and/or "deals" intent: **[3] DFW to LGA Search** 

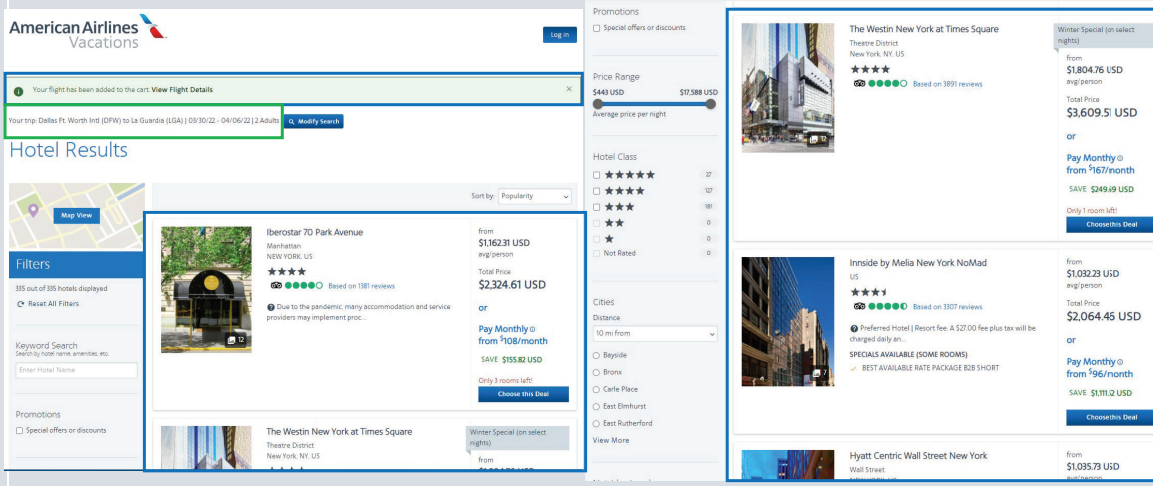
Claim Chart – Claim 1

Claim Feature	Evidence
1[d] classifying the query, using the at least one computing device, into at least one of the plurality of intents;	In another example, the Accused Instrumentalities classify the query into price, deals, class, city, availability amenities, etc. **[3] DFW to LGA Search** 

Claim Chart – Claim 1

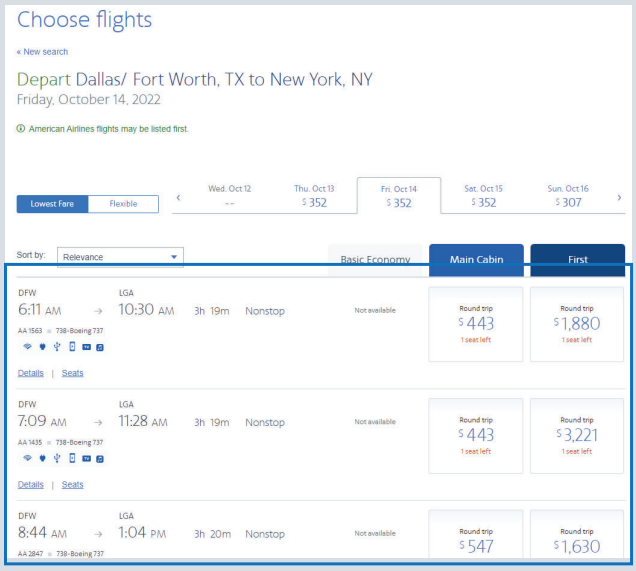
Claim Feature	Evidence
1[d] classifying the query, using the at least one computing device, into at least one of the plurality of intents;	In another example, the Accused Instrumentalities classify the query into price, flexible, availability, amenities, etc.  **[5] DFW to LGA Flights**

Claim Chart – Claim 1

Claim Feature	Evidence
1[e] identifying, using the at least one computing device, a plurality of data objects available over the network that match the at least one query keyword;	The Accused Instrumentalities identify, using the at least one computing device, a plurality of data objects available over the network that match the at least one query keyword. For example, the Accused Instrumentalities identify data objects (e.g., related to hotels, flights, properties, etc.) available over the network (e.g., Internet, etc.) that match the query keyword: **[3] DFW to LGA Search** 

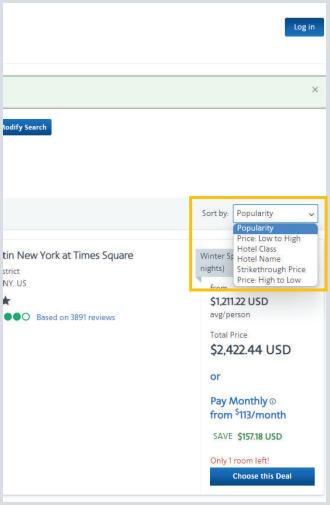
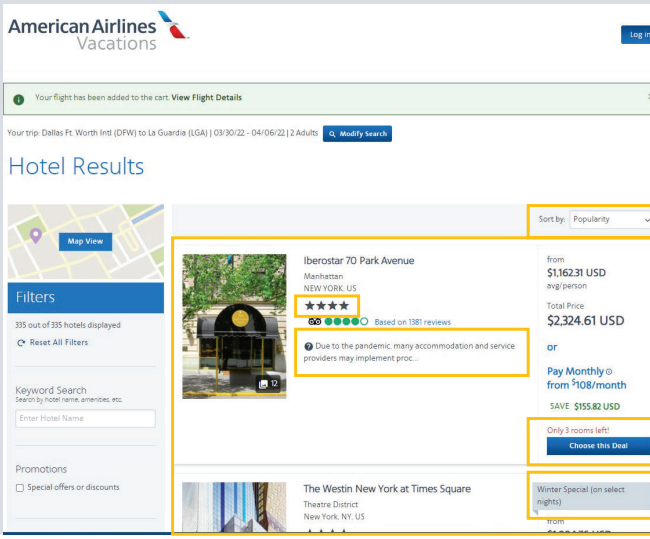
Claim Chart – Claim 1

Claim Feature	Evidence
1[e] identifying, using the at least one computing device, a plurality of data objects available over the network that match the at least one query keyword;	For example, the Accused Instrumentalities identify data objects (e.g., related to flights) available over the network (e.g., Internet, etc.) that match the query keyword:

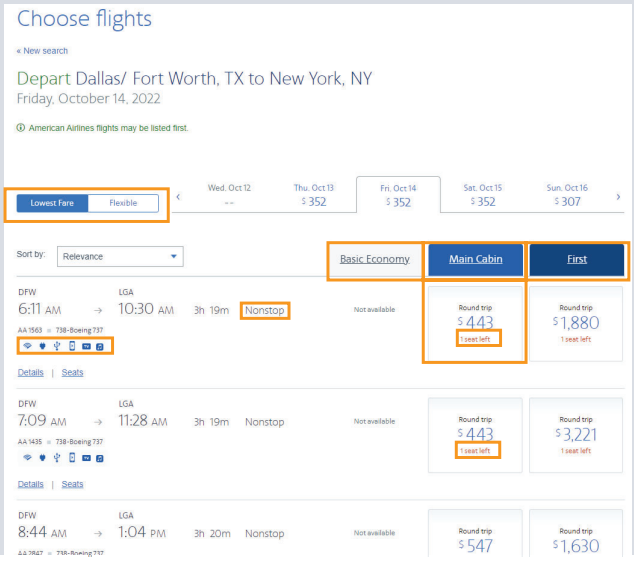


[5] DFW to LGA Flights

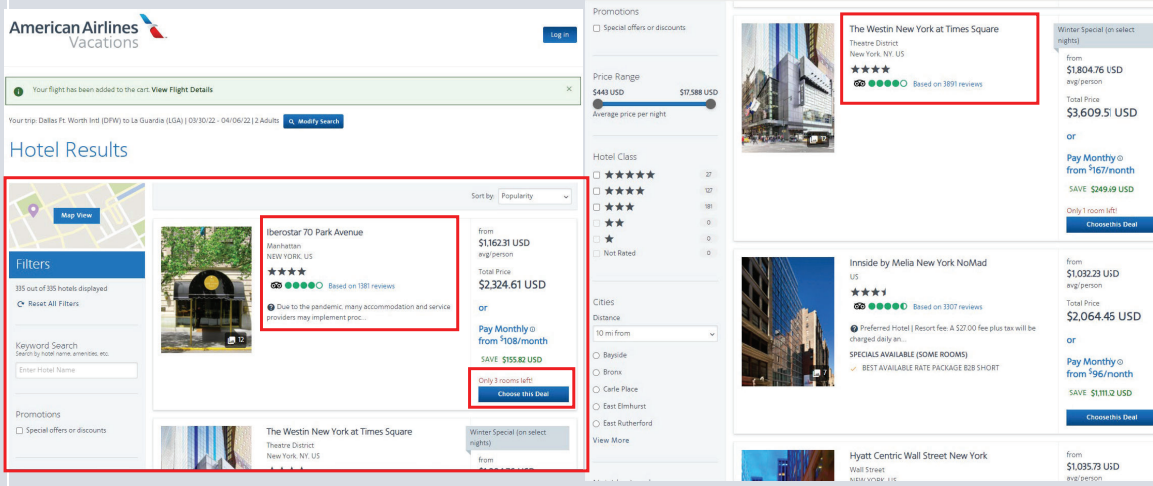
Claim Chart – Claim 1

Claim Feature	Evidence
1[f] assigning, using the at least one computing device, at least one of the plurality of intents to at least some of the plurality of data objects;	The Accused Instrumentalities assign, using the at least one computing device, at least one of the plurality of intents to at least some of the plurality of data objects. For example, the data objects are assigned intents such as deal status, rating, relevant protocols/procedures, Popularity, price, class, city, availability, amenities, etc. **[3] DFW to LGA Search**  

Claim Chart – Claim 1

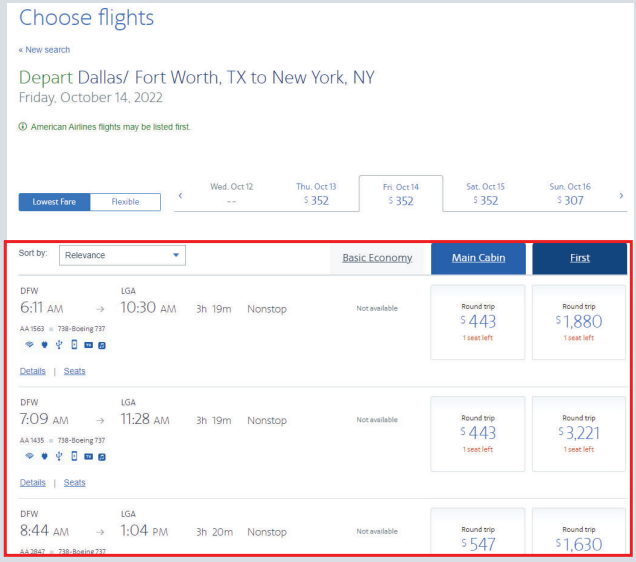
Claim Feature	Evidence
1[f] assigning, using the at least one computing device, at least one of the plurality of intents to at least some of the plurality of data objects;	For example, the data objects are assigned intents such as amenities, availability, etc.  **[5] DFW to LGA Flights**

Claim Chart – Claim 1

Claim Feature	Evidence
1[g] ranking, using the at least one computing device, the plurality of data objects;	The Accused Instrumentalities rank, using the at least one computing device, the plurality of data objects. For example the Accused Instrumentalities rank the data objects by, e.g., price, reviews, popularity, availability, popularity, etc.: **[3] DFW to LGA Search** 

Claim Chart – Claim 1

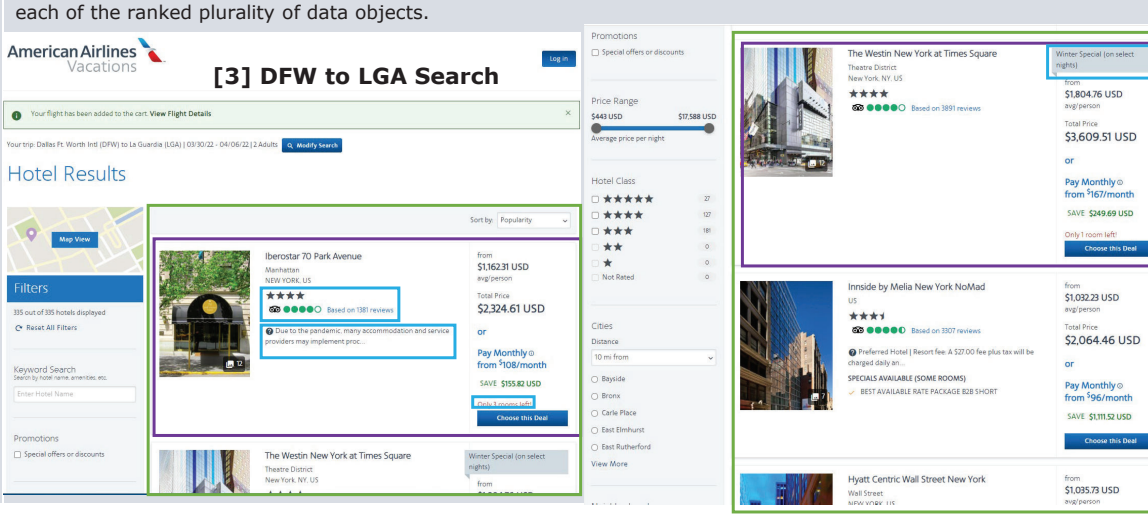
Claim Feature	Evidence
1[g] ranking, using the at least one computing device, the plurality of data objects;	For example, the Accused Instrumentalities rank the data objects by, e.g., price, time, availability, etc.:



[5] DFW to LGA Flights

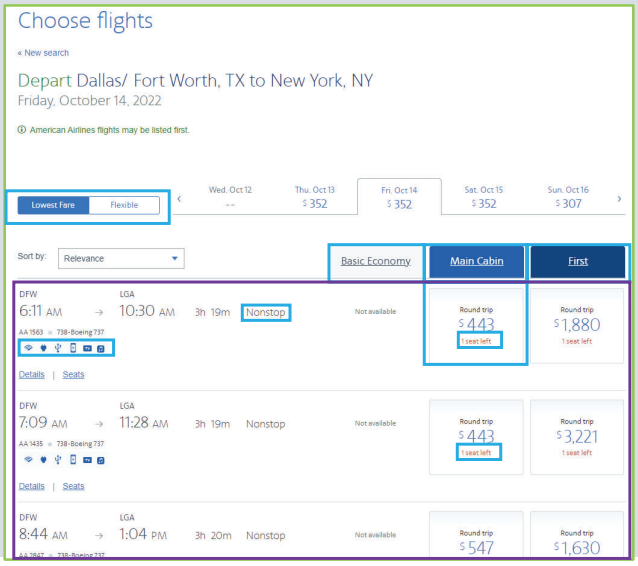
Claim Chart – Claim 1

Claim Feature	Evidence
1[h] building a result, using the at least one computing device, using the ranked plurality of data objects, the result comprises a plurality of display entries, at least one display entry customized to a respective assigned intent is constructed for each of the ranked plurality of data objects; and	The Accused Instrumentalities build a result, using the at least one computing device, using the ranked plurality of data objects, wherein the result comprises a plurality of display entries, wherein at least one display entry customized to a respective assigned intent is constructed for each of the ranked plurality of data objects. For example, the Accused Instrumentalities build a result comprising a plurality of display entries (e.g., sections of the result display), and such display entries are customized for a particular assigned intent (e.g., availability, deal specifics, etc.) and constructed for each of the ranked plurality of data objects.



Claim Chart – Claim 1

Claim Feature	Evidence
1[h] building a result, using the at least one computing device, using the ranked plurality of data objects, the result comprises a plurality of display entries, at least one display entry customized to a respective assigned intent is constructed for each of the ranked plurality of data objects; and	For example, the Accused Instrumentalities build a result comprising a plurality of display entries (e.g., sections of the result display), and such display entries are customized for a particular assigned intent (e.g., availability, amenities etc.) and constructed for each of the ranked plurality of data objects.

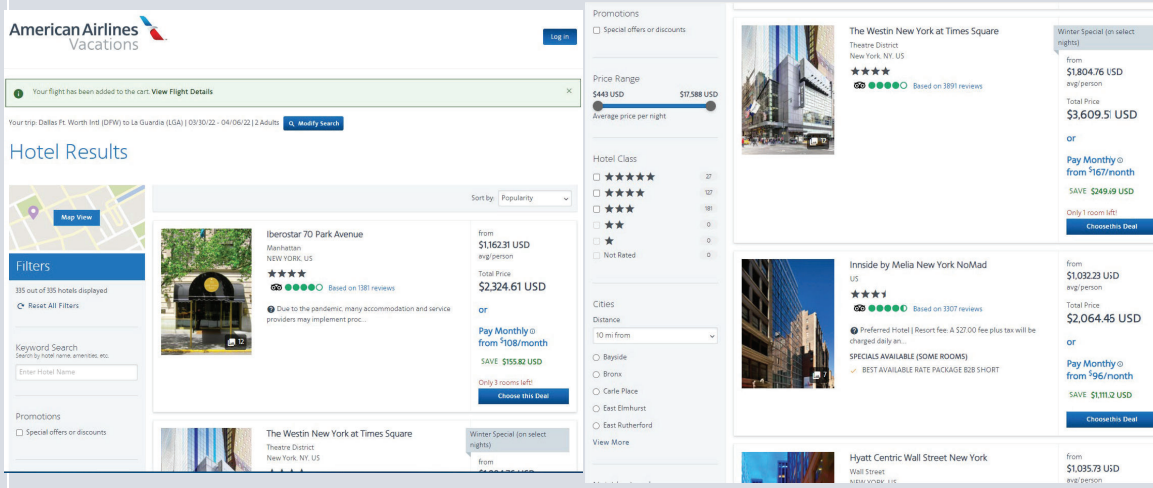


[5] DFW to LGA Flights

Claim Chart – Claim 1

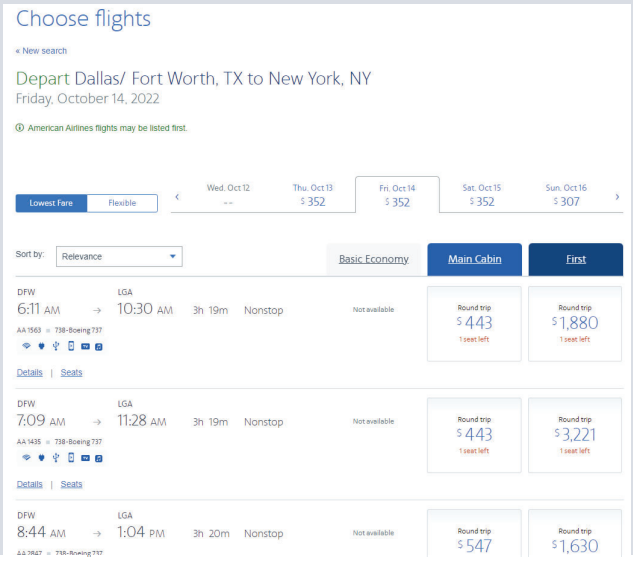
Claim Feature	Evidence
1[i] transmitting the result, over the network, to the user.	The Accused Instrumentalities transmit the result, over the network, to the user. For example, once the result is built, i is transmitted to a user over the network (e.g., the Internet).

[3] DFW to LGA Search



Claim Chart – Claim 1

Claim Feature	Evidence
1[i] transmitting the result, over the network, to the user.	The Accused Instrumentalities transmit the result, over the network, to the user. For example, once the result is built, i is transmitted to a user over the network (e.g., the Internet).



[5] DFW to LGA Flights

Claim 2

2[p] A computer-readable medium having computer-executable instructions for a method comprising the steps of:

2[a] receiving, over a network, a query from a user, wherein the query comprising at least one query token;

2[b] analyzing the query, using at least one computing device, to identify at least one query keyword;

2[c] determining a plurality of intents, by the computing device, from the at least one keyword, each of the plurality of intents indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query;

2[d] classifying the query, using the at least one computing device, into at least a subset of the plurality of intents;

2[e] identifying, using the at least one computing device, a plurality of data objects available over the network that match the at least one query keyword;

2[f] assigning, using the at least one computing device, at least one of the plurality of intents to at least some of the plurality of data objects;

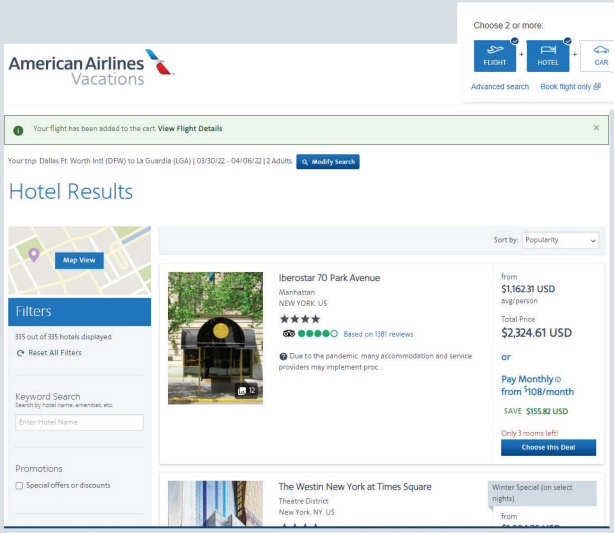
2[g] ranking, using the at least one computing device, the plurality of data objects;

2[h] building a result, using the at least one computing device, using the ranked plurality of data objects, the result comprises a plurality of display entries, wherein at least one display entry customized to a respective assigned intent is constructed for each of the ranked plurality of data objects; and

2[i] transmitting the result, over the network, to the user.

Claim Chart – Claim 2

Claim Feature	Evidence
2[pre] A computer-readable medium having computer-executable instructions for a method comprising the steps of:	AA infringes Claim 2 of the '157 Patent. The Accused Instrumentalities include a computer-readable medium having computer-executable instructions for a method comprising the steps of Claim 2. The Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s).

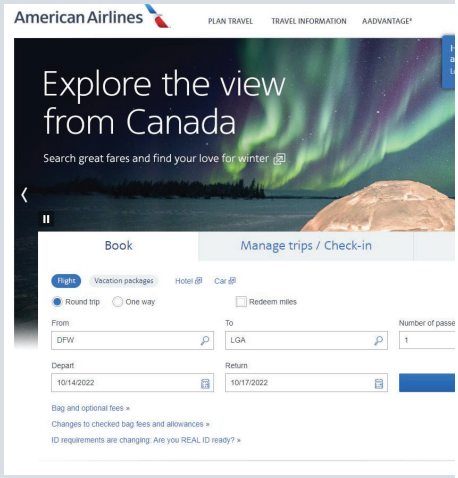


[1] AA Vacations Homepage

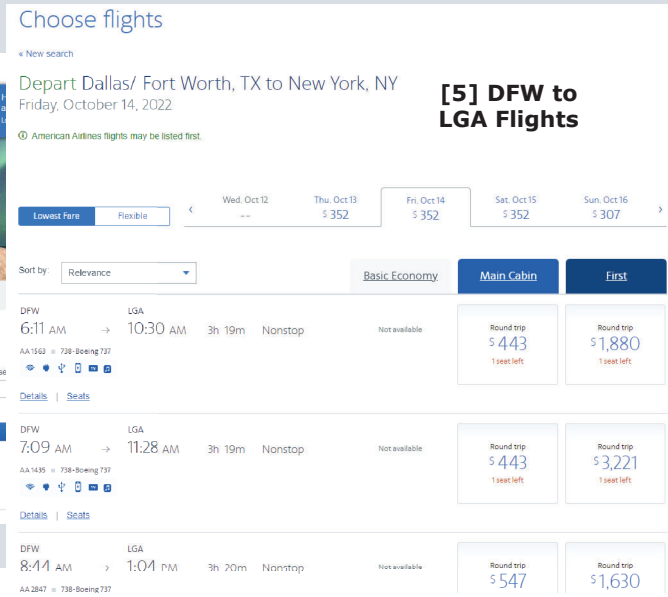
[3] DFW to LGA Search

Claim Chart – Claim 2

Claim Feature	Evidence
2[pre] A computer-readable medium having computer-executable instructions for a method comprising the steps of:	AA The Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/o software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s).



[4] AA Flights



[5] DFW to LGA Flights

Claim Chart – Claim 2

Claim Feature	Evidence
2[a] receiving, over a network, a query from a user, wherein the query comprising at least one query token;	The Accused Instrumentalities receive, over a network, a query from a user, the query comprising at least one query token. The discussion of Claim element 1[a] is incorporated herein by reference.

Claim Chart – Claim 2

Claim Feature	Evidence
2[b] analyzing the query, using at least one computing device, to identify at least one query keyword;	The Accused Instrumentalities analyze the query, using at least one computing device, to identify at least one query keyword. The discussion of Claim element 1[b] is incorporated herein by reference.

Claim Chart – Claim 2

Claim Feature	Evidence
2[c] determining a plurality of intents, by the computing device, from the at least one keyword, each of the plurality of intents indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query;	The Accused Instrumentalities determine a plurality of intents, by the computing device, from the at least one keyword each of the plurality of intents indicates a type of information regarding the query keyword that is likely to be desired by a user submitting the query. The discussion of Claim element 1[c] is incorporated herein by reference.

Claim Chart – Claim 2

Claim Feature	Evidence
2[d] classifying the query, using the at least one computing device, into at least a subset of the plurality of intents;	The Accused Instrumentalities classify the query, using the at least one computing device, into at least a subset of the plurality of intents. The discussion of Claim element 1[d] is incorporated herein by reference.

Claim Chart – Claim 2

Claim Feature	Evidence
2[e] identifying, using the at least one computing device, a plurality of data objects available over the network that match the at least one query keyword;	The Accused Instrumentalities identify, using the at least one computing device, a plurality of data objects available over the network that match the at least one query keyword. The discussion of Claim element 1[e] is incorporated herein by reference.

Claim Chart – Claim 2

Claim Feature	Evidence
2[f] assigning, using the at least one computing device, at least one of the plurality of intents to at least some of the plurality of data objects;	The Accused Instrumentalities assign, using the at least one computing device, at least one of the plurality of intents to at least some of the plurality of data objects. The discussion of Claim element 1[f] is incorporated herein by reference.

Claim Chart – Claim 2

Claim Feature	Evidence
2[g] ranking, using the at least one computing device, the plurality of data objects;	The Accused Instrumentalities rank, using the at least one computing device, the plurality of data objects. The discussion of Claim element 1[g] is incorporated herein by reference.

Claim Chart – Claim 2

Claim Feature	Evidence
2[h] building a result, using the at least one computing device, using the ranked plurality of data objects, the result comprises a plurality of display entries, wherein at least one display entry customized to a respective assigned intent is constructed for each of the ranked plurality of data objects; and	The Accused Instrumentalities build a result, using the at least one computing device, using the ranked plurality of data objects, the result comprises a plurality of display entries, wherein at least one display entry customized to a respective assigned intent is constructed for each of the ranked plurality of data objects. The discussion of Claim element 1[h] is incorporated herein by reference.

Claim Chart – Claim 2

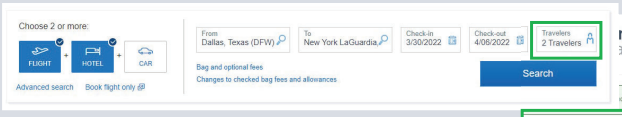
Claim Feature	Evidence
2[i] transmitting the result, over the network, to the user.	The Accused Instrumentalities transmit the result, over the network, to the user. The discussion of Claim element 1[i] is incorporated herein by reference.

Claim 6

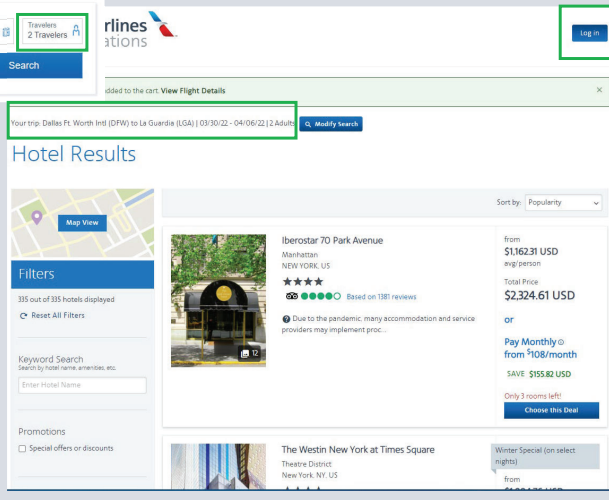
6 The method of claim 1 wherein, in the classifying step, profile information for the user is additionally used to classify the query.

Claim Chart – Claim 6

Claim Feature	Evidence
6 The method of claim 1 wherein, in the classifying step, profile information for the user is additionally used to classify the query.	AA infringes Claim 6 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, in the classifying step, profile information for the user is additionally used to classify the query. For example, user-specific profile information (e.g., shipper information etc.) are used to classify the query, and the particular details are within AA's possession, custody, and control.



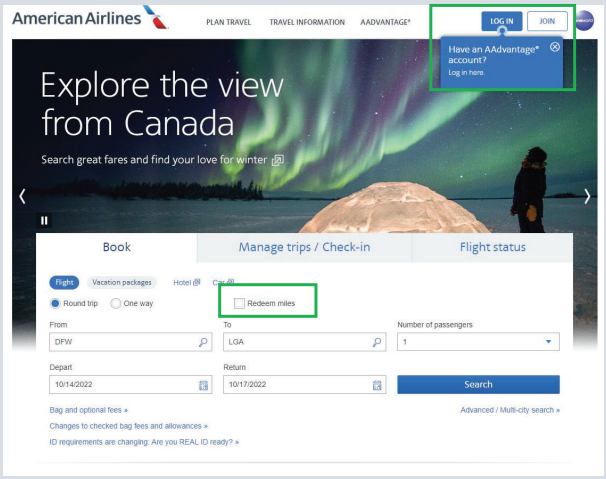
[1] AA Vacations Homepage



[3] DFW to LGA Search

Claim Chart – Claim 6

Claim Feature	Evidence
6 The method of claim 1 wherein, in the classifying step, profile information for the user is additionally used to classify the query.	AA infringes Claim 6 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, in the classifying step, profile information for the user is additionally used to classify the query. For example, user-specific profile information (e.g., member number miles, etc.) are used to classify the query, and the particular details are within AA's possession, custody, and control.



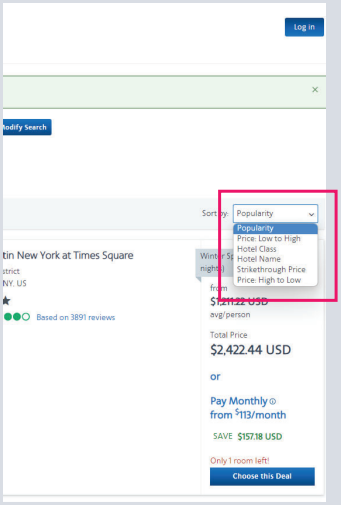
[4] AA Flights

Claim 11

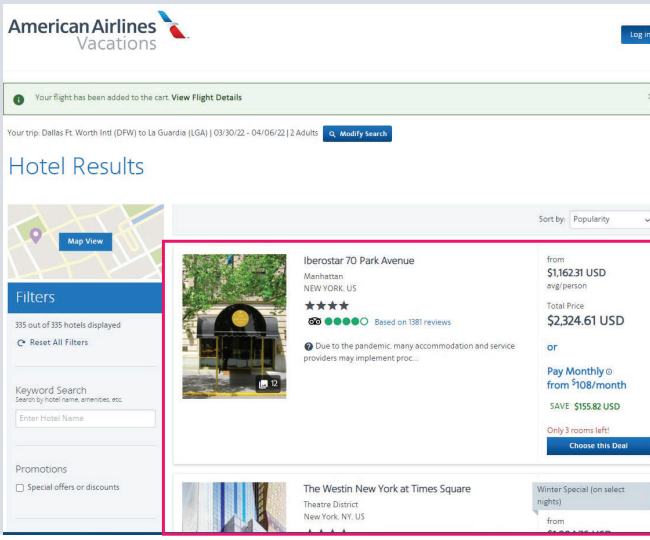
11 The method of claim 1 wherein the plurality of data objects are ranked by intent.

Claim Chart – Claim 11

Claim Feature	Evidence
11 The method of claim 1 wherein the plurality of data objects are ranked by intent.	AA infringes Claim 11 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, the plurality of data objects are ranked by intent. For example, the data objects are ranked by, e.g., popularity, price, etc., and the particular details are within AA's possession, custody, and control.

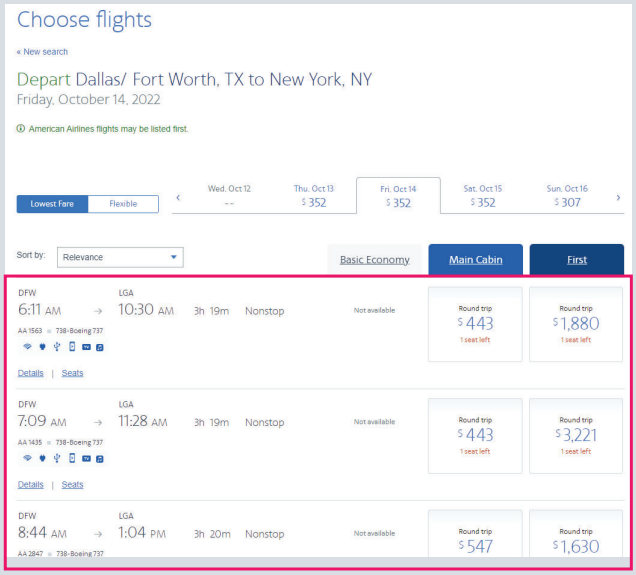


[3] DFW to LGA Search



Claim Chart – Claim 11

Claim Feature	Evidence
11 The method of claim 1 wherein the plurality of data objects are ranked by intent.	For example, the data objects are ranked by, e.g., time, price, etc., and the particular details are within AA's possession, custody, and control.



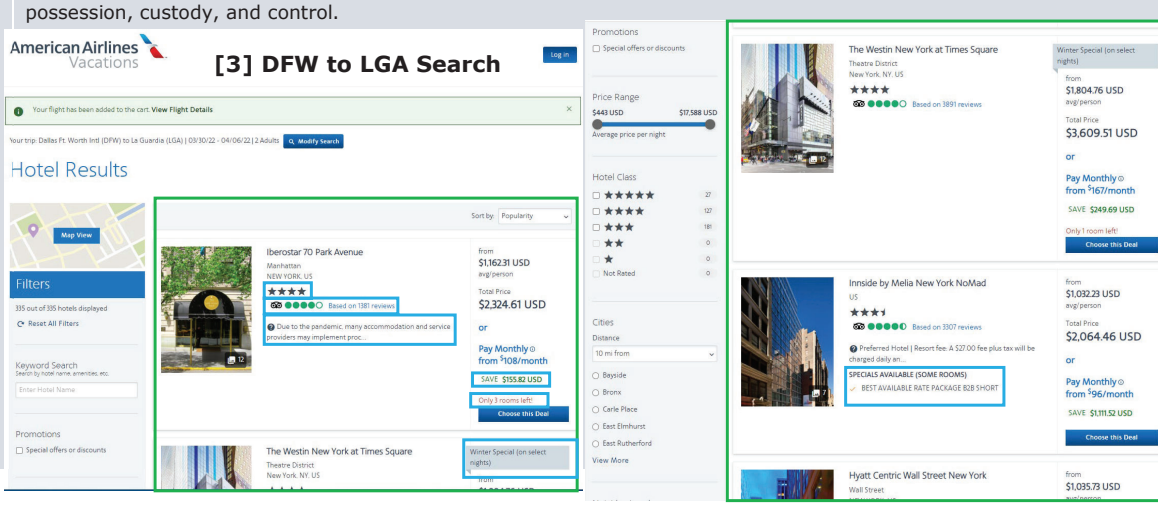
[5] DFW to LGA Flights

Claim 12

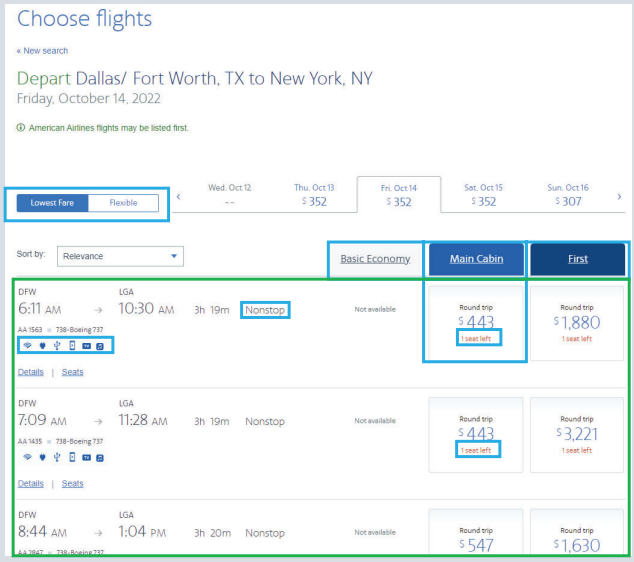
12 The method of claim 1 wherein, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object.

Claim Chart – Claim 12

Claim Feature	Evidence
12 The method of claim 1 wherein, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object.	AA infringes Claim 12 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object. For example, a data object is matched with a presentation rule to construct a display entry that can include ratings, deal indications, price cuts, etc., and the particular details are within AA's possession, custody, and control.



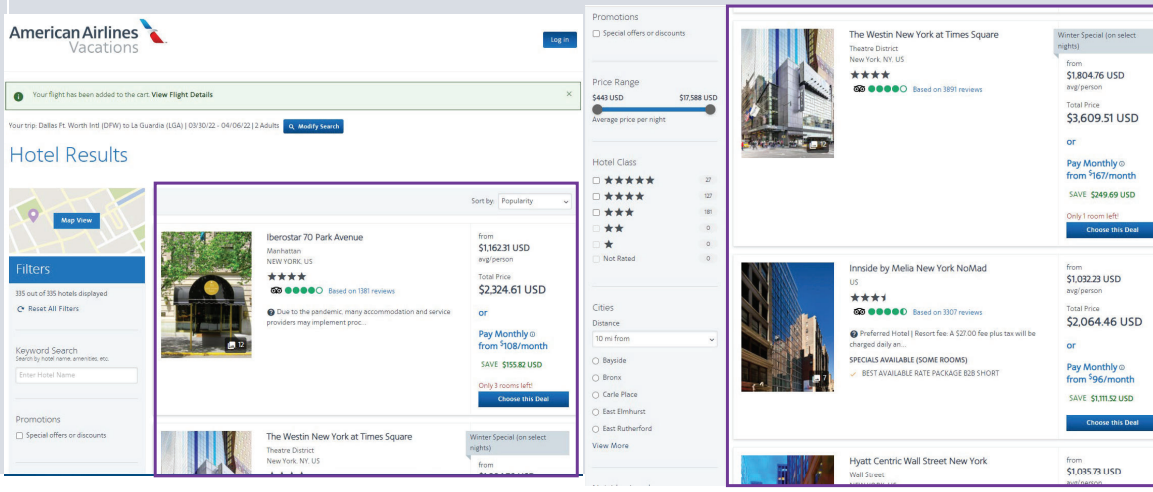
Claim Chart – Claim 12

Claim Feature	Evidence
12 The method of claim 1 wherein, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object.	For example, a data object is matched with a presentation rule to construct a display entry that can include availability amenities, etc., and the particular details are within AA's possession, custody, and control.  **[5] DFW to LGA Flights**

Claim 13

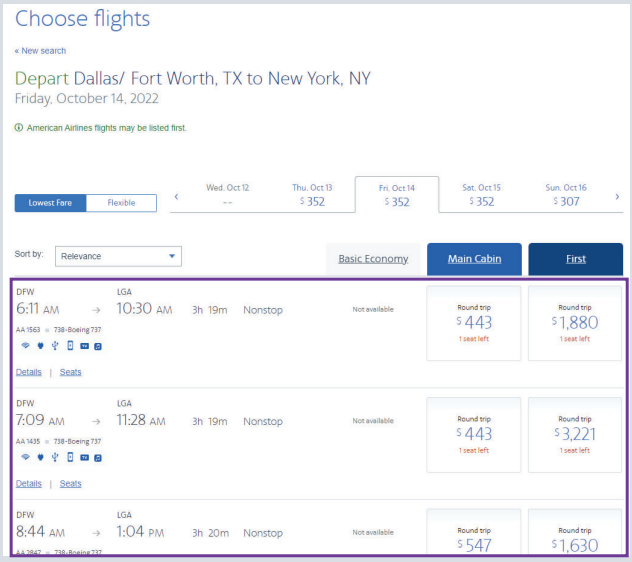
13 The method of claim 12 wherein the at least one presentation rules are supplied via a feed.

Claim Chart – Claim 13

Claim Feature	Evidence
13 The method of claim 12 wherein the at least one presentation rules are supplied via a feed.	AA infringes Claim 13 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claims 1 and 12, the discussion of which is incorporated herein by reference. Furthermore, upon information and belief, the at least one presentation rules are supplied via a feed. For example, the Accused Instrumentalities supply at least one presentation rule via a feed, where the particular details are within AA's possession, custody, and control. **[3] DFW to LGA Search** 

Claim Chart – Claim 13

Claim Feature	Evidence
13 The method of claim 12 wherein the at least one presentation rules are supplied via a feed.	For example, the Accused Instrumentalities supply at least one presentation rule via a feed, where the particular details are within AA's possession, custody, and control.

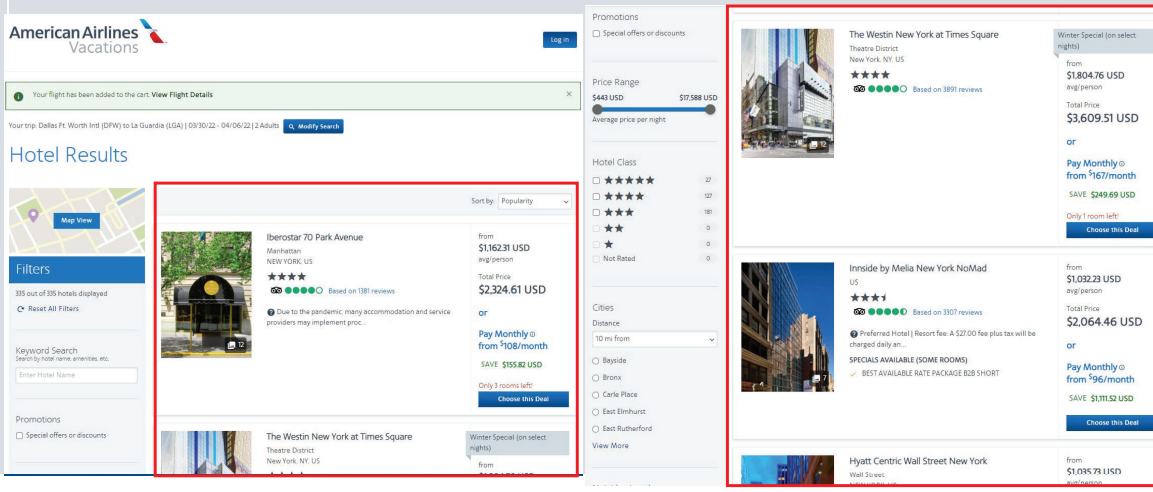


[5] DFW to LGA Flights

Claim 14

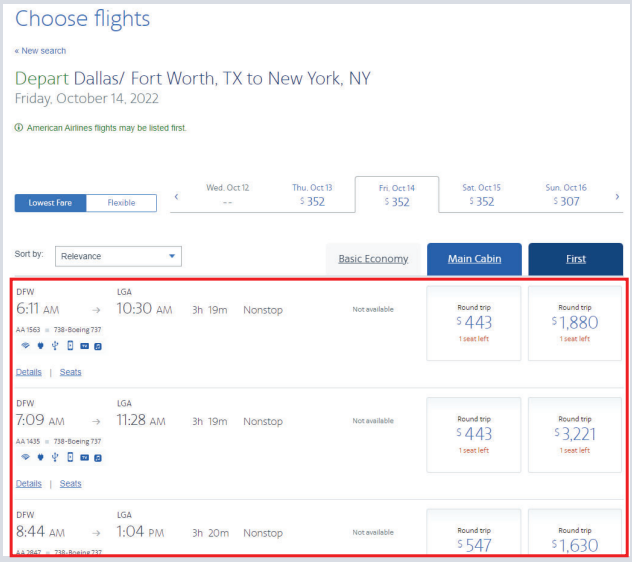
14 The method of claim 12 wherein the at least one presentation rules are embedded in the respective data object.

Claim Chart – Claim 14

Claim Feature	Evidence
14 The method of claim 12 wherein the at least one presentation rules are embedded in the respective data object.	AA infringes Claim 14 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claims 1 and 12, the discussion of which is incorporated herein by reference. Furthermore, the at least one presentation rule is embedded in the respective data object. For example, particular data objects follow particular presentation rules embedded in the respective data object, where the particular details are within AA's possession, custody, and control. **[3] DFW to LGA Search** 

Claim Chart – Claim 14

Claim Feature	Evidence
14 The method of claim 12 wherein the at least one presentation rules are embedded in the respective data object.	For example, particular data objects follow particular presentation rules embedded in the respective data object, where the particular details are within AA's possession, custody, and control.

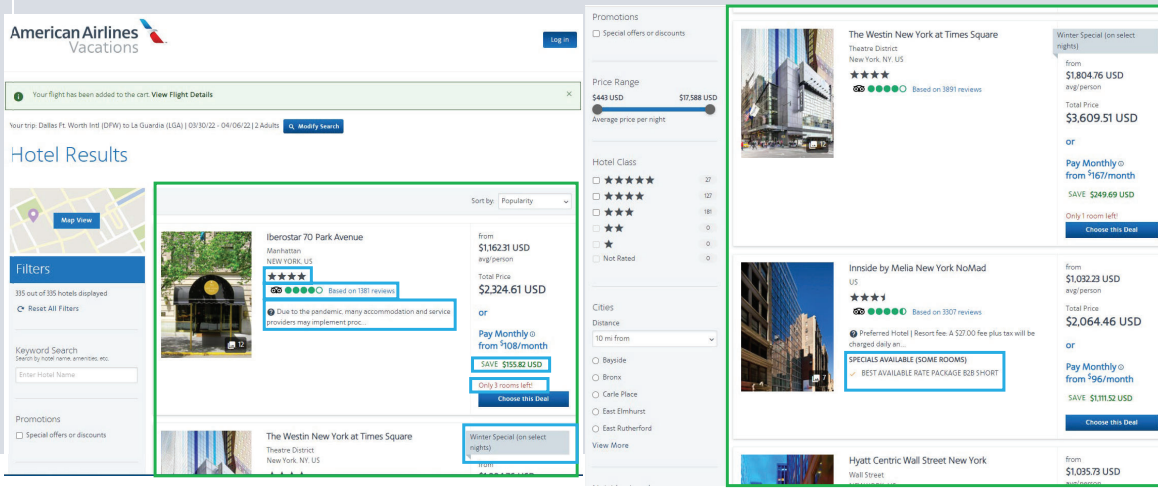


[5] DFW to LGA Flights

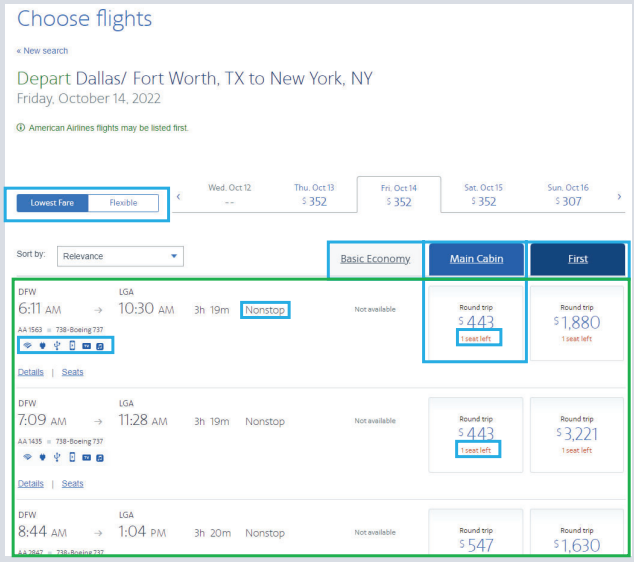
Claim 16

16 The method of claim 12 wherein at least one of the ranked plurality of data objects matches at least two presentation rules and each of the at least two presentation rules are used to construct the at least one display entry for the respective data object.

Claim Chart – Claim 16

Claim Feature	Evidence
16 The method of claim 12 wherein at least one of the ranked plurality of data objects matches at least two presentation rules and each of the at least two presentation rules are used to construct the at least one display entry for the respective data object.	AA infringes Claim 16 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claims 1 and 12, the discussion of which is incorporated herein by reference. Furthermore, at least one of the ranked plurality of data objects matches at least two presentation rules and each of the at least two presentation rules are used to construct the at least one display entry for the respective data object. For example, a data object is matched with at least two presentation rules to construct a display entry that includes ratings, deal indications, price cuts, etc., where the particular details are within AA's possession, custody, and control. **[3] DFW to LGA Search** 

Claim Chart – Claim 16

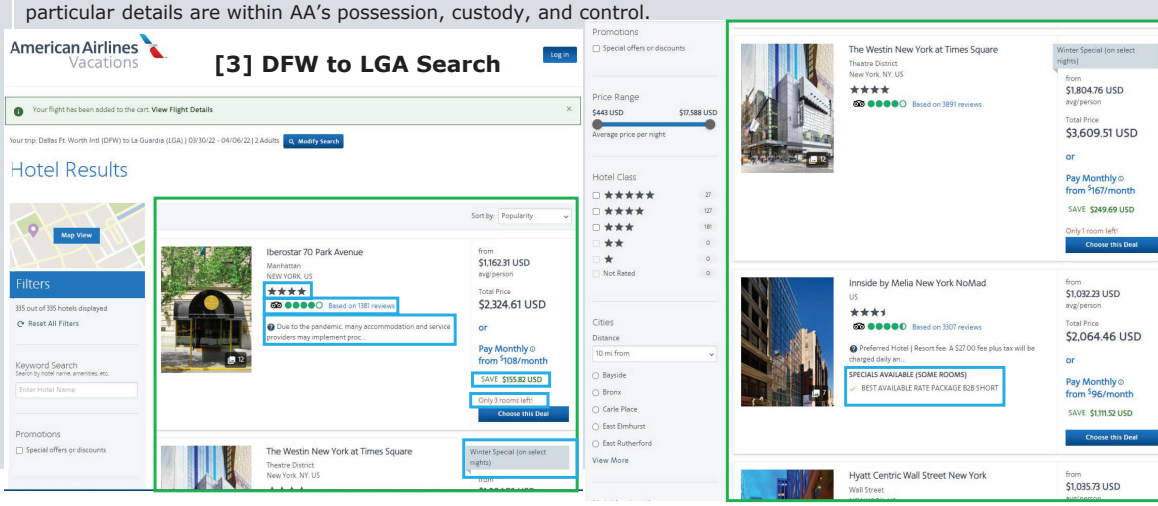
Claim Feature	Evidence
16 The method of claim 12 wherein at least one of the ranked plurality of data objects matches at least two presentation rules and each of the at least two presentation rules are used to construct the at least one display entry for the respective data object.	For example, a data object is matched with at least two presentation rules to construct a display entry that includes availability, amenities, etc., where the particular details are within AA's possession, custody, and control.  **[5] DFW to LGA Flights**

Claim 22

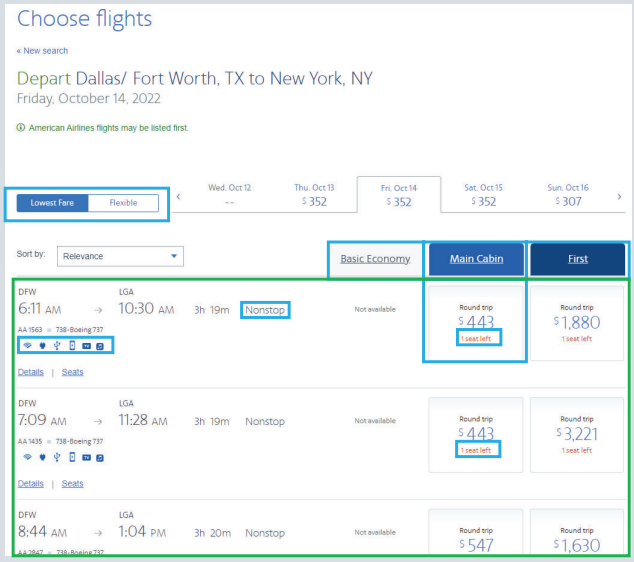
22 The computer-readable medium of claim 2 wherein, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object.

Claim Chart – Claim 22

Claim Feature	Evidence
22 The computer-readable medium of claim 2 wherein, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object.	AA infringes Claim 22 of the '157 patent. The Accused Instrumentalities satisfy each and every element of Claim 2, the discussion of which is incorporated herein by reference. Furthermore, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object. For example, each of the ranked data objects can be matched against at least one presentation rule to construct a display entry that includes ratings, deal indications, price cuts, etc., where the particular details are within AA's possession, custody, and control.



Claim Chart – Claim 22

Claim Feature	Evidence
22 The computer-readable medium of claim 2 wherein, in the building step, each of the ranked plurality of data objects to which at least one intent has been assigned are matched against at least one presentation rule associated with the respective data object and matching presentation rules are used to construct the at least one display entry for the respective data object.	For example, each of the ranked data objects can be matched against at least one presentation rule to construct a display entry that includes availability, amenities, etc., where the particular details are within AA's possession, custody and control.  **[5] DFW to LGA Flights**

END

Exhibit C

Preliminary Infringement Contentions

R2 Solutions LLC v. American Airlines, Inc.
Civil Action No. 4:22-cv-00353-ALM

U.S. Pat. No.: 7,698,329

Claims: 1, 4-5, 8, and 11-12

Preliminary Infringement Contentions

R2 Solutions LLC ("R2" or "Plaintiff") contends that American Airlines, Inc. ("AA" or "Defendant") has infringed and continues to infringe U.S. Pat. No. 7,698,329 (the "'329 Patent") by, among other things, making, having made, using, offering for sale, and/or selling within this judicial district and elsewhere in the United States and/or importing into this judicial district and into the United States – without license or authority – the "Accused Instrumentalities," which include systems, services, processes, methods, acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s). "Accused Instrumentalities" include, without limitation, all instrumentalities associated with receiving, handling, processing, responding to, or otherwise interacting with queries (including user queries and back-end queries) and all instrumentalities related to the location, identification, delivery, display, rendering, ranking, and communication of information relating to such queries. The "Accused Instrumentalities" include, without limitation, the front-end(s) and back-end(s) of AA's online travel agency and search engine services for hotels, vacation rentals, flights, and airport transfer incorporated into the AA web platform and/or AA mobile application. The Accused Instrumentalities practice the steps of the asserted claims or are capable of performing the claim elements of the asserted claims, or otherwise satisfy all elements, described or set forth in one or more claims of the '329 Patent in violation of at least 35 U.S.C. § 271(a). AA's infringements also include, and R2 is entitled to recover damages for, sales of infringing products and components sold and/or shipped abroad that are subsequently imported into the United States.

Preliminary Infringement Contentions

R2 contends that each and every limitation of the asserted claims of the '329 Patent is literally present or performed and infringed by each and every Accused Instrumentality. In the alternative, R2 contends that all limitations of the asserted claims are present under the doctrine of equivalents because (1) to the extent there are any differences between the Accused Instrumentalities and those claim limitations (which R2 denies), such differences are insubstantial. Equivalency may also be shown by the fact that (2) the Accused Instrumentalities perform substantially the same function, in substantially the same way, to achieve substantially the same result as the functions recited in the limitations of the asserted claims.

R2 reserves the right to supplement or amend its claim charts, preliminary infringement contentions, and expert reports as may be appropriate or permissible under the applicable rules and Scheduling Order, including the Court's modifications to P.R. 3-1 available in the Scheduling Order. Any reference or citation to the specification is for illustrative purposes only and is not intended to suggest that the scope of the claimed invention is limited to any particular preferred embodiment. R2 does not hereby waive any applicable work product protections.

For the avoidance of doubt, the claim elements charted herein are software limitations and/or associated with software limitations for the purposes of P.R. 3-1(g), and to the extent that such elements are performed by hardware, firmware, software, and/or any combination thereof, the particular details are in AA's possession, custody, and control.

References Cited

The following chart presents R2 Solutions' analysis of the Accused Instrumentalities based on publicly available information. The citations in the chart refer to the following publicly available documents, which are incorporated by reference as if fully set forth herein. This infringement analysis is preliminary, and R2's investigation into AA's infringement is ongoing. R2 reserves the right to provide additional theories under which AA products/systems infringe this patent:

[1] "**AA Vacations Homepage**," https://www.aavacations.com/en/.

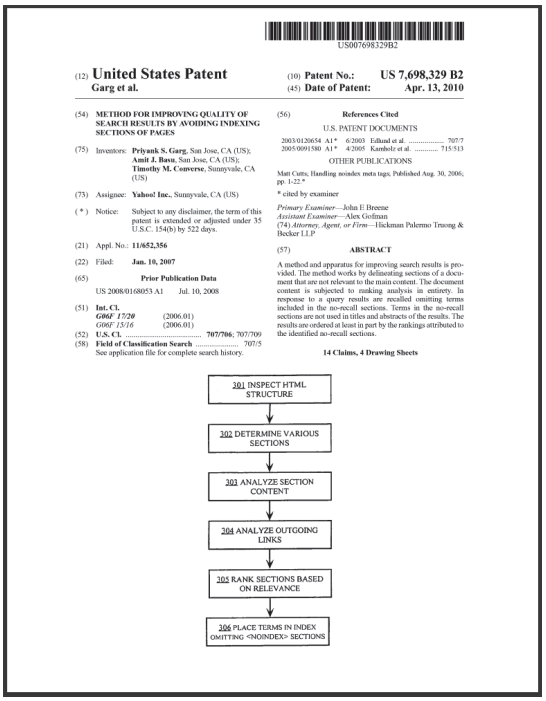
[2] "**DFW to LGA Search**," https://book.aavacations.com/apps/shopping/#/search-results/hotels/1riinvhawu3s7llzd6p/1.

[3] **"AA Vacations,"** https://www.aavacations.com/.

[4] "**AA Flights**," https://www.aa.com/homePage.do.

[5] "**DFW to LGA Flights**," https://www.aa.com/booking/flights/choose-flights/flight1?bookingPathStateId=1665678472499-826&redirectSearchToLegacyAACom=false.

U.S. Patent No. 7,698,329



US007698329B2

(12) **United States Patent**
Garg et al.

(10) **Patent No.: US 7,698,329 B2**
(45) **Date of Patent: Apr. 13, 2010**

(54) **METHOD FOR IMPROVING QUALITY OF SEARCH RESULTS BY AVOIDING INDEXING SECTIONS OF PAGES**

(75) Inventors: **Priyank S. Garg**, San Jose, CA (US); **Amit J. Basu**, San Jose, CA (US); **Timothy M. Converse**, Sunnyvale, CA (US)

(73) Assignee: **Yahoo! Inc.**, Sunnyvale, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 522 days.

(21) Appl. No.: **11/652,356**

(22) Filed: **Jan. 10, 2007**

(65) **Prior Publication Data**

US 2008/0168053 A1 Jul. 10, 2008

(51) **Int. Cl.**
G06F 17/20 (2006.01)
G06F 15/16 (2006.01)

(52) **U.S. Cl.** **707/706**; 707/709

(58) **Field of Classification Search** 707/5
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

2003/0120654 A1* 6/2003 Edlund et al. 707/7
2005/0091580 A1* 4/2005 Kambolz et al. 715/513

OTHER PUBLICATIONS

Matt Cutts; Handling noindex meta tags; Published Aug. 30, 2006; pp. 1-22.*

* cited by examiner

Primary Examiner—John E Breene
Assistant Examiner—Alex Gofman
(74) *Attorney, Agent, or Firm*—Hickman Palermo Truong & Becker LLP

(57) **ABSTRACT**

A method and apparatus for improving search results is provided. The method works by delineating sections of a document that are not relevant to the main content. The document content is subjected to ranking analysis in entirety. In response to a query results are recalled omitting terms included in the no-recall sections. Terms in the no-recall sections are not used in titles and abstracts of the results. The results are ordered at least in part by the rankings attributed to the identified no-recall sections.

14 Claims, 4 Drawing Sheets

Bibliographic Details	
Title	Method for improving quality of search results by avoiding indexing sections of pages
Priority Date	January 10, 2007
Filing Date	January 10, 2007
Issue Date	April 13, 2010
Expiry Date	June 15, 2028

(57) **ABSTRACT**

A method and apparatus for improving search results is provided. The method works by delineating sections of a document that are not relevant to the main content. The document content is subjected to ranking analysis in entirety. In response to a query results are recalled omitting terms included in the no-recall sections. Terms in the no-recall sections are not used in titles and abstracts of the results. The results are ordered at least in part by the rankings attributed to the identified no-recall sections.

Claim 1

1[pre] A method, comprising:

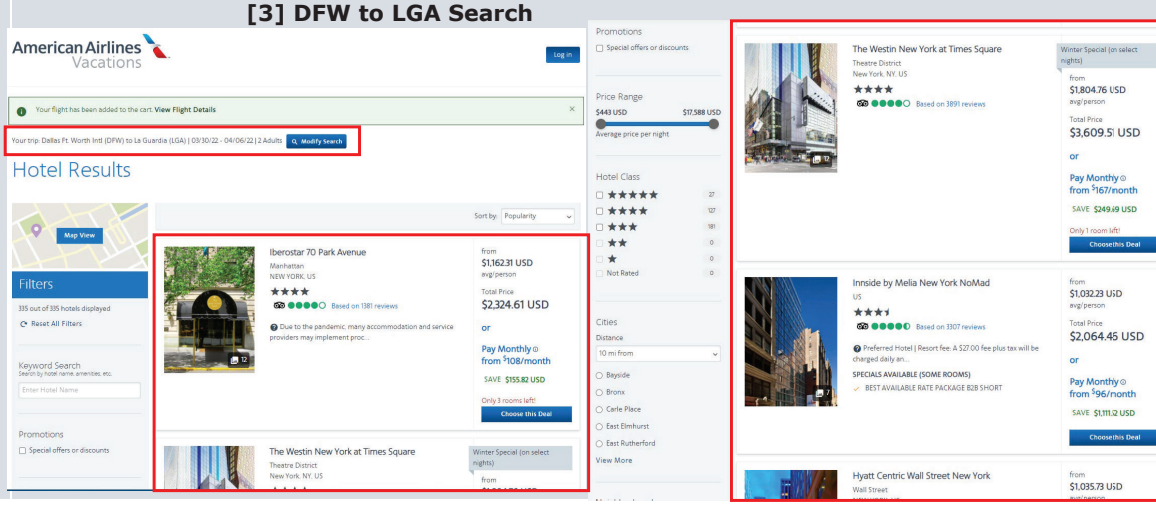
1[a] ranking a plurality of documents recalled by a search engine for a query;

1[b] wherein the plurality of documents contain certain documents, each document of said certain documents containing at least one section that is not used by said search engine for recall and one or more sections that are used by said search engine for recall;

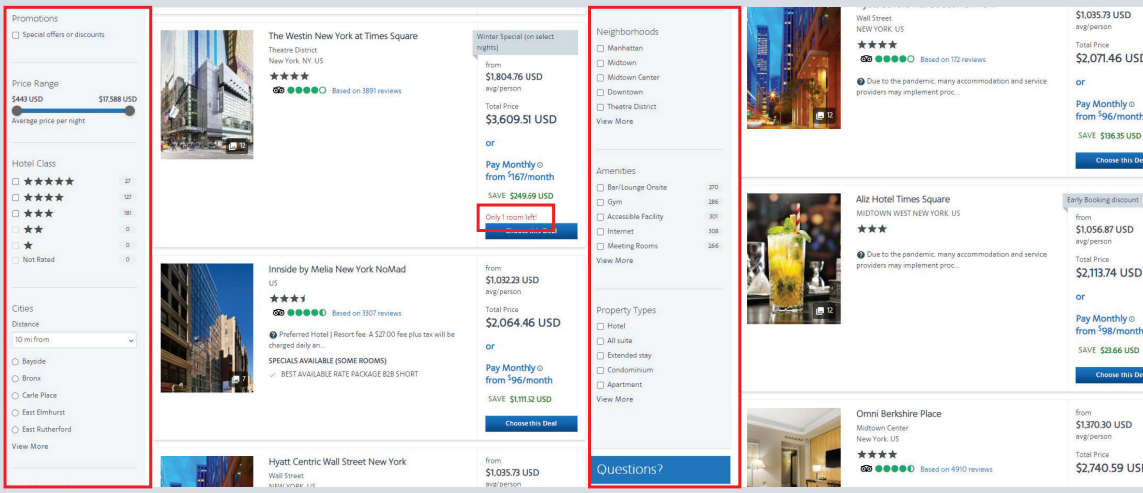
1[c] wherein ranking a plurality of documents includes ranking said plurality of documents based, at least in part, on the at least one section of said certain documents not used by said search engine to recall documents; and

1[d] wherein the method is performed by one or more computing devices.

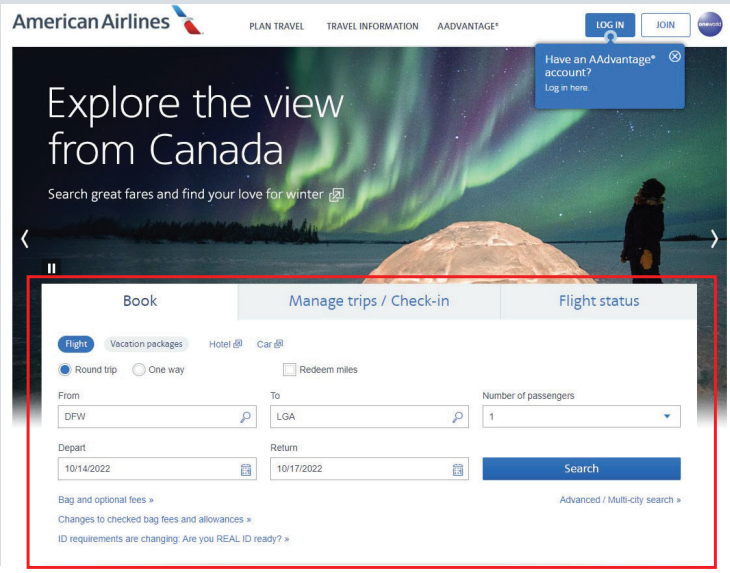
Claim Chart – Claim 1

Claim Feature	Evidence
1[pre] A method, comprising: **1[a]** ranking a plurality of documents recalled by a search engine for a query;	AA infringes Claim 1 of the '329 Patent. The Accused Instrumentalities rank a plurality of documents recalled by a search engine for a query. The Accused Instrumentalities include systems, services, processes, methods, acts components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s). For example, Accused Instrumentalities include a search engine that recalls a plurality of documents for a query: **[3] DFW to LGA Search** 

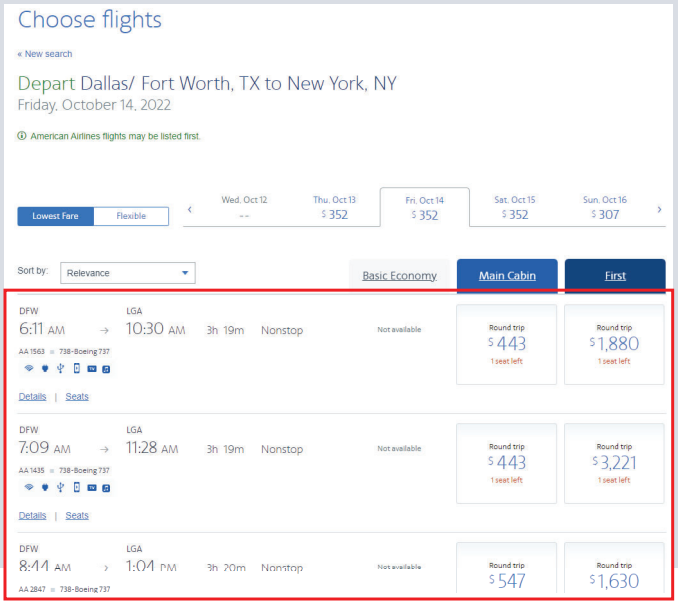
Claim Chart – Claim 1

Claim Feature	Evidence
1[pre] A method, comprising: **1[a]** ranking a plurality of documents recalled by a search engine for a query;	The Accused Instrumentalities rank a plurality of documents recalled by a search engine for a query. The Accused Instrumentalities utilize proprietary ranking algorithms to rank the recalled documents automatically, e.g., by availability, such as on receipt of a query. The particular details are within AA's possession, custody, and control. **[3] DFW to LGA Search** 

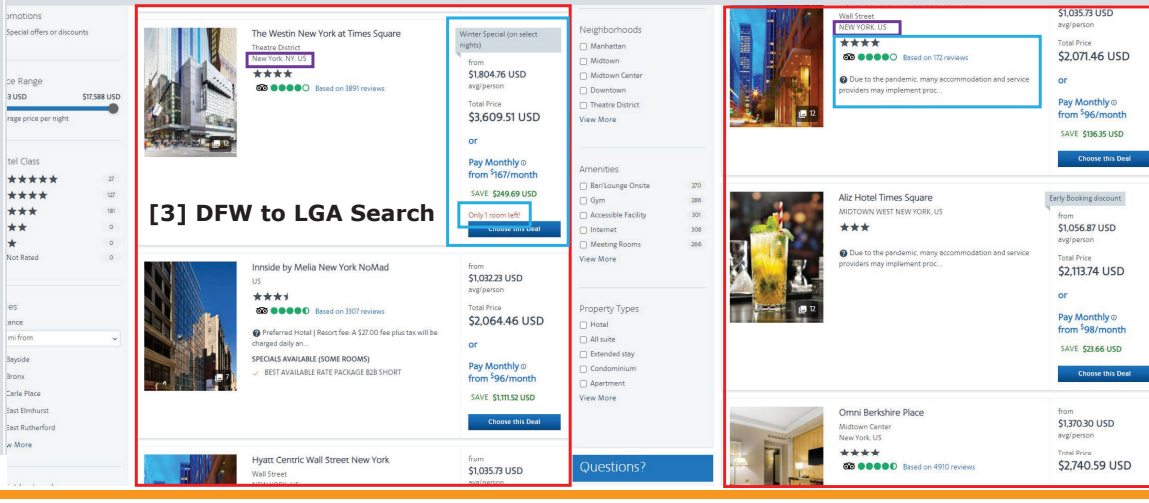
Claim Chart – Claim 1

Claim Feature	Evidence
1[pre] A method, comprising: **1[a]** ranking a plurality of documents recalled by a search engine for a query;	The Accused Instrumentalities rank a plurality of documents recalled by a search engine for a query. The Accused Instrumentalities utilize proprietary ranking algorithms to rank the recalled documents automatically, e.g., by price, such as on receipt of a query. The particular details are within AA's possession, custody, and control.  **[4] AA Flights**

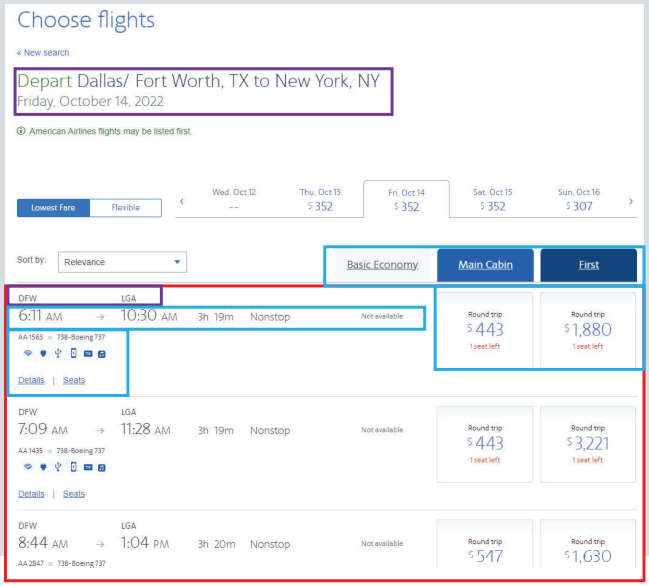
Claim Chart – Claim 1

Claim Feature	Evidence
1[pre] A method, comprising: **1[a]** ranking a plurality of documents recalled by a search engine for a query;	The Accused Instrumentalities rank a plurality of documents recalled by a search engine for a query. The Accused Instrumentalities utilize proprietary ranking algorithms to rank the recalled documents automatically, e.g., by price, such as on receipt of a query. The particular details are within AA's possession, custody, and control.  **[5] DFW to LGA Flights**

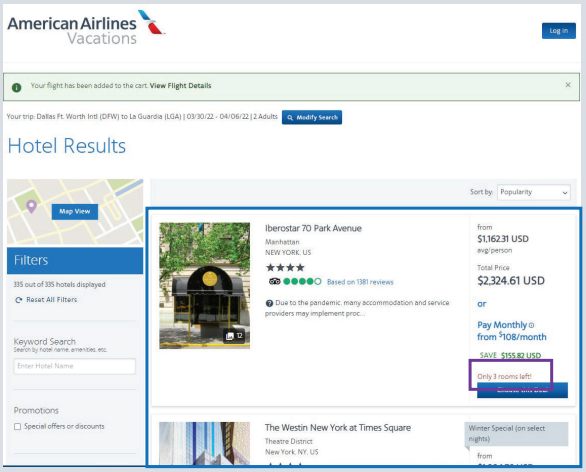
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] wherein the plurality of documents contain certain documents, each document of said certain documents containing at least one section that is not used by said search engine for recall and one or more sections that are used by said search engine for recall;	The plurality of documents of the Accused Instrumentalities contain certain documents, each document of said certain documents containing at least one section that is not used by said search engine for recall and one or more sections that are used by said search engine for recall. For example, the plurality of documents recalled (e.g., corresponding to properties, rooms, etc.) for a search query (e.g., location(s), date, etc.) contains certain documents (e.g., relevant to properties, rooms, etc.) containing at least one section that is not used by the search engine for recall (e.g., availability, etc.) and one or more sections used by the search engine for recall (e.g., location, etc.). See, e.g.: 

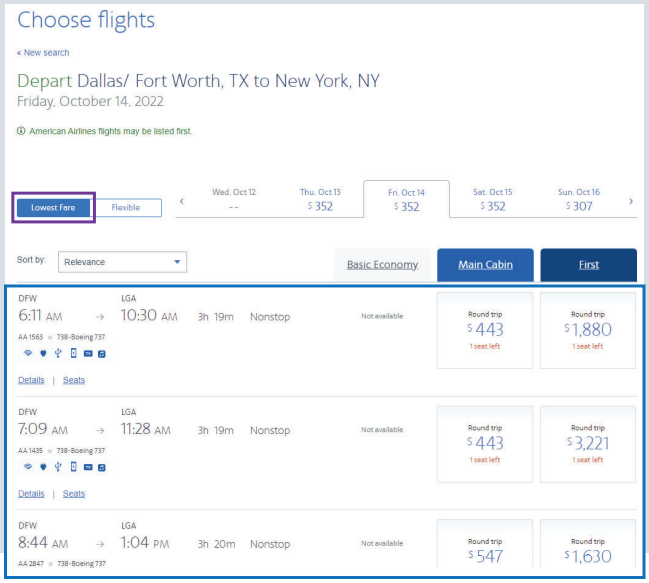
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] wherein the plurality of documents contain certain documents, each document of said certain documents containing at least one section that is not used by said search engine for recall and one or more sections that are used by said search engine for recall;	For example, the plurality of documents recalled (e.g., corresponding to flights) for a search query (e.g., location(s) date, etc.) contains certain documents (e.g., relevant to flights) containing at least one section that is not used by the search engine for recall (e.g., price, etc.) and one or more sections used by the search engine for recall (e.g., location etc.). See, e.g.: **[5] DFW to LGA Flights** 

Claim Chart – Claim 1

Claim Feature	Evidence
1[c] wherein ranking a plurality of documents includes ranking said plurality of documents based, at least in part, on the at least one section of said certain documents not used by said search engine to recall documents; and	Ranking a plurality of documents of the Accused Instrumentalities includes ranking said plurality of documents based at least in part, on the at least one section of said certain documents not used by said search engine to recall documents. For example, the plurality of documents (e.g., corresponding to properties, rooms, etc.) are ranked based at least in part on the at least one section of the certain documents not used by the search engine (e.g., availability etc.) to recall the documents. See, e.g.:  **[3] DFW to LGA Search**

Claim Chart – Claim 1

Claim Feature	Evidence
1[c] wherein ranking a plurality of documents includes ranking said plurality of documents based, at least in part, on the at least one section of said certain documents not used by said search engine to recall documents; and	For example, the plurality of documents (e.g., corresponding to flights, etc.) are ranked based at least in part on the at least one section of the certain documents not used by the search engine (e.g., price, etc.) to recall the documents. See, e.g.: **[5] DFW to LGA Flights** 

Claim Chart – Claim 1

Claim Feature	Evidence
1[d] wherein the method is performed by one or more computing devices.	The method is performed by one or more computing devices. For example, the Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s).

Claim 4

4[a] The method of claim 1, further including said search engine generating data representing results of said query, wherein:

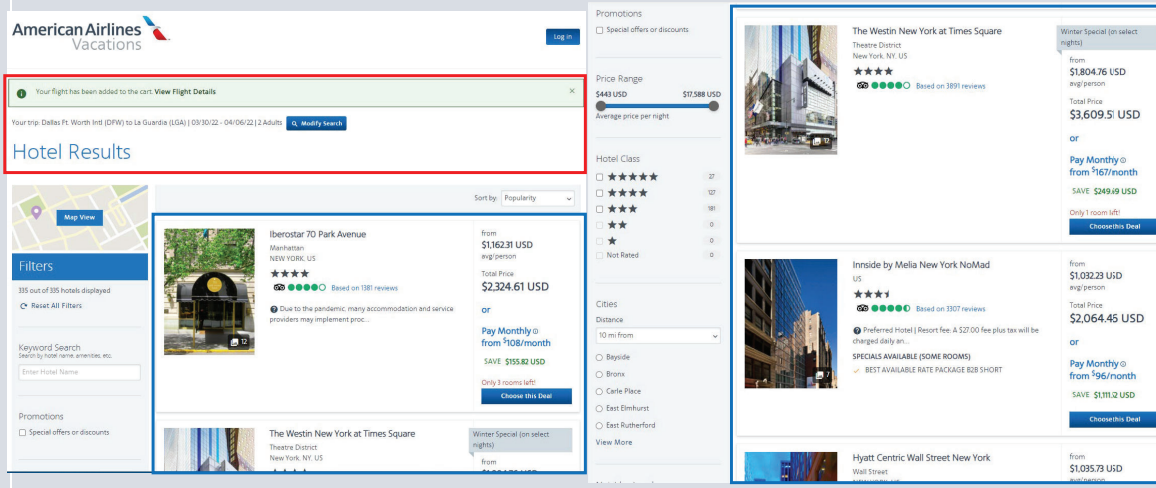
4[b] said data includes an abstract describing each document of said plurality of documents; and

4[c] for each respective abstract of each document of said certain documents, said abstract excludes terms from the respective at least one section not used by said search engine to recall said each document.

Claim Chart – Claim 4

Claim Feature	Evidence
4 **[a]** The method of claim 1, further including said search engine generating data representing results of said query, wherein:	AA infringes Claim 4 of the '329 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, the method performed by the Accused Instrumentalities includes said search engine generating data representing results of said query.

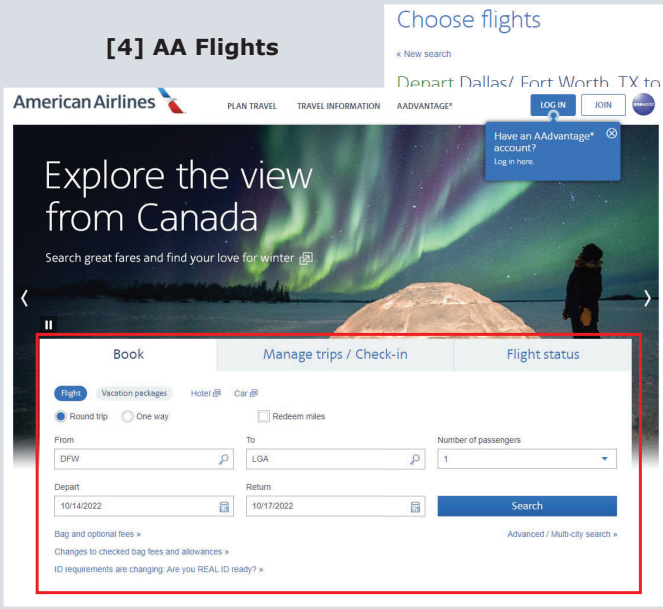
[3] DFW to LGA Search



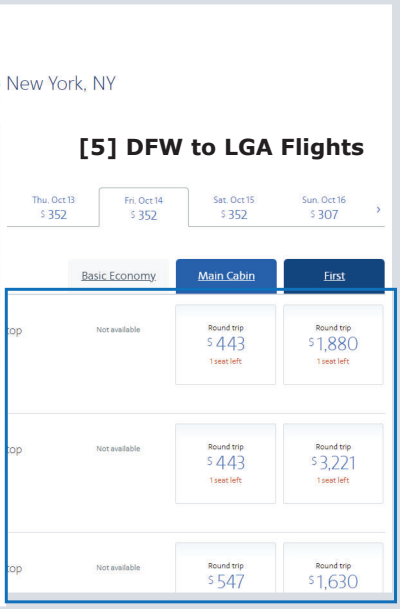
Claim Chart – Claim 4

Claim Feature	Evidence
4 **[a]** The method of claim 1, further including said search engine generating data representing results of said query, wherein:	Furthermore, the method performed by the Accused Instrumentalities includes said search engine generating data representing results of said query.

[4] AA Flights



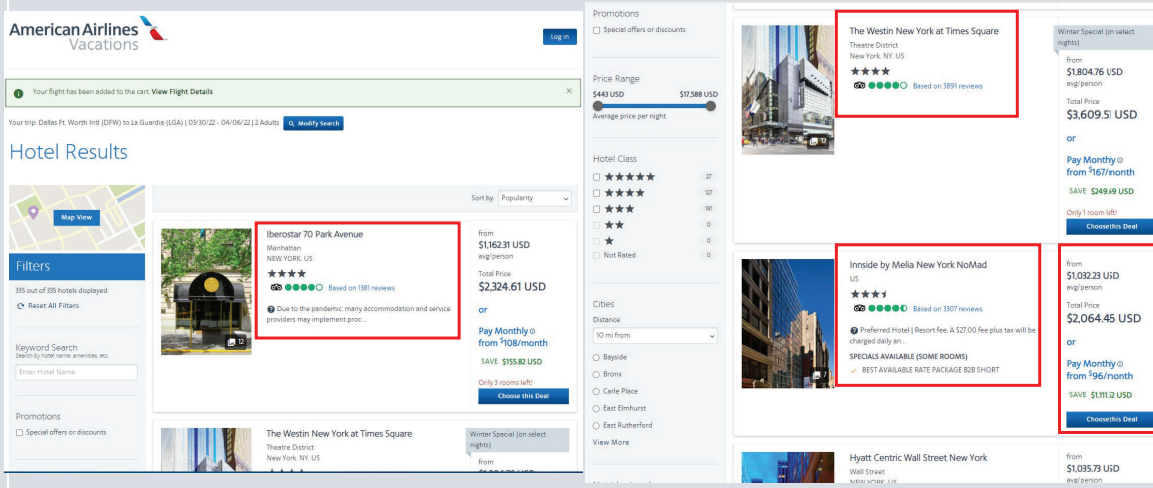
[5] DFW to LGA Flights



Claim Chart – Claim 4

Claim Feature	Evidence
4 **[b]** said data includes an abstract describing each document of said plurality of documents; and	In the method performed by the Accused Instrumentalities, said data includes an abstract describing each document o said plurality of documents. For example, for each result of a search query, there is an abstract describing the document. For example, the abstract can describe each document by name, location, price, etc.

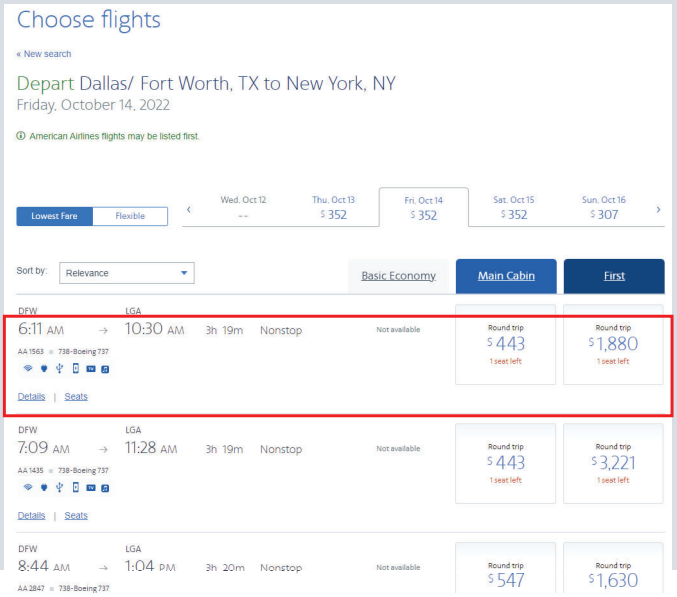
[3] DFW to LGA Search



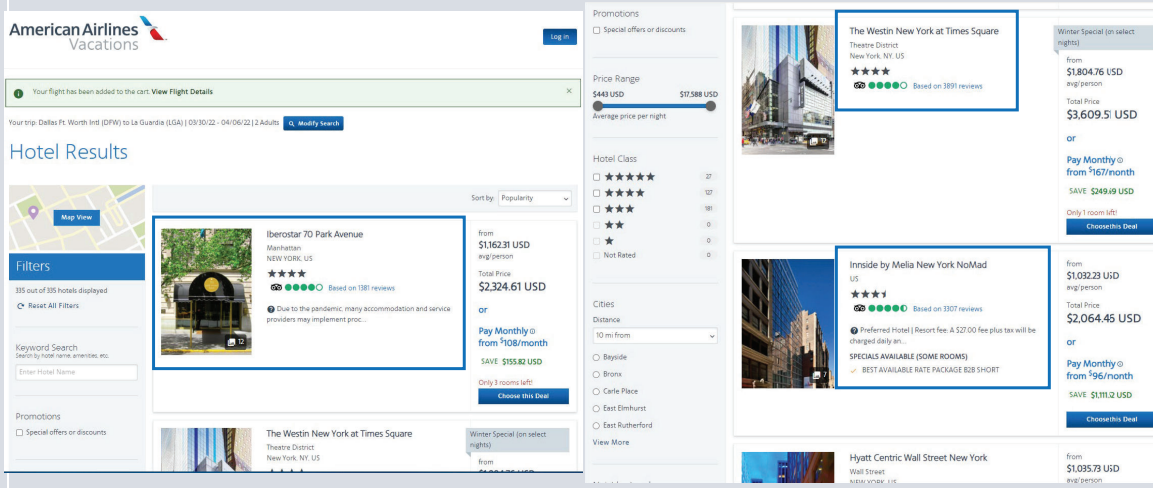
Claim Chart – Claim 4

Claim Feature	Evidence
4 **[b]** said data includes an abstract describing each document of said plurality of documents; and	For example, for each result of a search query, there is an abstract describing the document. For example, the abstrac can describe each document by time, price, etc.

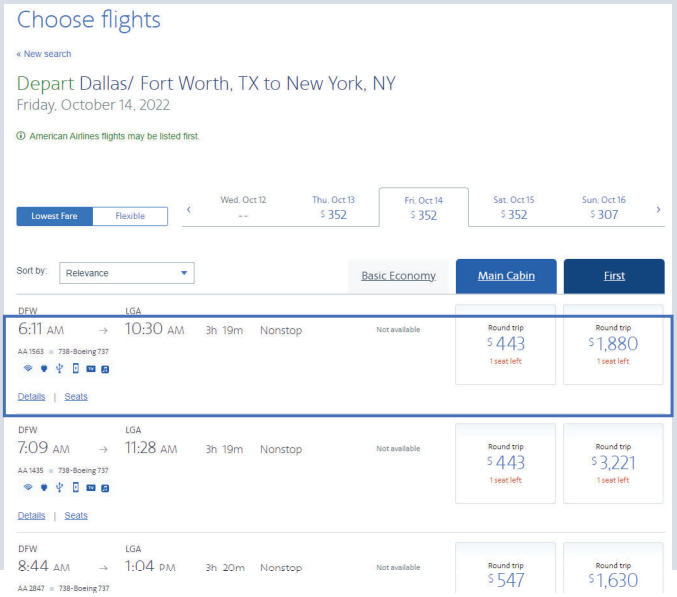
[5] DFW to LGA Flights



Claim Chart – Claim 4

Claim Feature	Evidence
4 [c] for each respective abstract of each document of said certain documents, said abstract excludes terms from the respective at least one section not used by said search engine to recall said each document.	In the method performed by the Accused Instrumentalities, for each respective abstract of each document of said certain documents, said abstract excludes terms from the respective at least one section not used by said search engine to recall said each document. For example, each abstract describes each document by name, location, etc. and excludes terms from the section not used by said search engine to recall the documents (e.g., availability, sponsorship, etc.). **[3] DFW to LGA Search** 

Claim Chart – Claim 4

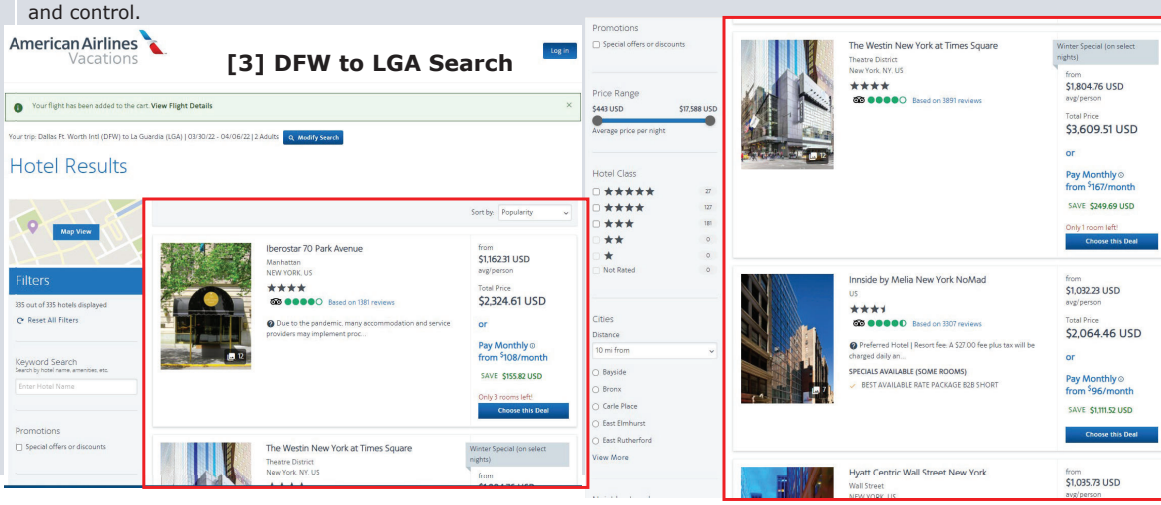
Claim Feature	Evidence
4 [c] for each respective abstract of each document of said certain documents, said abstract excludes terms from the respective at least one section not used by said search engine to recall said each document.	For example, each abstract describes each document by time, price, etc. and excludes terms from the section not used by said search engine to recall the documents (e.g., location, etc.). **[5] DFW to LGA Flights** 

Claim 5

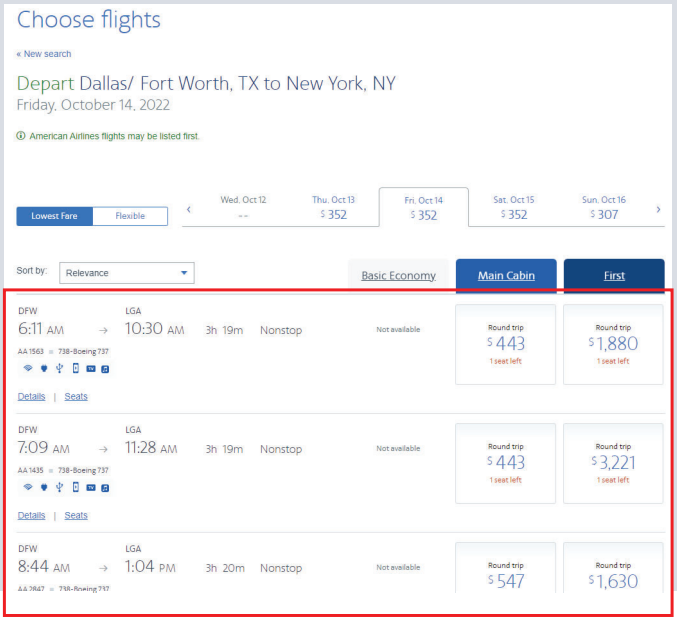
5 The method of claim 1, said search engine using a search engine index to recall said plurality of documents; and wherein said search engine index does not index any term in said certain documents that is found only in one or more sections of said certain documents established as a no-recall section.

Claim Chart – Claim 5

Claim Feature	Evidence
5 The method of claim 1, said search engine using a search engine index to recall said plurality of documents; and wherein said search engine index does not index any term in said certain documents that is found only in one or more sections of said certain documents established as a no-recall section.	AA infringes Claim 5 of the '329 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities utilize a search engine index to recall said plurality of documents, wherein said search engine index does not index any term in said certain documents that is found only in one or more sections of said certain documents established as a no-recall section. For example, said search engine does not index terms found in the sponsorship and/or availability section, and documents cannot be searched by sponsorship or availability. The particular details are within AA's possession, custody and control.



Claim Chart – Claim 5

Claim Feature	Evidence
5 The method of claim 1, said search engine using a search engine index to recall said plurality of documents; and wherein said search engine index does not index any term in said certain documents that is found only in one or more sections of said certain documents established as a no-recall section.	For example, said search engine does not index terms found in the price and/or time sections, and documents canno be searched by time or price. The particular details are within AA's possession, custody, and control. **[5] DFW to LGA Flights** 

Claim 8

8[pre] A computer-readable storage medium that stores instructions which, when executed by one or more processors, cause performance of:

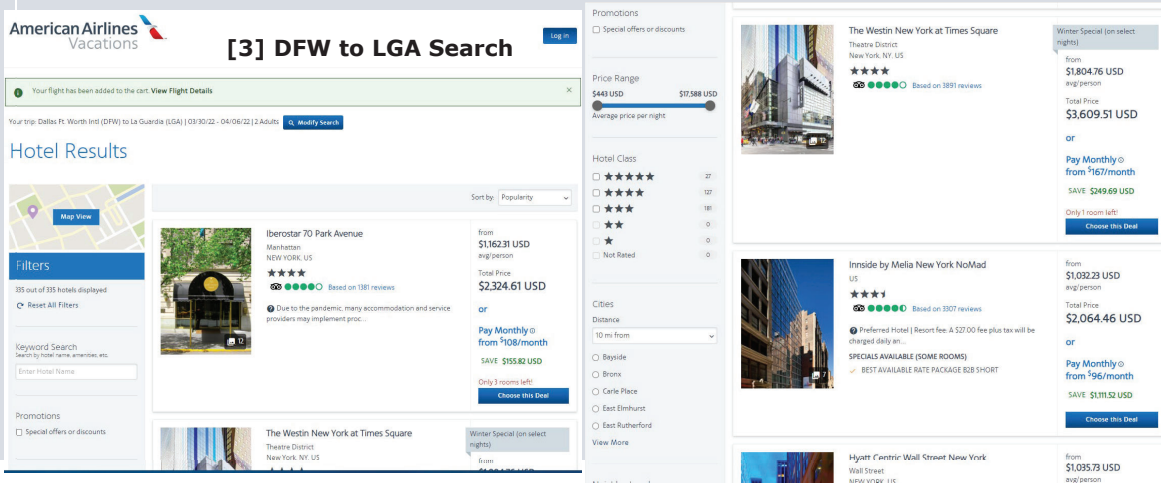
8[a] ranking a plurality of documents recalled by a search engine for a query;

8[b] wherein the plurality of documents contain certain documents, each document of said certain documents containing at least one section that is not used by said search engine for recall and one or more sections that are used by said search engine for recall; and

8[c] wherein ranking a plurality of documents includes ranking said plurality of documents based, at least in part, on the at least one section of said certain documents not used by said search engine to recall documents.

Claim Chart – Claim 8

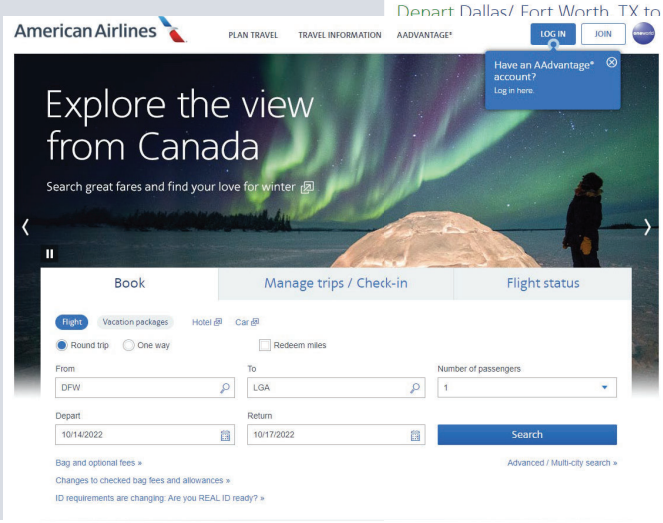
Claim Feature	Evidence
8 [pre] A computer-readable storage medium that stores instructions which, when executed by one or more processors, cause performance of:	AA infringes Claim 8 of the '329 patent. The Accused Instrumentalities comprise a computer-readable storage medium that stores instructions which, when executed by one or more processors cause performance of the steps discussed below. The Accused Instrumentalities systems, services, processes, methods, acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s).



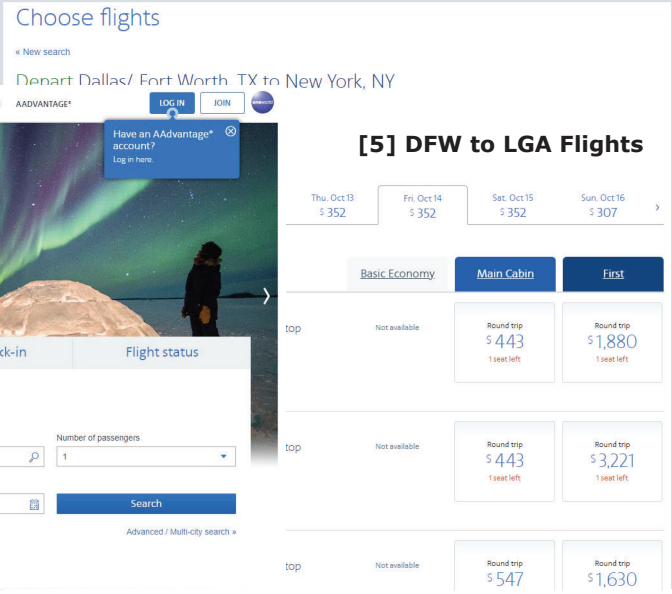
Claim Chart – Claim 8

Claim Feature	Evidence
8 [pre] A computer-readable storage medium that stores instructions which, when executed by one or more processors, cause performance of:	The Accused Instrumentalities systems, services, processes, methods, acts, components (hardware and/or software), o other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s).

[4] AA Flights



[5] DFW to LGA Flights



Claim Chart – Claim 8

Claim Feature	Evidence
8 [a] ranking a plurality of documents recalled by a search engine for a query;	The Accused Instrumentalities rank a plurality of documents recalled by a search engine for a query. The discussion of Claim element 1[a] is incorporated herein by reference.

Claim Chart – Claim 8

Claim Feature	Evidence
8 [b] wherein the plurality of documents contain certain documents, each document of said certain documents containing at least one section that is not used by said search engine for recall and one or more sections that are used by said search engine for recall; and	The plurality of documents contain certain documents, each document of said certain documents containing at least one section that is not used by said search engine for recall and one or more sections that are used by said search engine for recall. The discussion of Claim element 1[b] is incorporated herein by reference.

Claim Chart – Claim 8

Claim Feature	Evidence
8 [c] wherein ranking a plurality of documents includes ranking said plurality of documents based, at least in part, on the at least one section of said certain documents not used by said search engine to recall documents.	The Accused Instrumentalities rank a plurality of documents, which includes ranking said plurality of documents based at least in part, on the at least one section of said certain documents not used by said search engine to recal documents. The discussion of Claim element 1[c] is incorporated herein by reference.

Claim 11

11[a] The computer-readable storage medium of claim 8 further comprising instructions for including said search engine generating data representing results of said query, wherein:

11[b] said data includes an abstract describing each document of said plurality of documents; and

11[c] for each respective abstract of each document of said certain documents, said abstract excludes terms from the respective at least one section not used by said search engine to recall said each document.

Claim Chart – Claim 11

Claim Feature	Evidence
11 [a] The computer-readable storage medium of claim 8 further comprising instructions for including said search engine generating data representing results of said query, wherein:	AA infringes Claim 11 of the '329 patent. The Accused Instrumentalities satisfy each and every element of Claim 8, the discussion of which is incorporated herein by reference. Furthermore, the computer-readable storage medium furthe comprises instructions for including said search engine generating data representing results of said query. The discussion of Claim element 4[a] is incorporated herein by reference.

Claim Chart – Claim 11

Claim Feature	Evidence
11 **[b]** said data includes an abstract describing each document of said plurality of documents; and	The data includes an abstract describing each document of said plurality of documents. The discussion of Claim element 4[b] is incorporated herein by reference.

Claim Chart – Claim 11

Claim Feature	Evidence
11 **[b]** for each respective abstract of each document of said certain documents, said abstract excludes terms from the respective at least one section not used by said search engine to recall said each document.	For each respective abstract of each document of said certain documents, said abstract excludes terms from the respective at least one section not used by said search engine to recall said each document. The discussion of Claim element 4[c] is incorporated herein by reference.

Claim 12

12 The computer-readable storage medium of claim 8, said search engine using a search engine index to recall said plurality of documents; and wherein said search engine index does not index any term in said certain documents that is found only in one or more sections of said certain documents established as a no-recall section.

Claim Chart – Claim 12

Claim Feature	Evidence
12 The computer-readable storage medium of claim 8, said search engine using a search engine index to recall said plurality of documents; and wherein said search engine index does not index any term in said certain documents that is found only in one or more sections of said certain documents established as a no-recall section.	AA infringes Claim 12 of the '329 patent. The Accused Instrumentalities satisfy each and every element of Claim 8, the discussion of which is incorporated herein by reference. Furthermore, said search engine uses a search engine index to recall said plurality of documents; and wherein said search engine index does not index any term in said certain documents that is found only in one or more sections of said certain documents established as a no-recall section. The discussion of Claim 5 is incorporated herein by reference.

END

Exhibit D

Preliminary Infringement Contentions

R2 Solutions LLC v. American Airlines, Inc.
Civil Action No. 4:22-cv-00353-ALM

U.S. Pat. No.: 8,209,317

Claims: 1-5 and 8-14

Preliminary Infringement Contentions

R2 Solutions LLC ("R2" or "Plaintiff") contends that American Airlines, Inc. ("AA" or "Defendant") has infringed and continues to infringe U.S. Pat. No. 8,209,317 (the "'317 Patent") by, among other things, making, having made, using, offering for sale, and/or selling within this judicial district and elsewhere in the United States and/or importing into this judicial district and into the United States – without license or authority – the "Accused Instrumentalities," which include systems, services, processes, methods, acts, components (hardware and/or software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s). "Accused Instrumentalities" include, without limitation, all instrumentalities associated with receiving, handling, processing, responding to, or otherwise interacting with queries (including user queries and back-end queries) and all instrumentalities related to the location, identification, delivery, display, rendering, ranking, and communication of information relating to such queries. The "Accused Instrumentalities" include, without limitation, the front-end(s) and back-end(s) of AA's online travel agency and search engine services for hotels, vacation rentals, flights, and airport transfer incorporated into the AA web platform and/or AA mobile application. The Accused Instrumentalities practice the steps of the asserted claims or are capable of performing the claim elements of the asserted claims, or otherwise satisfy all elements, described or set forth in one or more claims of the '317 Patent in violation of at least 35 U.S.C. § 271(a). AA's infringements also include, and R2 is entitled to recover damages for, sales of infringing products and components sold and/or shipped abroad that are subsequently imported into the United States.

Preliminary Infringement Contentions

R2 contends that each and every limitation of the asserted claims of the '317 Patent is literally present or performed and infringed by each and every Accused Instrumentality. In the alternative, R2 contends that all limitations of the asserted claims are present under the doctrine of equivalents because (1) to the extent there are any differences between the Accused Instrumentalities and those claim limitations (which R2 denies), such differences are insubstantial. Equivalency may also be shown by the fact that (2) the Accused Instrumentalities perform substantially the same function, in substantially the same way, to achieve substantially the same result as the functions recited in the limitations of the asserted claims.

R2 reserves the right to supplement or amend its claim charts, preliminary infringement contentions, and expert reports as may be appropriate or permissible under the applicable rules and Scheduling Order, including the Court's modifications to P.R. 3-1 available in the Scheduling Order. Any reference or citation to the specification is for illustrative purposes only and is not intended to suggest that the scope of the claimed invention is limited to any particular preferred embodiment. R2 does not hereby waive any applicable work product protections.

For the avoidance of doubt, the claim elements charted herein are software limitations and/or associated with software limitations for the purposes of P.R. 3-1(g), and to the extent that such elements are performed by hardware, firmware, software, and/or any combination thereof, the particular details are in AA's possession, custody, and control.

References Cited

The following chart presents R2 Solutions' analysis of the Accused Instrumentalities based on publicly available information. The citations in the chart refer to the following publicly available documents, which are incorporated by reference as if fully set forth herein. This infringement analysis is preliminary, and R2's investigation into AA's infringement is ongoing. R2 reserves the right to provide additional theories under which AA products/systems infringe this patent:

[1] "**AA Vacations Homepage**," https://www.aavacations.com/en/.

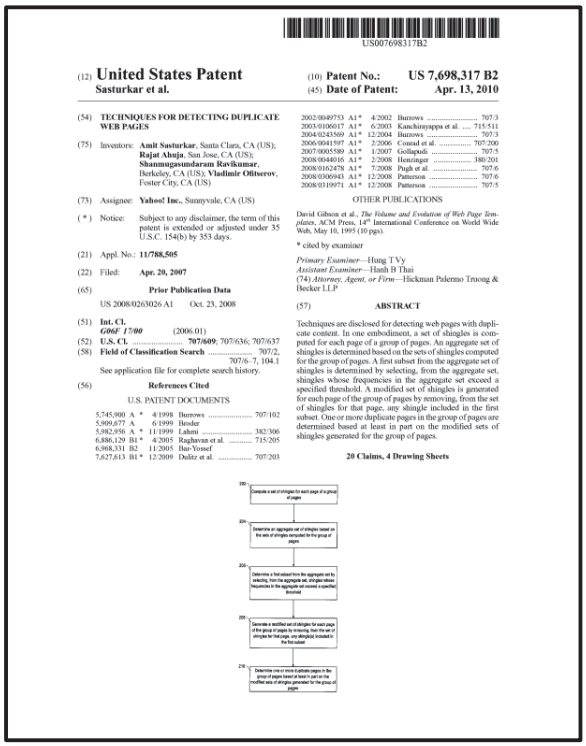
[2] "**DFW to LGA Search**," https://book.aavacations.com/apps/shopping/#/search-results/hotels/1riinvhawu3s7llzd6p/1.

[3] **"AA Vacations,"** https://www.aavacations.com/.

[4] "**AA Flights**," https://www.aa.com/homePage.do.

[5] "**DFW to LGA Flights**," https://www.aa.com/booking/flights/choose-flights/flight1?bookingPathStateId=1665678472499-826&redirectSearchToLegacyAACom=false.

U.S. Patent No. 7,698,317



US007698317B2

(12) **United States Patent**
Sasturkar et al.

(10) **Patent No.: US 7,698,317 B2**
(45) **Date of Patent: Apr. 13, 2010**

(54) **TECHNIQUES FOR DETECTING DUPLICATE WEB PAGES**

(75) Inventors: **Amit Sasturkar**, Santa Clara, CA (US); **Rajat Ahuja**, San Jose, CA (US); **Shanmugasundaram Ravikumar**, Berkeley, CA (US); **Vladimir Ofitserov**, Foster City, CA (US)

(73) Assignee: **Yahoo! Inc.**, Sunnyvale, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 353 days.

(21) Appl. No.: **11/788,505**

(22) Filed: **Apr. 20, 2007**

(65) **Prior Publication Data**

US 2008/0263026 A1 Oct. 23, 2008

(51) **Int. Cl.**
G06F 17/00 (2006.01)

(52) **U.S. Cl.** **707/609**; 707/636; 707/637

(58) **Field of Classification Search** 707/2, 707/6–7, 104.1

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,745,900 A * 4/1998 Burrows 707/102
5,909,677 A 6/1999 Broder
5,982,956 A * 11/1999 Lahmi 382/306
6,886,129 B1 * 4/2005 Raghavan et al. 715/205
6,968,331 B2 11/2005 Bar-Yossef
7,627,613 B1 * 12/2009 Dulitz et al. 707/203
2002/0049753 A1 * 4/2002 Burrows 707/3
2003/0196017 A1 * 6/2003 Kanchirayappa et al. 715/511
2004/0243569 A1 * 12/2004 Burrows 707/3
2006/0041597 A1 * 2/2006 Conrad et al. 707/200
2007/0005580 A1 * 1/2007 Gollapudi 707/5
2008/0044016 A1 * 2/2008 Henzinger 380/201
2008/0162478 A1 * 7/2008 Pugh et al. 707/6
2008/0306943 A1 * 12/2008 Patterson 707/6
2008/0319971 A1 * 12/2008 Patterson 707/5

OTHER PUBLICATIONS

David Gibson et al., *The Volume and Evolution of Web Page Templates*, ACM Press, 14th International Conference on World Wide Web, May 10, 1995 (10 pgs).

* cited by examiner

Primary Examiner—Hung T Vy
Assistant Examiner—Hanh B Thai
(74) *Attorney, Agent, or Firm*—Hickman Palermo Truong & Becker LLP

(57) **ABSTRACT**

Techniques are disclosed for detecting web pages with duplicate content. In one embodiment, a set of shingles is computed for each page of a group of pages. An aggregate set of shingles is determined based on the sets of shingles computed for the group of pages. A first subset from the aggregate set of shingles is determined by selecting, from the aggregate set, shingles whose frequencies in the aggregate set exceed a specified threshold. A modified set of shingles is generated for each page of the group of pages by removing, from the set of shingles for that page, any shingle included in the first subset. One or more duplicate pages in the group of pages are determined based at least in part on the modified sets of shingles generated for the group of pages.

20 Claims, 4 Drawing Sheets

Bibliographic Details	
Title	Techniques for detecting duplicate web pages
Priority Date	April 20, 2007
Filing Date	April 20, 2007
Issue Date	April 13, 2010
Expiry Date	April 7, 2028

(57) **ABSTRACT**

Methods and systems for reconstructing a full query based on a partial query are disclosed. Existing interfaces for search engines may be rigid and require users to submit full queries to perform searched. The methods and systems described herein may solve these problems by allowing a flexible way for users to submit a partial query and reconstruct a full query based on the partial query. A search may then be performed using the reconstructed query.

Claim 1

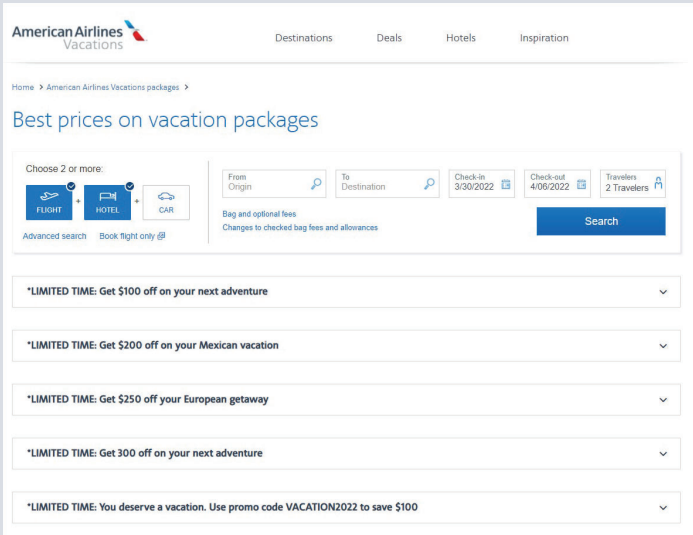
1[pre] A computer database system for providing search results to a user in response to user submissions over a data network, the computer database system comprising:

1[a] a database configured to store information about events in the computer database system; and

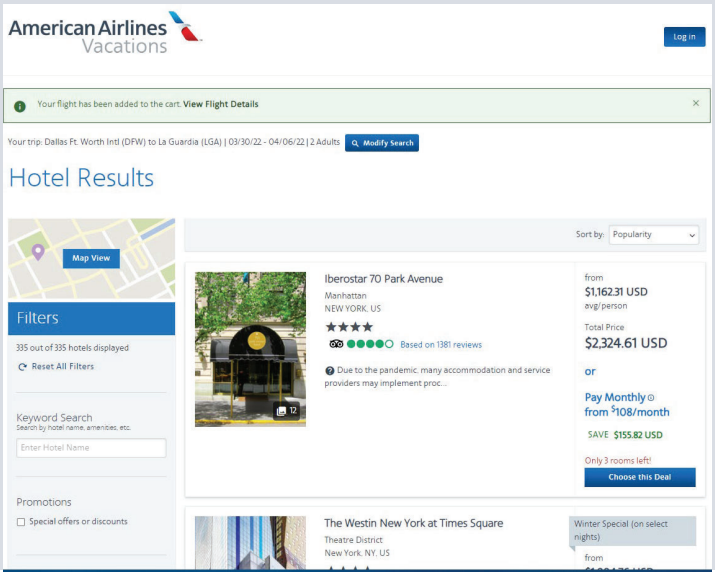
1[b] a query reconstruction server in data communication with the database and operative to receive a partial query submitted at a remote user client system by a user seeking search results matching the submitted partial query and, in response to the received partial query, determine a full query based on (i) the received partial query, and (ii) information stored in the database about queries previously-submitted by users,

1[c] wherein the submitted partial query comprises an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user and the full query is better representative of the entire search query desired by the user.

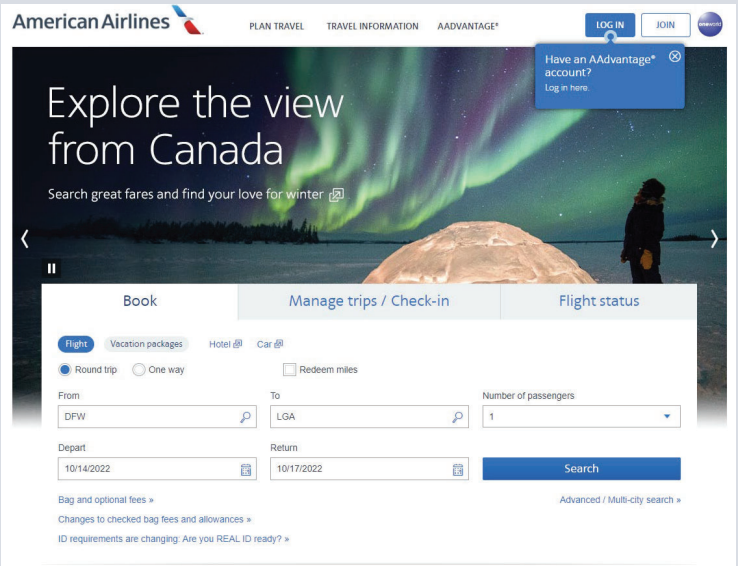
Claim Chart – Claim 1

Claim Feature	Evidence
1[pre] A computer database system for providing search results to a user in response to user submissions over a data network, the computer database system comprising:	AA infringes Claim 1 of the '317 Patent. The Accused Instrumentalities include a computer database system fo providing search results to a user in response to user submissions over a data network. The Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/or software), or othe instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s): **[1] AA Vacations Homepage** 

Claim Chart – Claim 1

Claim Feature	Evidence
1[pre] A computer database system for providing search results to a user in response to user submissions over a data network, the computer database system comprising:	The Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/o software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s): **[3] DFW to LGA Search** 

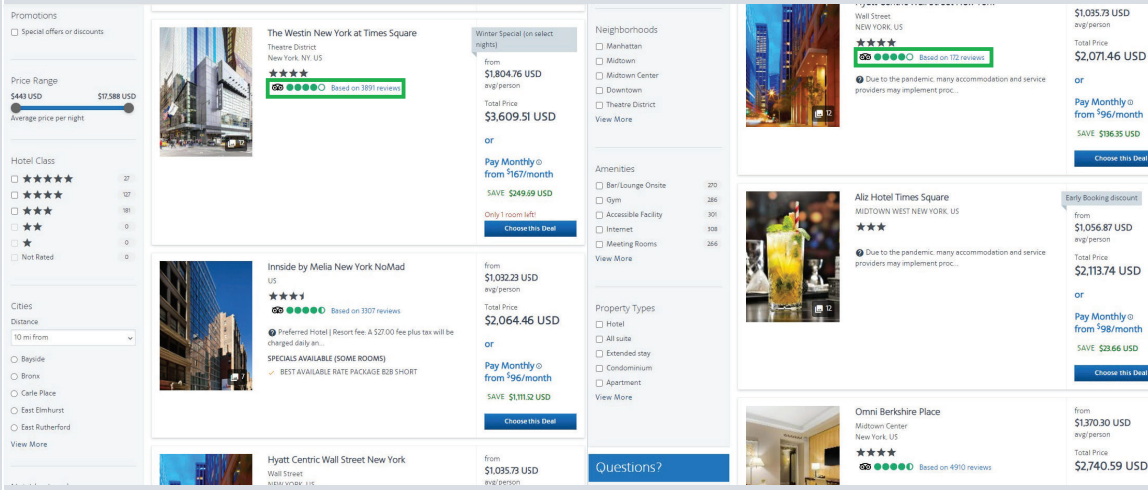
Claim Chart – Claim 1

Claim Feature	Evidence
1[pre] A computer database system for providing search results to a user in response to user submissions over a data network, the computer database system comprising:	The Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/o software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s):  **[4] AA Flights**

Claim Chart – Claim 1

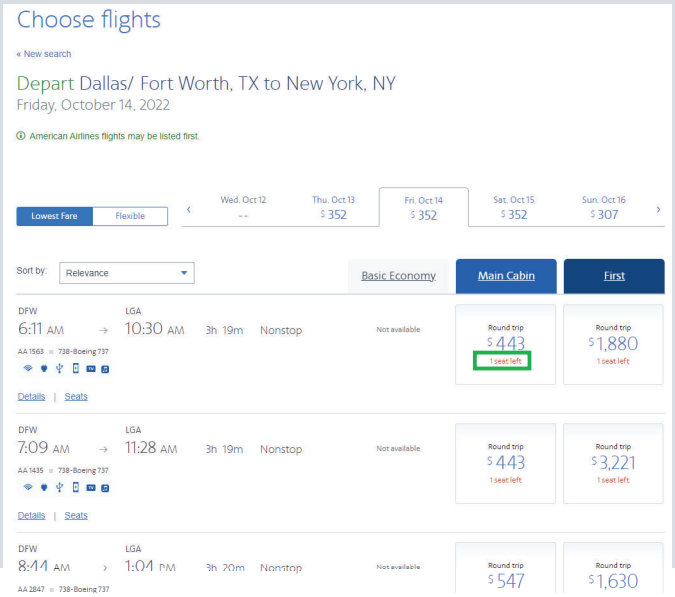
Claim Feature	Evidence
1[a] a database configured to store information about events in the computer database system; and	The Accused Instrumentalities include a database configured to store information about events in the compute database system. For example, the Accused Instrumentalities store information regarding reviews of particula offerings:

[3] DFW to LGA Search



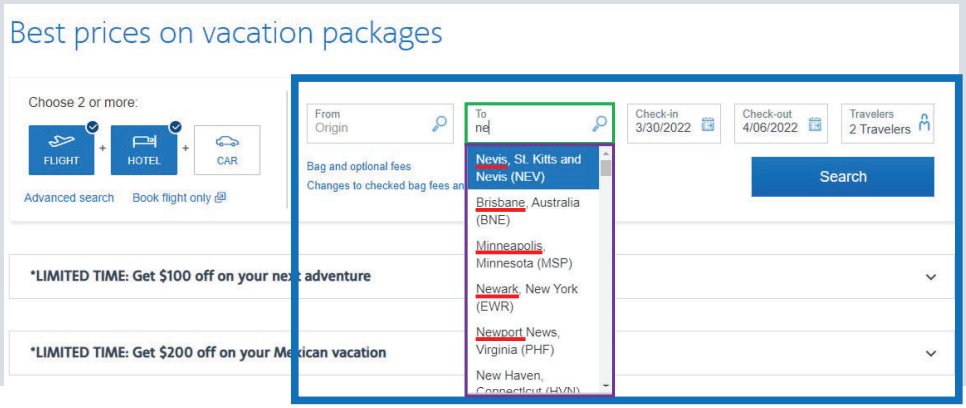
Claim Chart – Claim 1

Claim Feature	Evidence
1[a] a database configured to store information about events in the computer database system; and	The Accused Instrumentalities include a database configured to store information about events in the compute database system. For example, the Accused Instrumentalities store information regarding bookings of flights:

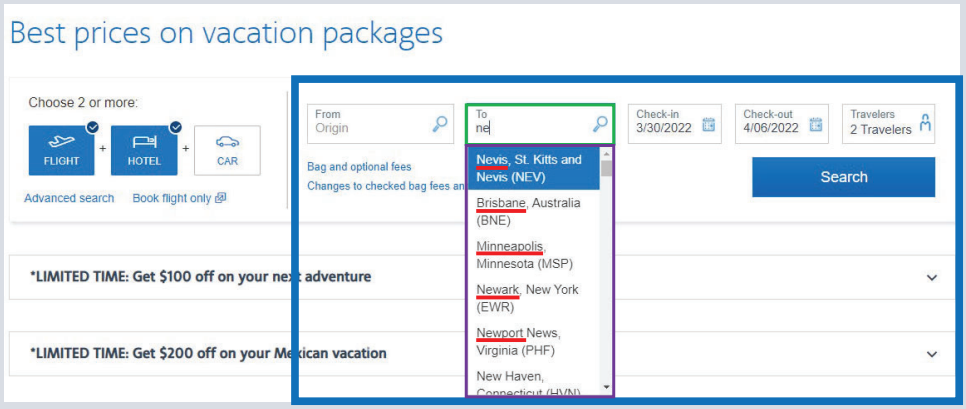


[5] DFW to LGA Flights

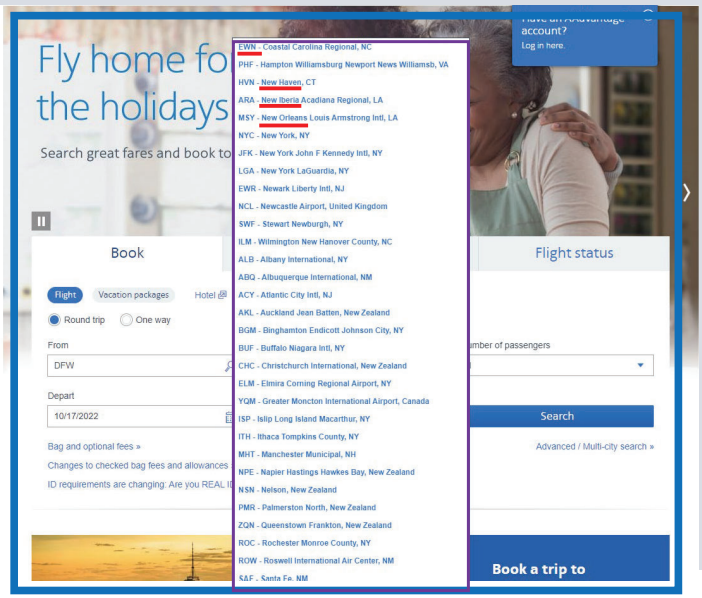
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] a query reconstruction server in data communication with the database and operative to receive a partial query submitted at a remote user client system by a user seeking search results matching the submitted partial query and, in response to the received partial query, determine a full query based on (i) the received partial query, and (ii) information stored in the database about queries previously-submitted by users,	The Accused Instrumentalities include a query reconstruction server in data communication with the database and operative to receive a partial query submitted at a remote user client system by a user seeking search results matching the submitted partial query and, in response to the received partial query, determine a full query based on (i) the received partial query, and (ii) information stored in the database about queries previously-submitted by users. For example, the Accused Instrumentalities can determine a full query (e.g., city name, airport, etc.) in response to receiving a partial query (e.g., partial word, characters, city name, etc.), such as can be submitted by a user at a remote user client system. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

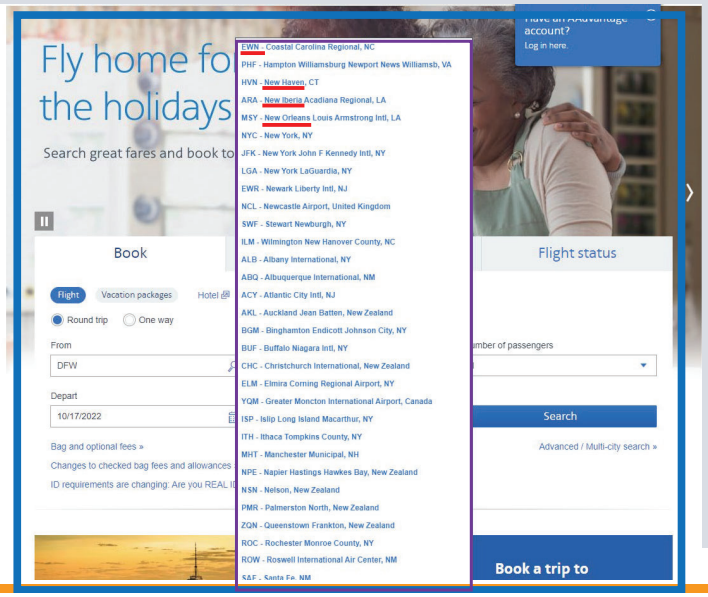
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] a query reconstruction server in data communication with the database and operative to receive a partial query submitted at a remote user client system by a user seeking search results matching the submitted partial query and, in response to the received partial query, determine a full query based on (i) the received partial query, and (ii) information stored in the database about queries previously-submitted by users,	The query reconstruction server of the Accused Instrumentalities determines a full query based at least partly on the received partial query and information stored in the database about queries previously-submitted by users. For example, Accused Instrumentalities determine a full query based on popular searches and/or destination, e.g., searches and/or destinations corresponding to queries previously submitted by users. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

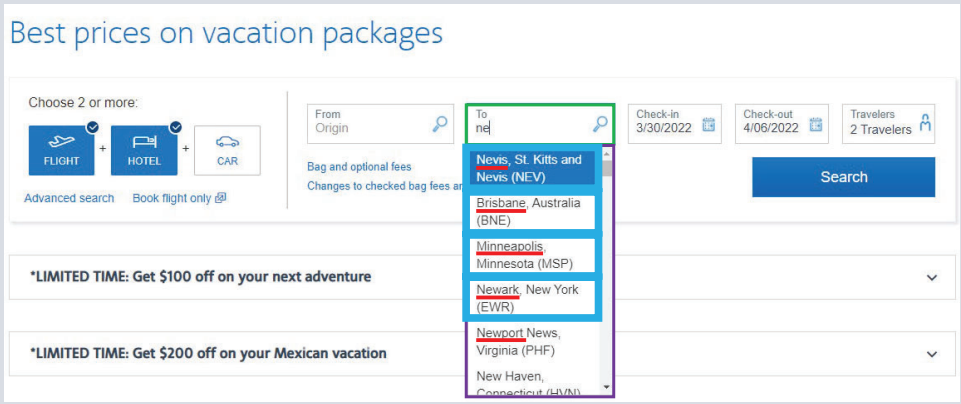
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] a query reconstruction server in data communication with the database and operative to receive a partial query submitted at a remote user client system by a user seeking search results matching the submitted partial query and, in response to the received partial query, determine a full query based on (i) the received partial query, and (ii) information stored in the database about queries previously-submitted by users,	For example, the Accused Instrumentalities can determine a full query (e.g., city name, airport, etc.) in response to receiving a partial query (e.g., partial word, characters, city name, etc.), such as can be submitted by a user at a remote user client system. Here, "new" is typed into the search bar and hidden in the graphic. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

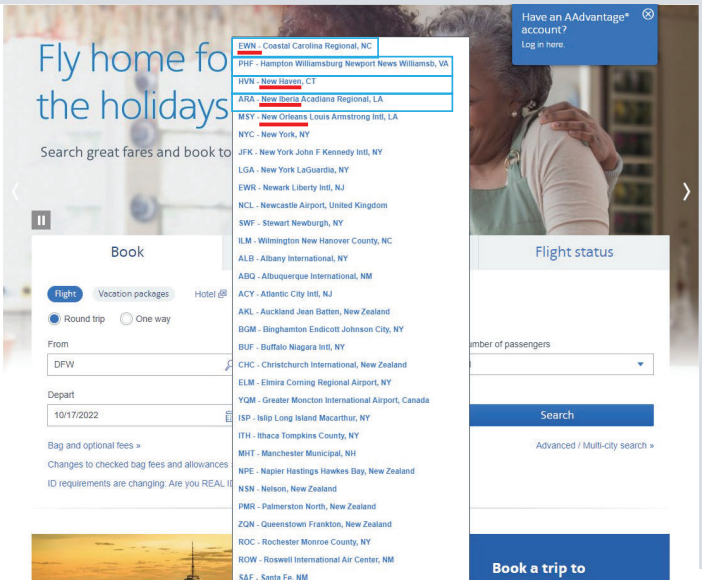
Claim Chart – Claim 1

Claim Feature	Evidence
1[b] a query reconstruction server in data communication with the database and operative to receive a partial query submitted at a remote user client system by a user seeking search results matching the submitted partial query and, in response to the received partial query, determine a full query based on (i) the received partial query, and (ii) information stored in the database about queries previously-submitted by users,	The query reconstruction server of the Accused Instrumentalities determines a full query based at least partly on the received partial query and information stored in the database about queries previously-submitted by users. For example, Accused Instrumentalities determine a full query based on popular searches and/or destination, e.g., searches and/or destinations corresponding to queries previously submitted by users. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim Chart – Claim 1

Claim Feature	Evidence
1[c] wherein the submitted partial query comprises an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user and the full query is better representative of the entire search query desired by the user.	With respect to the Accused Instrumentalities, the submitted partial query comprises an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user and the full query is bette representative of the entire search query desired by the user. For example, a partial query considered by the query reconstruction server of the Accused Instrumentalities can be an abbreviated or incomplete search query (e.g. characters, partial word, etc.) that is not fully representative of an entire search query (e.g., city name, airport tag etc.). The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

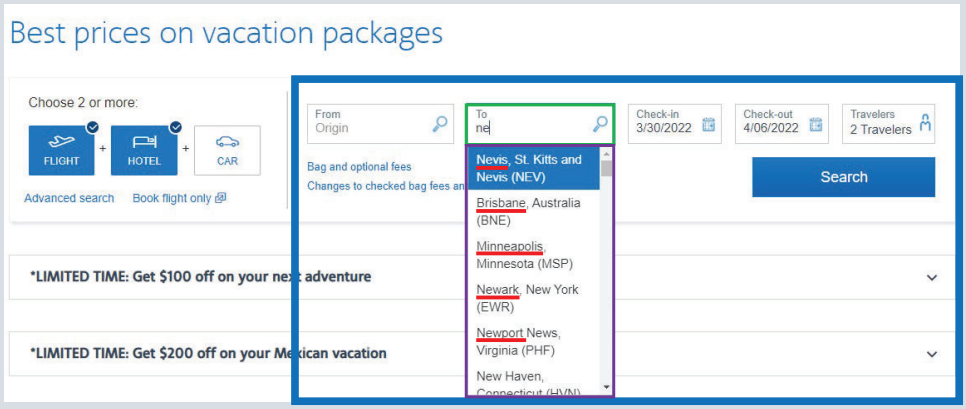
Claim Chart – Claim 1

Claim Feature	Evidence
1[c] wherein the submitted partial query comprises an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user and the full query is better representative of the entire search query desired by the user.	For example, a partial query considered by the query reconstruction server of the Accused Instrumentalities can be ar abbreviated or incomplete search query (e.g., characters, partial word, etc.) that is not fully representative of an entire search query (e.g., city name, airport tag, etc.). The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

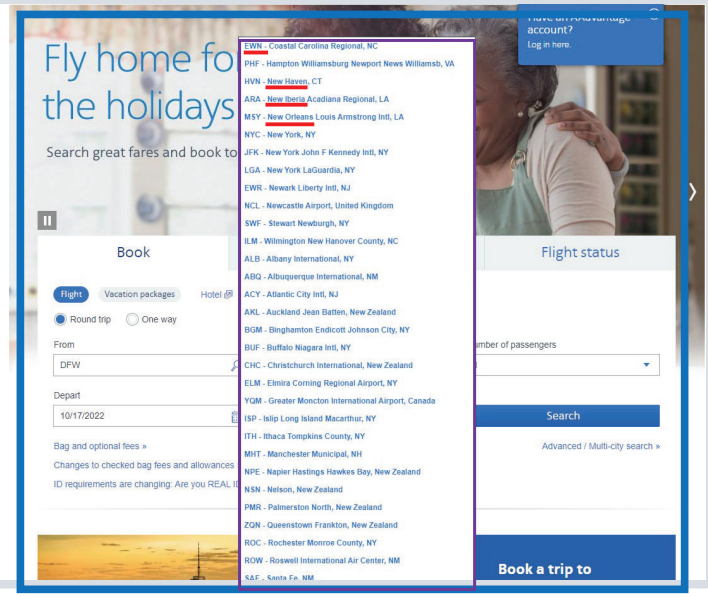
Claim 2

2 The computer database system of claim 1 wherein the query reconstruction server is further operative to determine as the full query the most frequently submitted, previously-submitted query that matches partial words of the partial query.

Claim Chart – Claim 2

Claim Feature	Evidence
2 The computer database system of claim 1 wherein the query reconstruction server is further operative to determine as the full query the most frequently submitted, previously-submitted query that matches partial words of the partial query.	AA infringes Claim 2 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, the query reconstruction server of the Accused Instrumentalities is further operative to determine as the full query the most frequently submitted, previously-submitted query that matches partial words of the partial query. For example, the Accused Instrumentalities determine a full query based on popular, previously-submitted queries that match the partial query, and the particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

Claim Chart – Claim 2

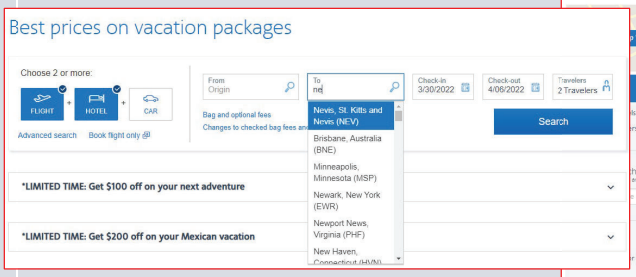
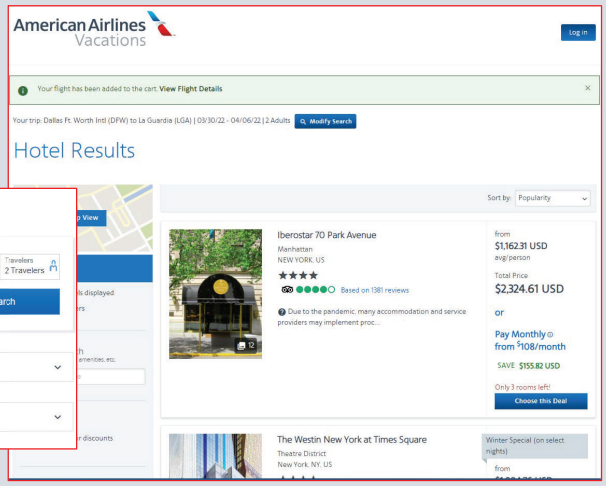
Claim Feature	Evidence
2 The computer database system of claim 1 wherein the query reconstruction server is further operative to determine as the full query the most frequently submitted, previously-submitted query that matches partial words of the partial query.	For example, the Accused Instrumentalities determine a full query based on popular, previously-submitted queries that match the partial query, and the particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim 3

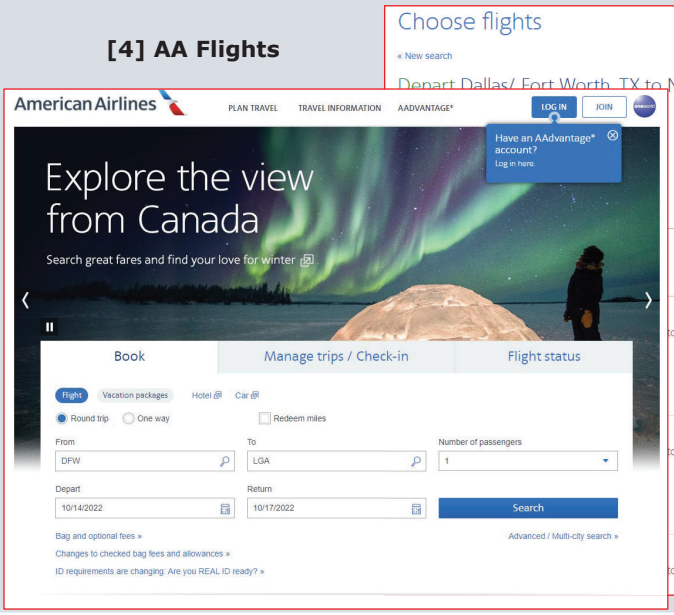
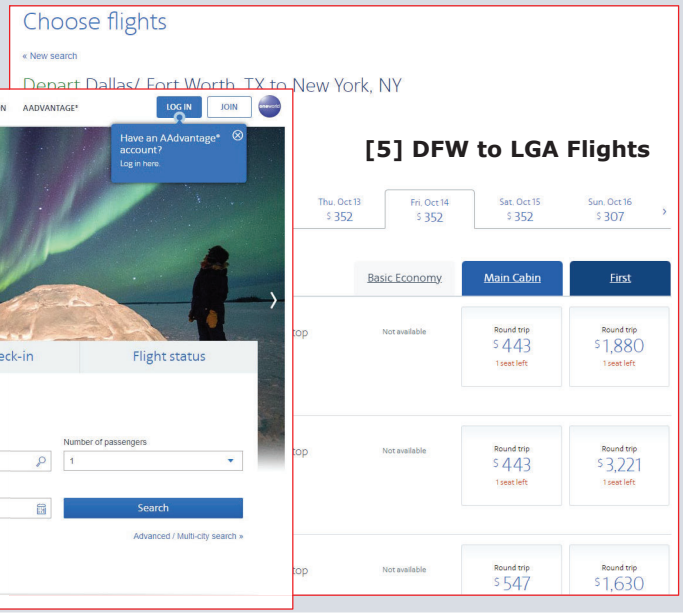
3 The computer database system of claim 2 further comprising:

a web server operative to receive search results produced by a search engine and deliver web pages that include the received search results to a browser of the user.

Claim Chart – Claim 3

Claim Feature	Evidence
3 The computer database system of claim 2 further comprising: a web server operative to receive search results produced by a search engine and deliver web pages that include the received search results to a browser of the user.	AA infringes Claim 3 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claims 1 and 2, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities comprise a web server operative to receive search results produced by a search engine and deliver web pages that include the received search results to a browser of the user. For example, the Accused Instrumentalities deliver web pages with search results. **[1] AA Vacations Homepage**  **[3] DFW to LGA Search** 

Claim Chart – Claim 3

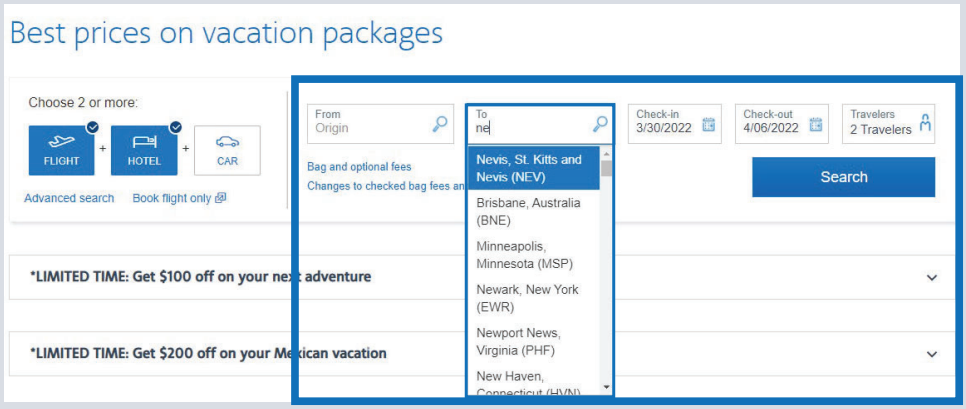
Claim Feature	Evidence
3 The computer database system of claim 2 further comprising: a web server operative to receive search results produced by a search engine and deliver web pages that include the received search results to a browser of the user.	For example, the Accused Instrumentalities deliver web pages with search results. **[4] AA Flights**  **[5] DFW to LGA Flights** 

Claim 4

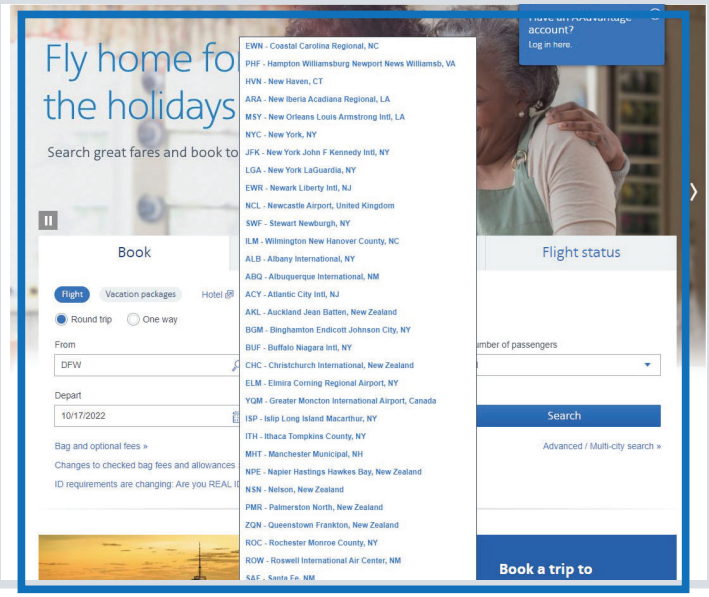
4 The computer database system of claim 1 further comprising:

a query data management server in data communication with the query reconstruction server and the database, the query data management server being operative to provide to the query reconstruction server the information stored in the database about queries previously-submitted by users.

Claim Chart – Claim 4

Claim Feature	Evidence
4 The computer database system of claim 1 further comprising: a query data management server in data communication with the query reconstruction server and the database, the query data management server being operative to provide to the query reconstruction server the information stored in the database about queries previously-submitted by users.	AA infringes Claim 4 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claim 1, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities include a query data management server in data communication with the query reconstruction server and the database, the query data management server being operative to provide to the query reconstruction server the information stored in the database about queries previously-submitted by users. For example, Accused Instrumentalities store previous and/o popular searches, and the particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

Claim Chart – Claim 4

Claim Feature	Evidence
4 The computer database system of claim 1 further comprising: a query data management server in data communication with the query reconstruction server and the database, the query data management server being operative to provide to the query reconstruction server the information stored in the database about queries previously-submitted by users.	For example, Accused Instrumentalities store previous and/or popular searches, and the particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim 5

5[pre] The computer database system of claim 4 wherein the query data management server is further configured to:

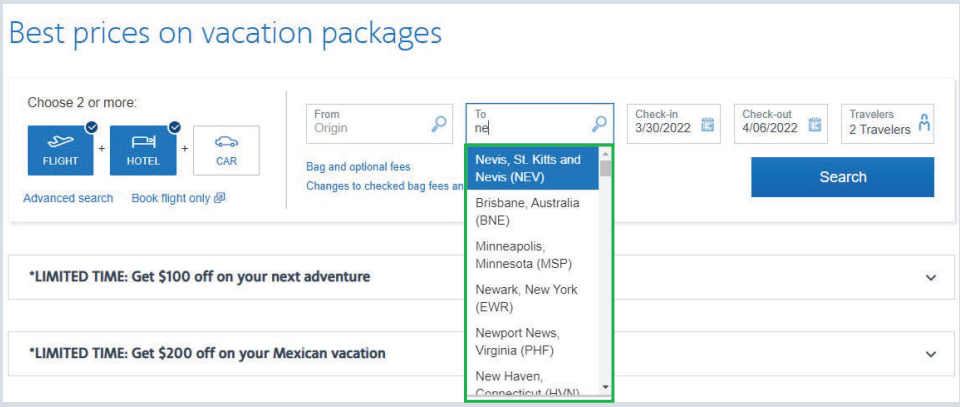
5[a] retrieve from the database historical query data relating to a plurality of previously submitted search queries;

5[b] build a query index based on the historical query data;

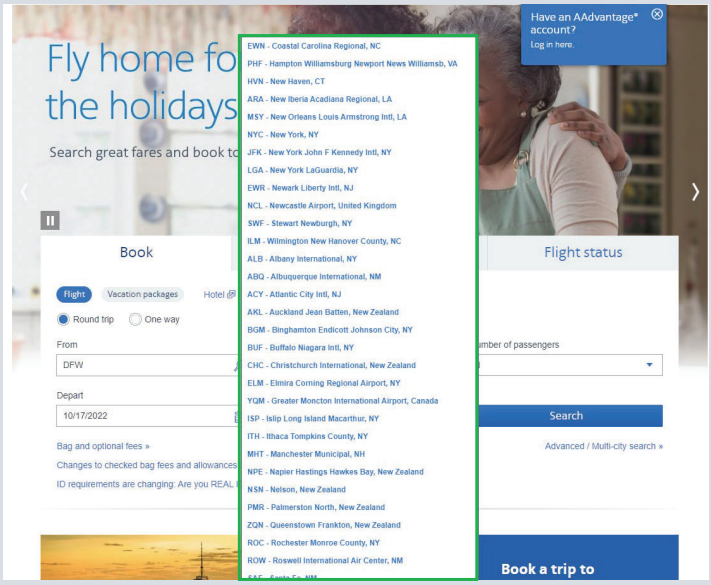
5[c] store the query index in the database; and

5[d] provide the query index to the query reconstruction server.

Claim Chart – Claim 5

Claim Feature	Evidence
5[pre] The computer database system of claim 4 wherein the query data management server is further configured to: **5[a]** retrieve from the database historical query data relating to a plurality of previously submitted search queries;	AA infringes Claim 5 of the '317 Patent. The Accused Instrumentalities satisfy each and every element of Claims 1 and 4, the discussion of which is incorporated herein by reference. Furthermore, the query data management server of the Accused Instrumentalities is configured to retrieve from the database historical query data relating to a plurality of previously submitted search queries. For example, the Accused Instrumentalities retrieve popular searches, and the particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

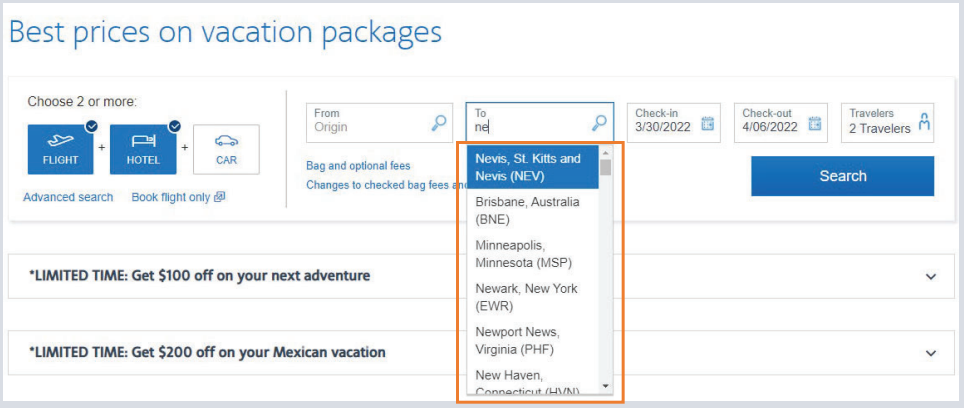
Claim Chart – Claim 5

Claim Feature	Evidence
5[pre] The computer database system of claim 4 wherein the query data management server is further configured to: **5[a]** retrieve from the database historical query data relating to a plurality of previously submitted search queries;	For example, the Accused Instrumentalities retrieve previous searches, and the particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim Chart – Claim 5

Claim Feature	Evidence
5[b] build a query index based on the historical query data;	The Accused Instrumentalities build a query index based on the historical query data. For example, the Accused Instrumentalities build indexes based on historical query data that can, e.g., be queried. The particular details are within AA's possession, custody, and control.

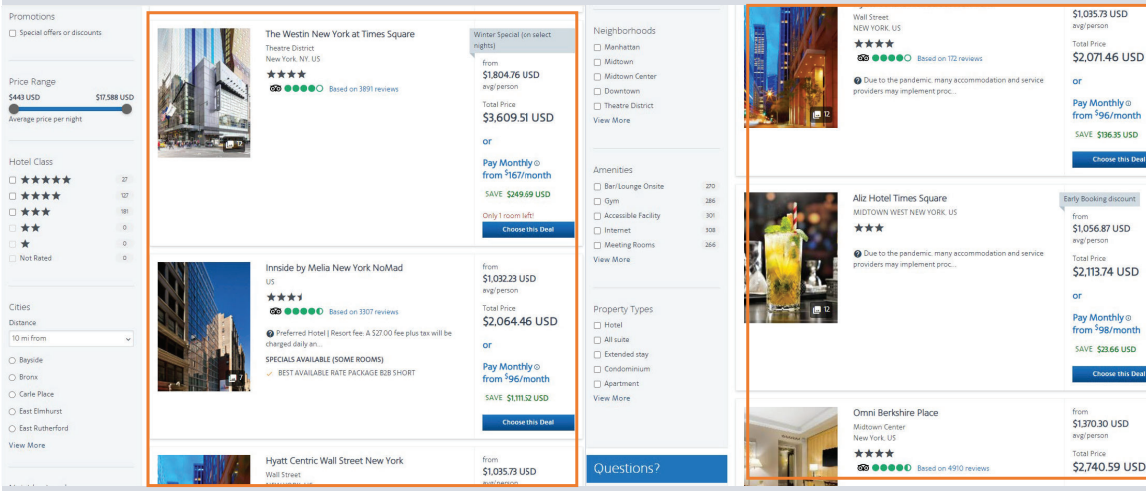
[1] AA Vacations Homepage



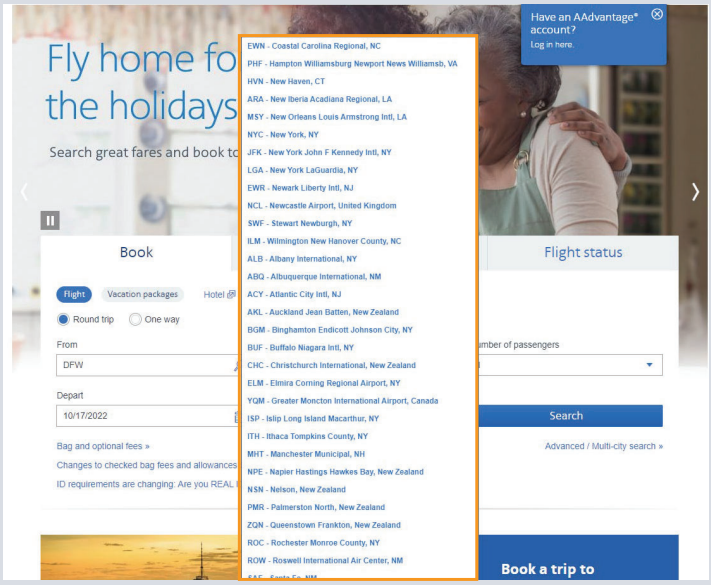
Claim Chart – Claim 5

Claim Feature	Evidence
5[b] build a query index based on the historical query data;	For example, the Accused Instrumentalities build indexes based on historical query data that can, e.g., be queried. The particular details are within AA's possession, custody, and control.

[3] DFW to LGA Search

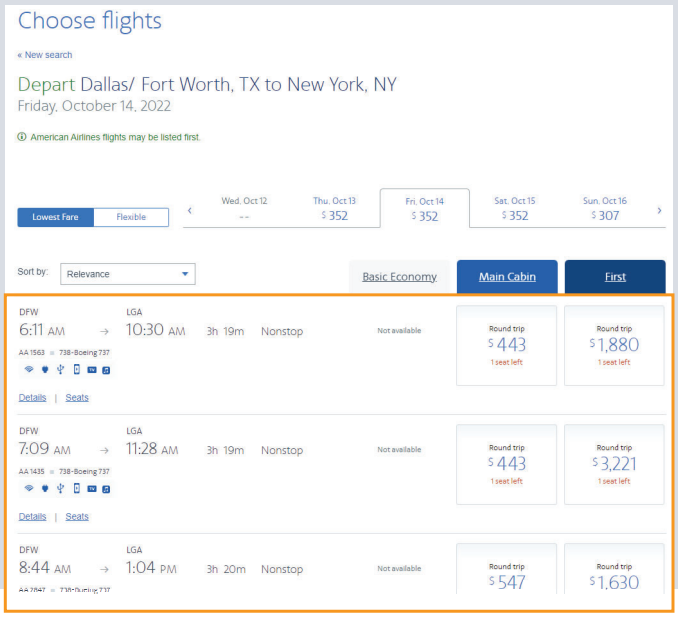


Claim Chart – Claim 5

Claim Feature	Evidence
5[b] build a query index based on the historical query data;	For example, the Accused Instrumentalities build indexes based on historical query data that can, e.g., be queried. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim Chart – Claim 5

Claim Feature	Evidence
5[b] build a query index based on the historical query data;	For example, the Accused Instrumentalities build indexes based on historical query data that can, e.g., be queried. The particular details are within AA's possession, custody, and control.

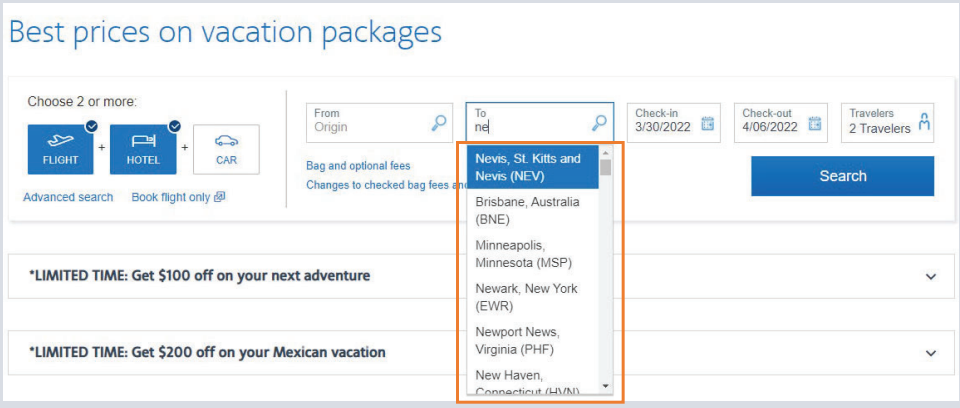


[5] DFW to LGA Flights

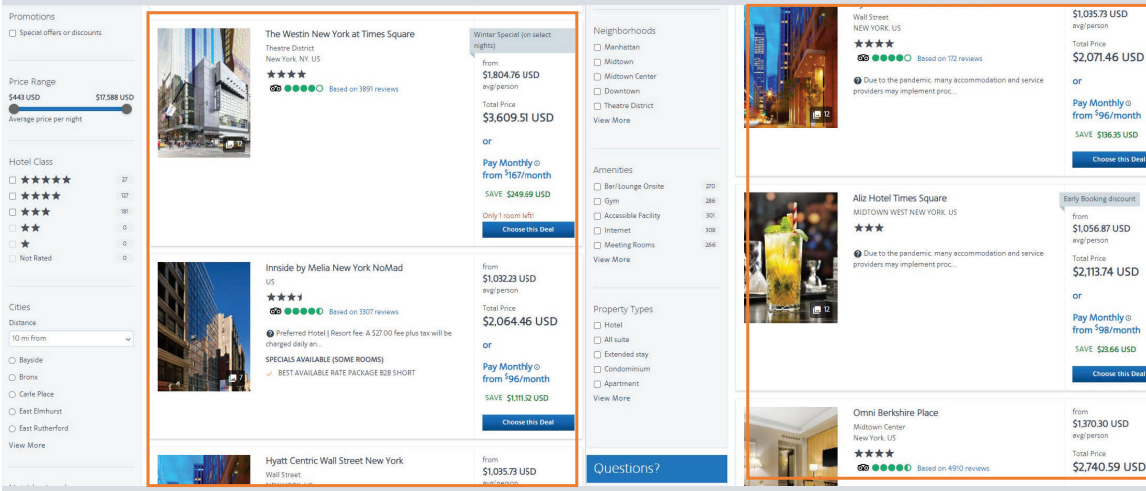
Claim Chart – Claim 5

Claim Feature	Evidence
5[c] store the query index in the database; and	The Accused Instrumentalities store the query index in the database. For example, the Accused Instrumentalities store query indexes in the database such that the indexes can, e.g., be queried. The particular details are within AA's possession, custody, and control.

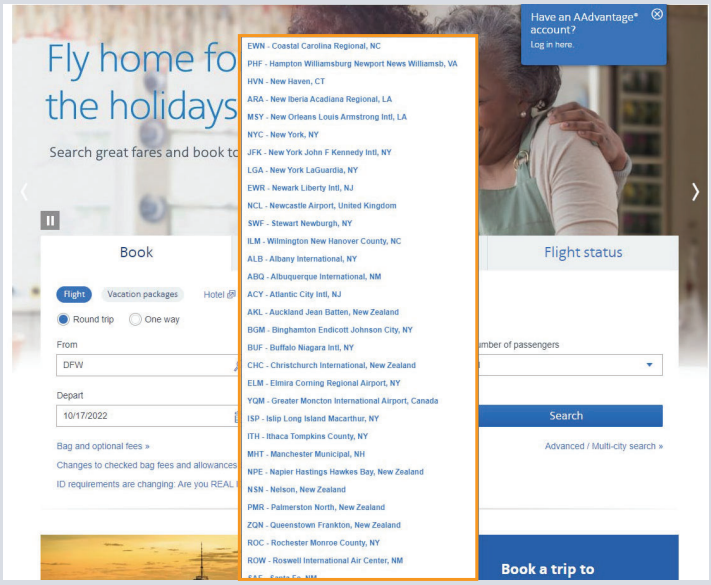
[1] AA Vacations Homepage



Claim Chart – Claim 5

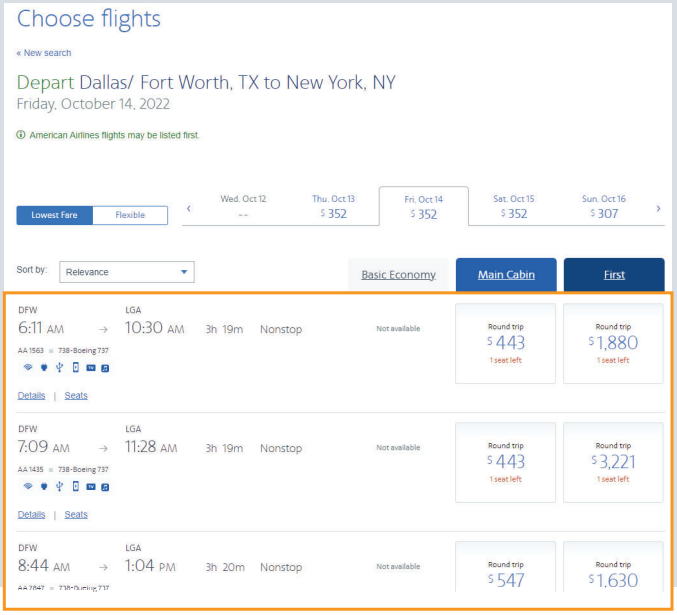
Claim Feature	Evidence
5[c] store the query index in the database; and	For example, upon information and belief, the Accused Instrumentalities store query indexes in the database such that the indexes can, e.g., be queried. The particular details are within AA's possession, custody, and control. **[3] DFW to LGA Search** 

Claim Chart – Claim 5

Claim Feature	Evidence
5[c] store the query index in the database; and	For example, upon information and belief, the Accused Instrumentalities store query indexes in the database such that the indexes can, e.g., be queried. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

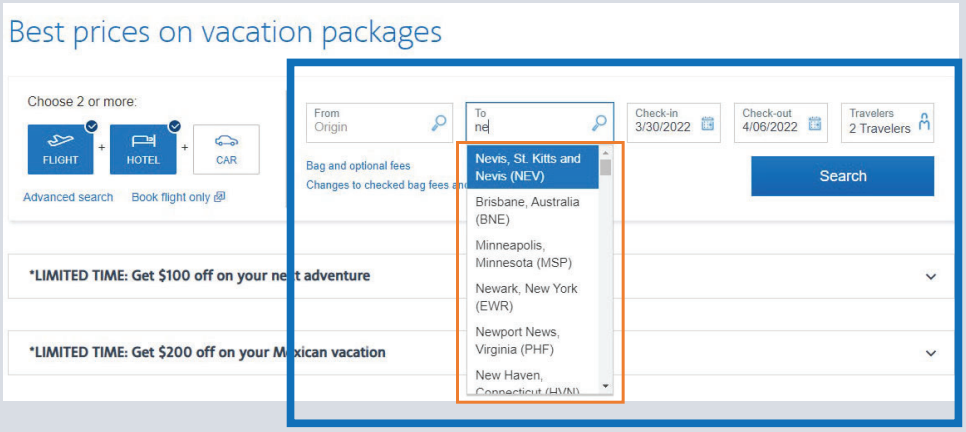
Claim Chart – Claim 5

Claim Feature	Evidence
5[c] store the query index in the database; and	For example, upon information and belief, the Accused Instrumentalities store query indexes in the database such that the indexes can, e.g., be queried. The particular details are within AA's possession, custody, and control.

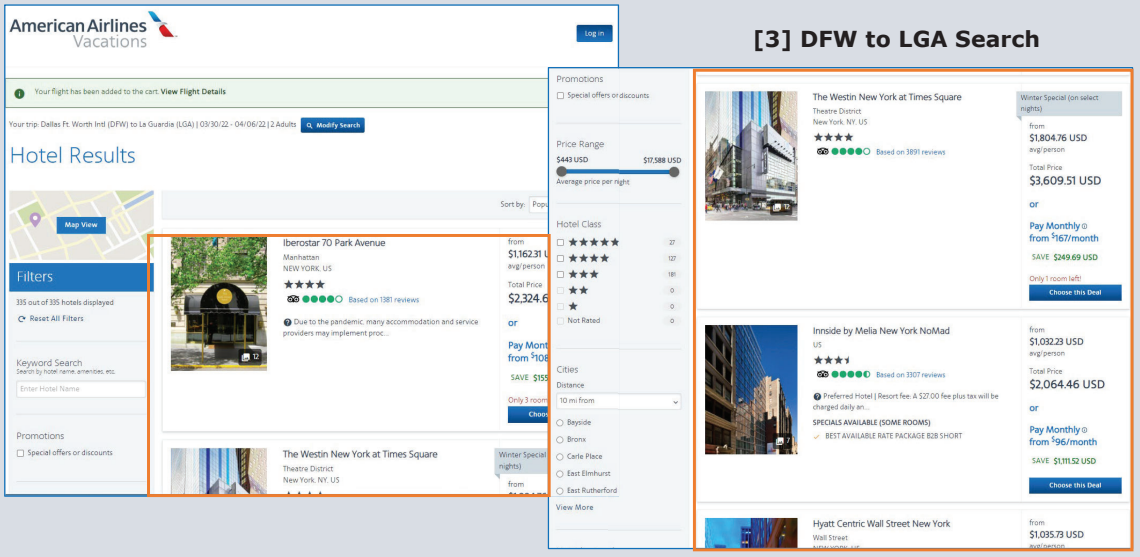


[5] DFW to LGA Flights

Claim Chart – Claim 5

Claim Feature	Evidence
5[d] provide the query index to the query reconstruction server.	The Accused Instrumentalities provide the query index to the query reconstruction server. For example, the query reconstruction server is provided the query index, such as to, e.g., facilitating the determination of the full query. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

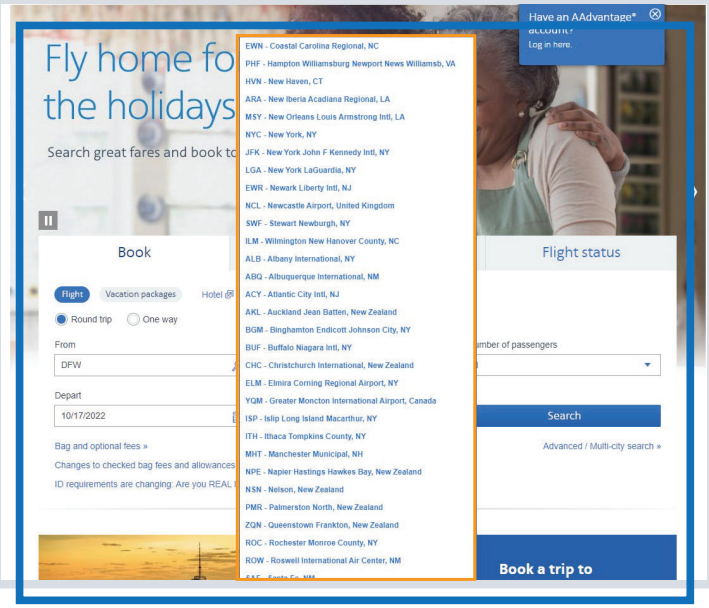
Claim Chart – Claim 5

Claim Feature	Evidence
5[d] provide the query index to the query reconstruction server.	The Accused Instrumentalities provide the query index to the query reconstruction server. For example, upon information and belief, the query reconstruction server is provided the query index, such as to, e.g., facilitating the determination of the fully query. The particular details are within AA's possession, custody, and control. 

Claim Chart – Claim 5

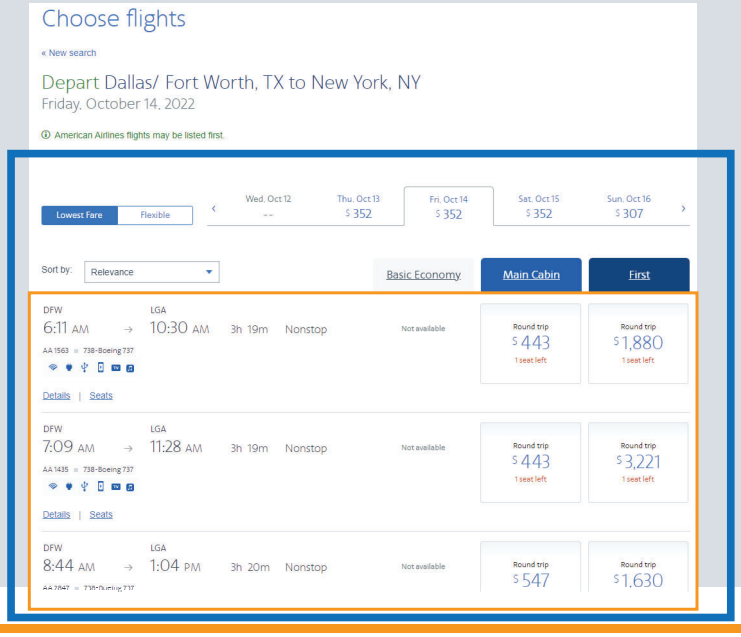
Claim Feature	Evidence
5[d] provide the query index to the query reconstruction server.	For example, upon information and belief, the query reconstruction server is provided the query index, such as to, e.g. facilitating the determination of the fully query. The particular details are within AA's possession, custody, and control.

[4] AA Flights



Claim Chart – Claim 5

Claim Feature	Evidence
5[d] provide the query index to the query reconstruction server.	For example, upon information and belief, the query reconstruction server is provided the query index, such as to, e.g. facilitating the determination of the fully query. The particular details are within AA's possession, custody, and control.



[5] DFW to LGA Flights

Claim 8

8[pre] A computer implemented method for a computer database system, the method comprising:

8[a] storing in memory data about previously received queries at the computer database system;

8[b] at a server, retrieving selected data from the memory including query data representative of previously submitted queries and information based on frequency counts for the respective previously submitted queries;

8[c] at the server, constructing indexes using the retrieved selected data;

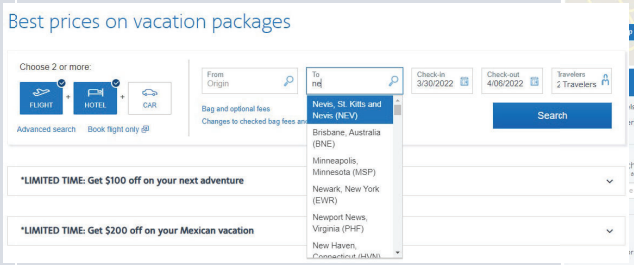
8[d] receiving at the server a partial query submitted by a user at a remote user client system, the submitted partial query including an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user; and

8[e] at the server, using the indexes to determine a full query corresponding to the submitted partial query, the full query being better representative of the entire search query desired by the user.

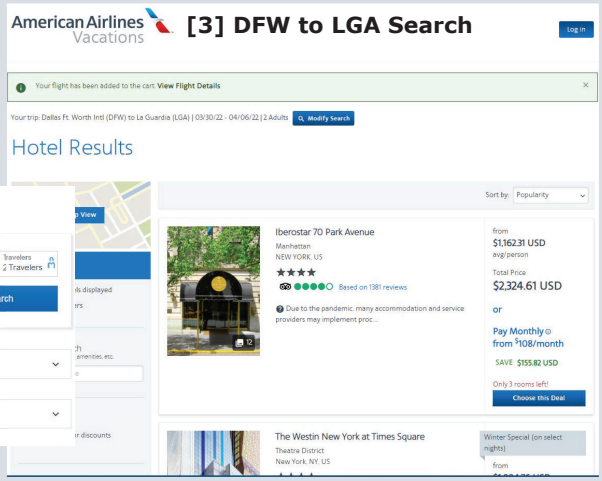
Claim Chart – Claim 8

Claim Feature	Evidence
8[pre] A computer implemented method for a computer database system, the method comprising:	AA infringes Claim 8 of the '317 Patent. The Accused Instrumentalities perform a computer implemented method for a computer database system. For example, the Accused Instrumentalities comprise a computer database system fo providing search results to a user in response to user submissions over a data network. The Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/or software), or othe instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s):

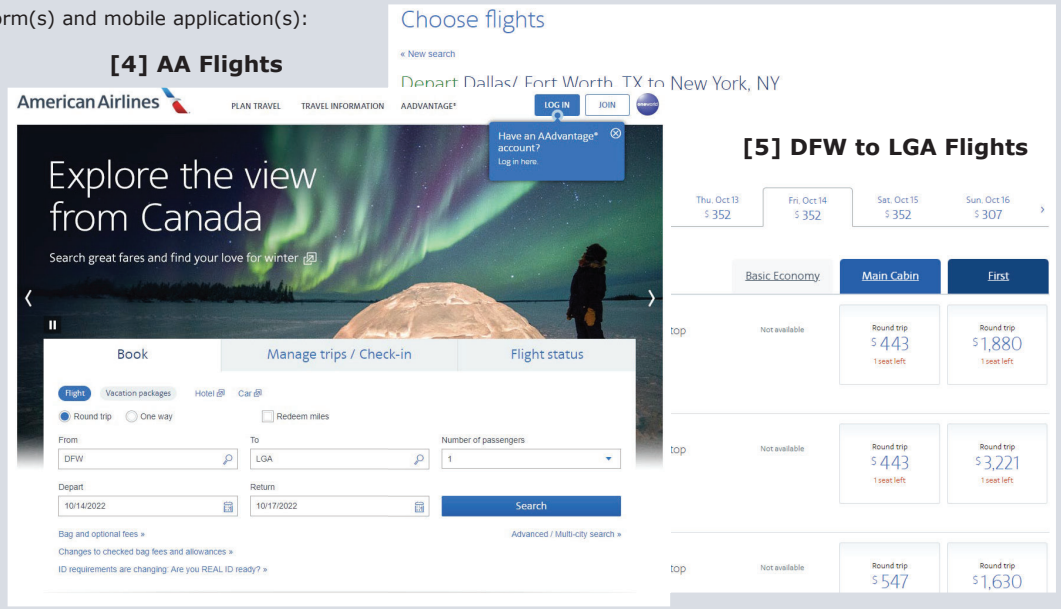
[1] AA Vacations Homepage



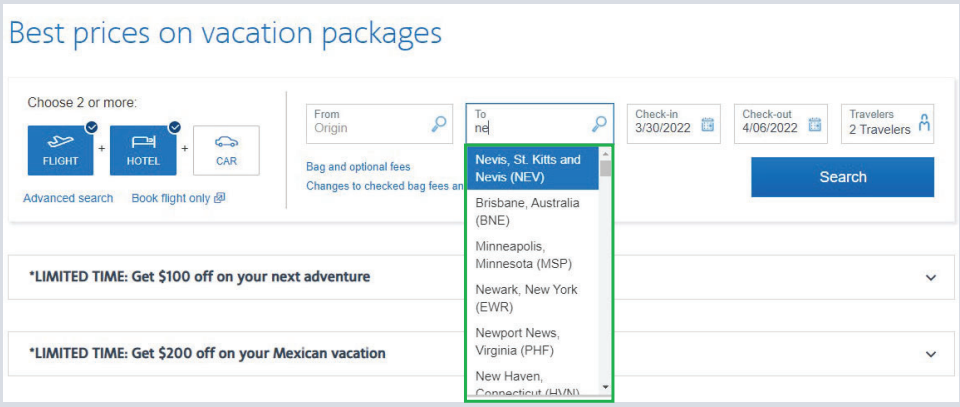
[3] DFW to LGA Search



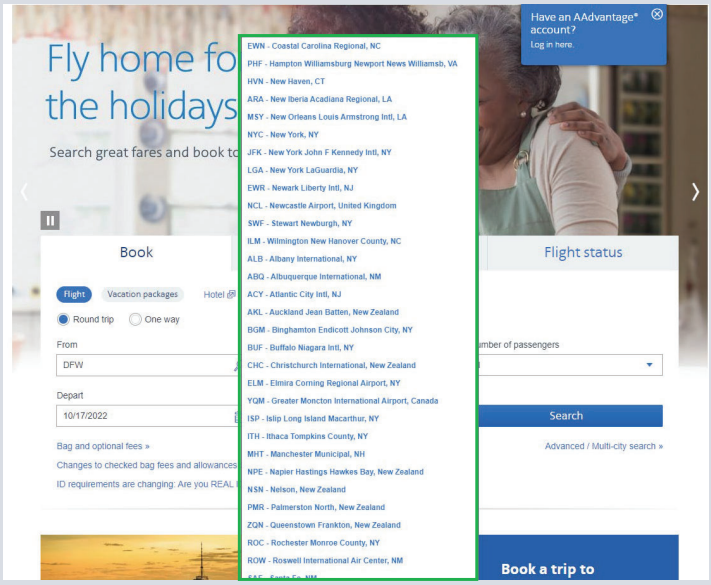
Claim Chart – Claim 8

Claim Feature	Evidence
8[pre] A computer implemented method for a computer database system, the method comprising:	The Accused Instrumentalities include systems, services, processes, methods, acts, components (hardware and/o software), or other instrumentalities associated with searches or queries performed in connection with AA's web platform(s) and mobile application(s): **[4] AA Flights** 

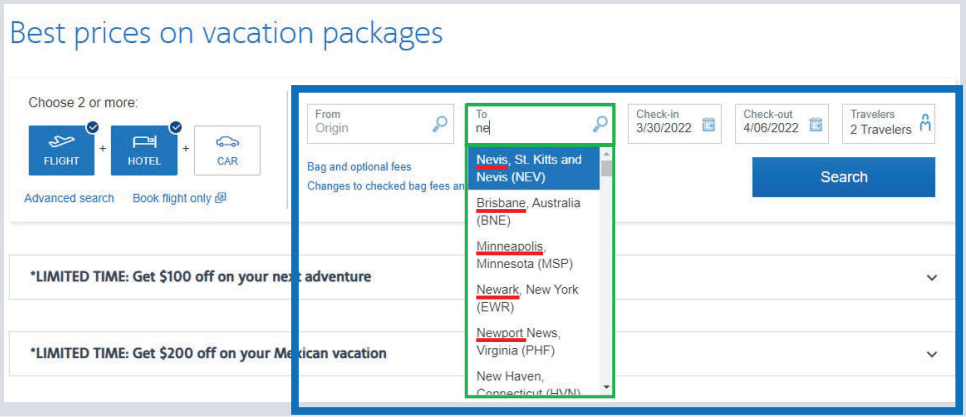
Claim Chart – Claim 8

Claim Feature	Evidence
8[a] storing in memory data about previously received queries at the computer database system;	The Accused Instrumentalities store in memory data about previously received queries at the computer database system. For example, the Accused Instrumentalities store popular searches of users. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

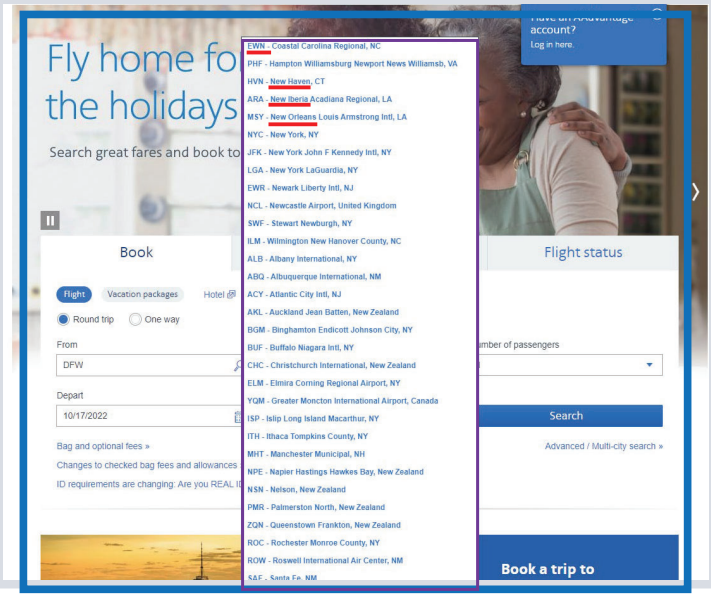
Claim Chart – Claim 8

Claim Feature	Evidence
8[a] storing in memory data about previously received queries at the computer database system;	For example, the Accused Instrumentalities store previous searches of users. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

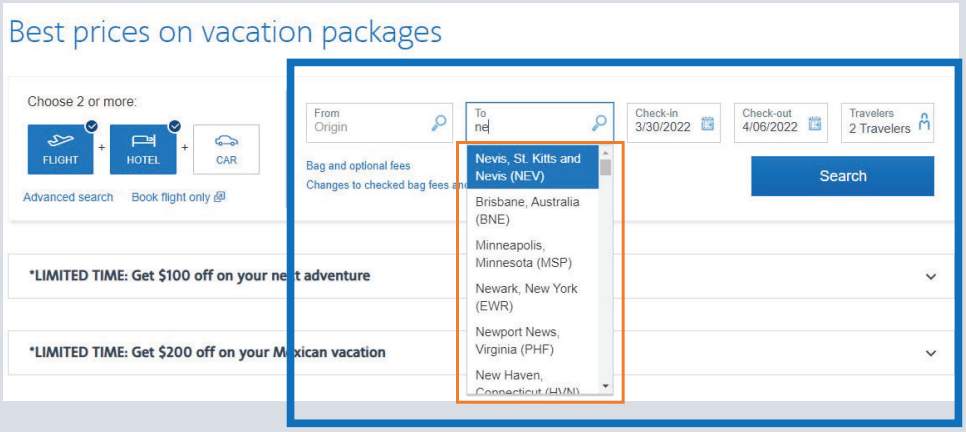
Claim Chart – Claim 8

Claim Feature	Evidence
8[b] at a server, retrieving selected data from the memory including query data representative of previously submitted queries and information based on frequency counts for the respective previously submitted queries;	The Accused Instrumentalities, at a server, retrieve selected data from the memory including query data representative of previously submitted queries and information based on frequency counts for the respective previously submitted queries. For example, Accused Instrumentalities retrieve previously submitted queries and retrieve frequency counts of previously submitted queries, such as to determine popular queries. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

Claim Chart – Claim 8

Claim Feature	Evidence
8[b] at a server, retrieving selected data from the memory including query data representative of previously submitted queries and information based on frequency counts for the respective previously submitted queries;	For example, Accused Instrumentalities retrieve previously submitted queries and retrieve frequency counts o previously submitted queries, such as to determine popular queries. The particular details are within AA's possession custody, and control. **[4] AA Flights** 

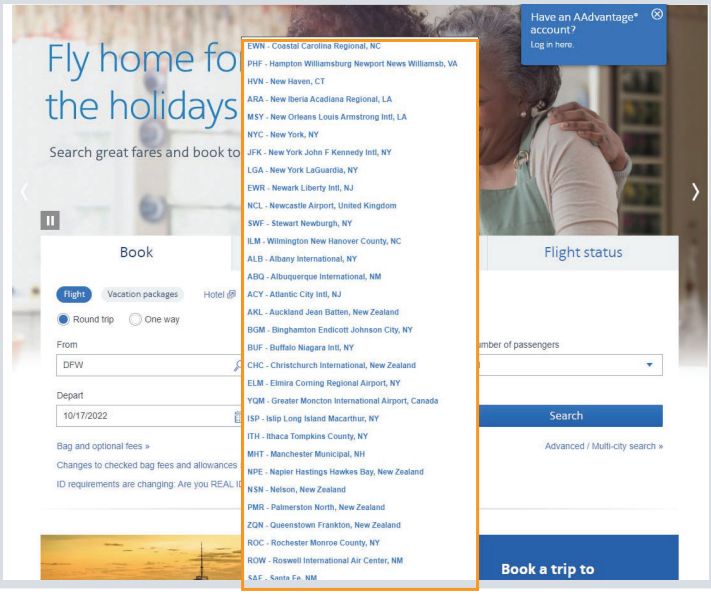
Claim Chart – Claim 8

Claim Feature	Evidence
8[c] at the server, constructing indexes using the retrieved selected data;	The Accused Instrumentalities, at the server, construct indexes using the retrieved selected data. For example, the Accused Instrumentalities construct indexes that can be searched using the retrieved data. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

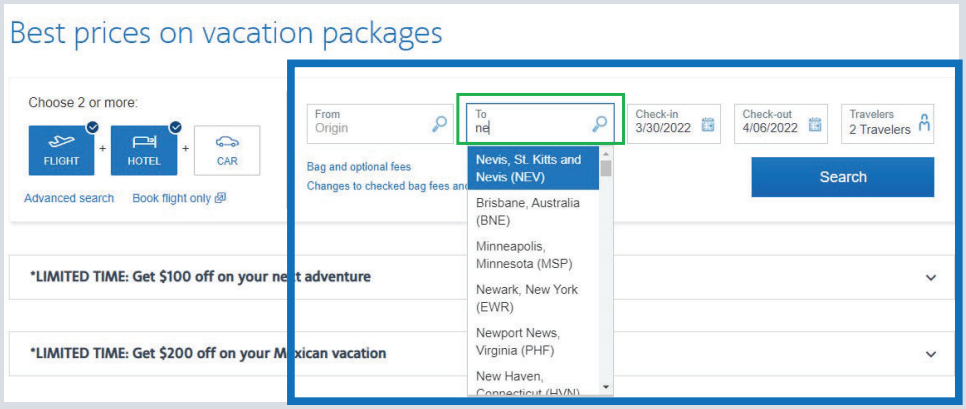
Claim Chart – Claim 8

Claim Feature	Evidence
8[c] at the server, constructing indexes using the retrieved selected data;	For example, the Accused Instrumentalities construct indexes that can be searched using the retrieved data. The particular details are within AA's possession, custody, and control.

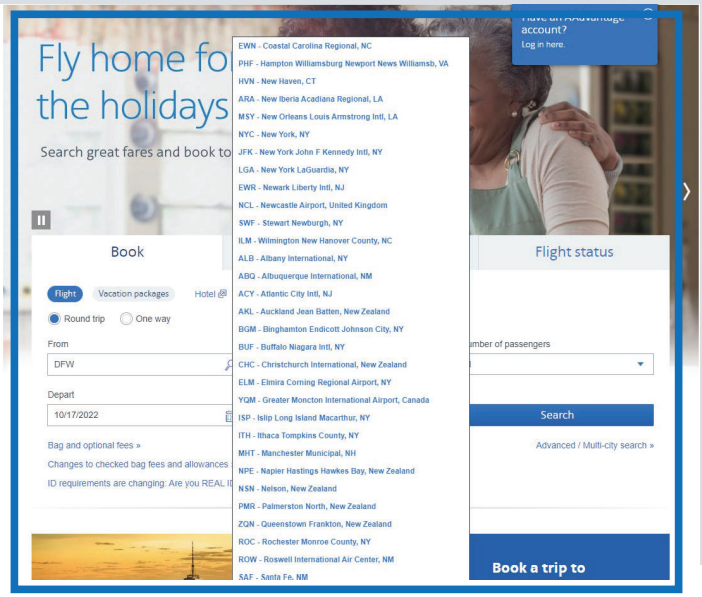
[4] AA Flights



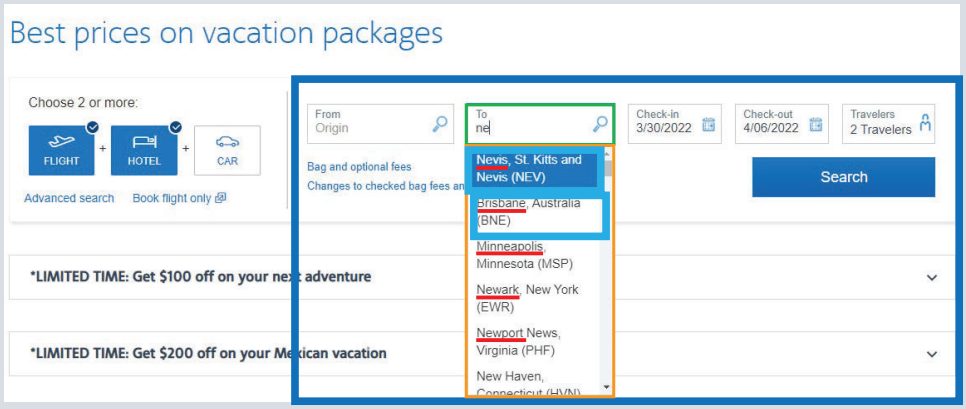
Claim Chart – Claim 8

Claim Feature	Evidence
8[d] receiving at the server a partial query submitted by a user at a remote user client system, the submitted partial query including an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user; and	The Accused Instrumentalities receive at the server a partial query submitted by a user at a remote user client system the submitted partial query including an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user. For example, a partial query considered by the query reconstruction server o the Accused Instrumentalities can be an abbreviated or incomplete search query (e.g., characters, partial word, etc.) that is not fully representative of an entire search query (e.g., city name, city/country name, etc.). The particula details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

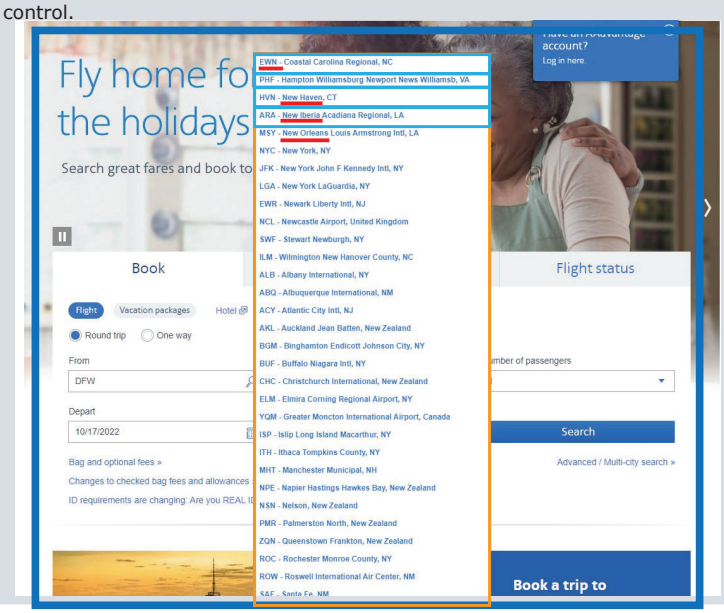
Claim Chart – Claim 8

Claim Feature	Evidence
8[d] receiving at the server a partial query submitted by a user at a remote user client system, the submitted partial query including an abbreviated or incomplete search query which is not fully representative of an entire search query desired by the user; and	For example, a partial query considered by the query reconstruction server of the Accused Instrumentalities can be an abbreviated or incomplete search query (e.g., characters, partial word, etc.) that is not fully representative of an entire search query (e.g., city name, city/country name, etc.). The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim Chart – Claim 8

Claim Feature	Evidence
8[e] at the server, using the indexes to determine a full query corresponding to the submitted partial query, the full query being better representative of the entire search query desired by the user.	The Accused Instrumentalities, at the server, use the indexes to determine a full query corresponding to the submitted partial query, the full query being better representative of the entire search query desired by the user. For example, the server uses the indexes and the partial query to determine a full query, and the full query can be better representative of the entire search query (e.g., city name, city/country name, etc.). The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

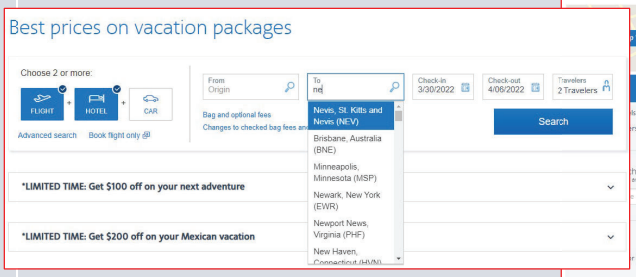
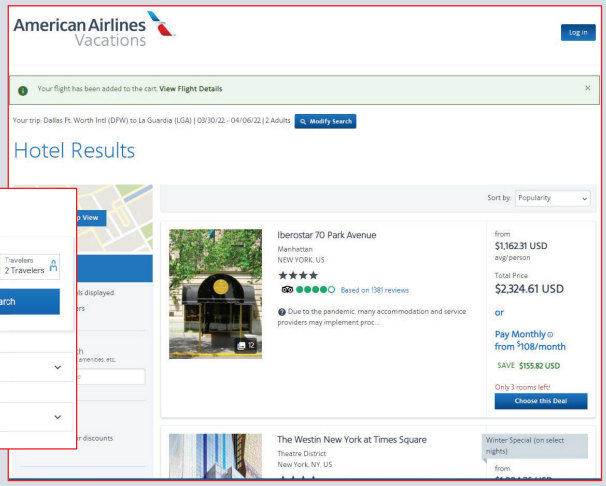
Claim Chart – Claim 8

Claim Feature	Evidence
8[e] at the server, using the indexes to determine a full query corresponding to the submitted partial query, the full query being better representative of the entire search query desired by the user.	For example, the server uses the indexes and the partial query to determine a full query, and the full query can be better representative of the entire search query (e.g., city name, city/country name, etc.). The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim 9

9 The method of claim 8 further comprising communicating the full query from the server to a search engine for subsequent processing of the full query.

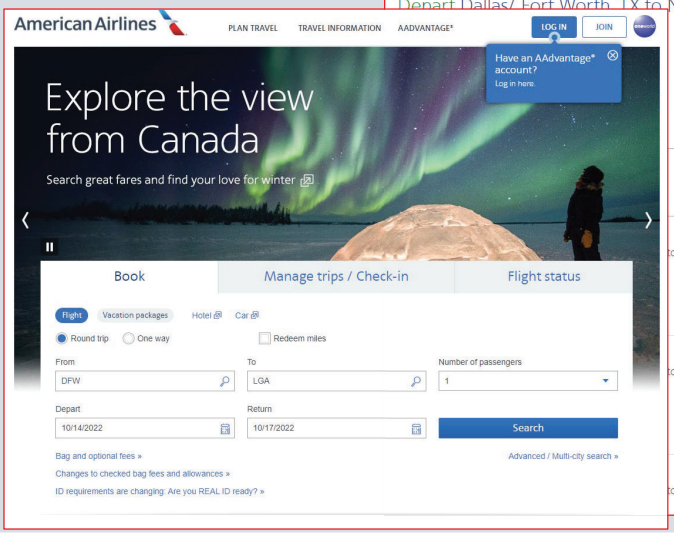
Claim Chart – Claim 9

Claim Feature	Evidence
9 The method of claim 8 further comprising communicating the full query from the server to a search engine for subsequent processing of the full query.	AA infringes Claim 9 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claim 8, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities communicate the full query from the server to a search engine for subsequent processing of the full query. For example, the Accused Instrumentalities process the full query to, e.g., generate a search result based on the full query. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage**  **[3] DFW to LGA Search** 

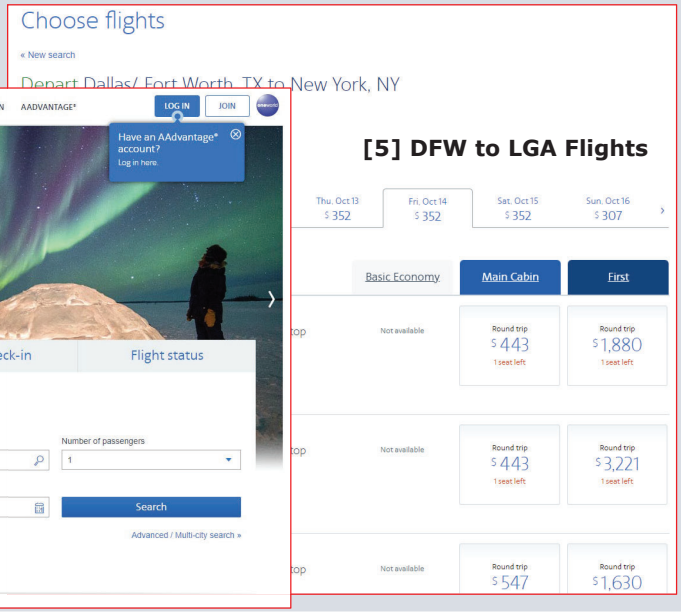
Claim Chart – Claim 9

Claim Feature	Evidence
9 The method of claim 8 further comprising communicating the full query from the server to a search engine for subsequent processing of the full query.	For example, the Accused Instrumentalities process the full query to, e.g., generate a search result based on the ful query. The particular details are within AA's possession, custody, and control.

[4] AA Flights



[5] DFW to LGA Flights



Claim 10

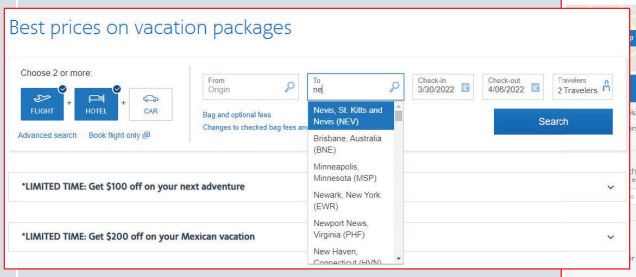
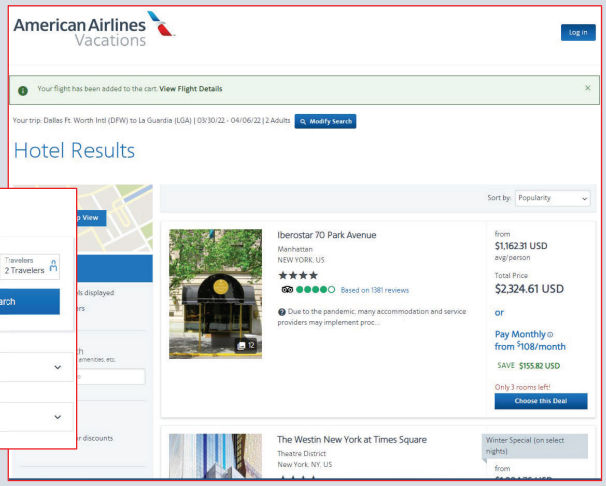
10[pre] The method of claim 8 further comprising;

10[a] submitting the full query from the server to a search engine;

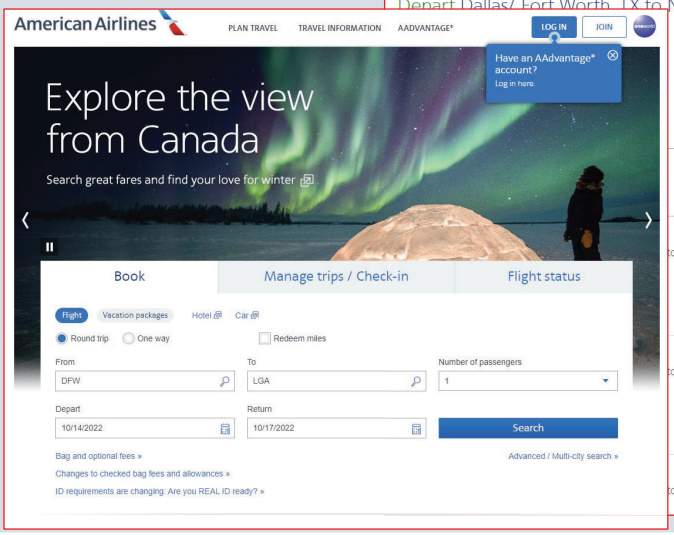
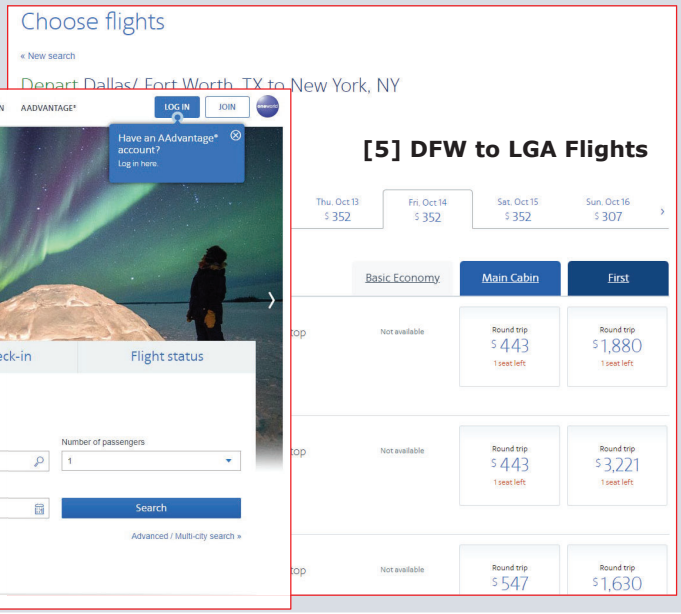
10[b] receiving search results at the server from the search engine; and

10[c] communicating a web page including at least some of the search results from the server to the user.

Claim Chart – Claim 10

Claim Feature	Evidence
10[pre] The method of claim 8 further comprising; **10[a]** submitting the full query from the server to a search engine;	AA infringes Claim 10 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claim 8, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities submit the full query from the server to a search engine. For example, the Accused Instrumentalities submit the full query to a search engine to generate a result. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage**  **[3] DFW to LGA Search** 

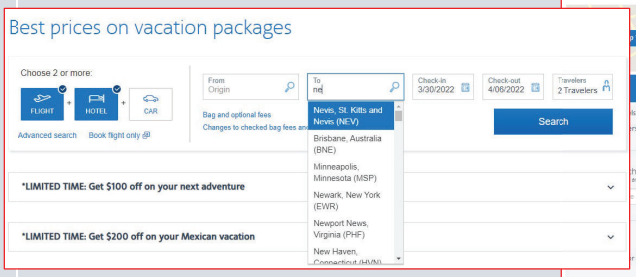
Claim Chart – Claim 10

Claim Feature	Evidence
10[pre] The method of claim 8 further comprising; **10[a]** submitting the full query from the server to a search engine;	For example, the Accused Instrumentalities submit the full query to a search engine to generate a result. The particular details are within AA's possession, custody, and control. **[4] AA Flights**  **[5] DFW to LGA Flights** 

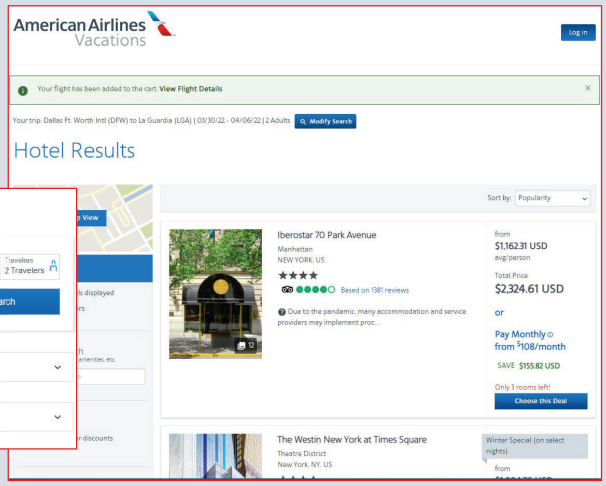
Claim Chart – Claim 10

Claim Feature	Evidence
10[b] receiving search results at the server from the search engine; and	The Accused Instrumentalities receive search results at the server from the search engine. For example, the Accused Instrumentalities receive search results to display to a user. The particular details are within AA's possession, custody and control.

[1] AA Vacations Homepage



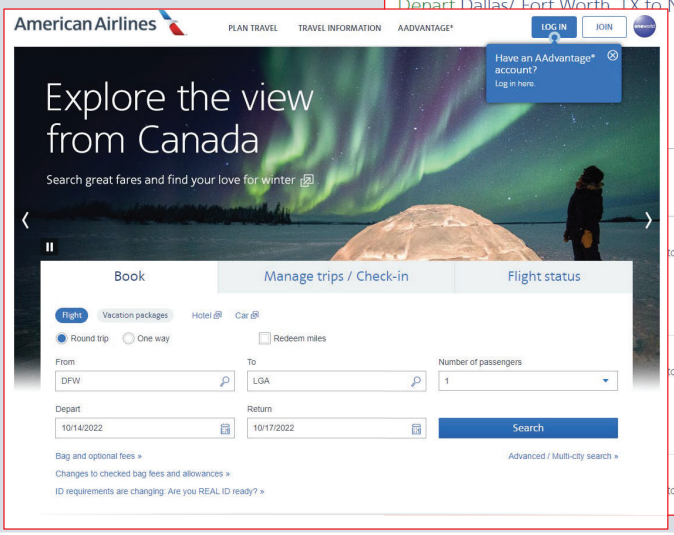
[3] DFW to LGA Search



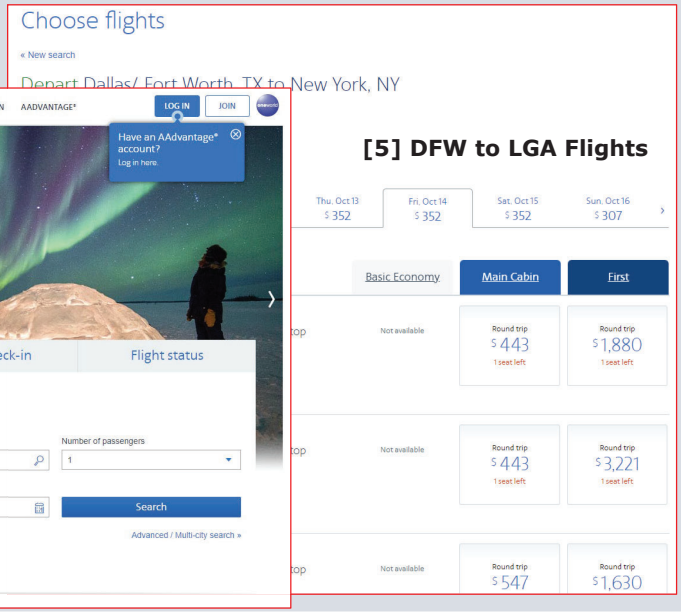
Claim Chart – Claim 10

Claim Feature	Evidence
10[b] receiving search results at the server from the search engine; and	For example, the Accused Instrumentalities receive search results to display to a user. The particular details are within AA's possession, custody, and control.

[4] AA Flights



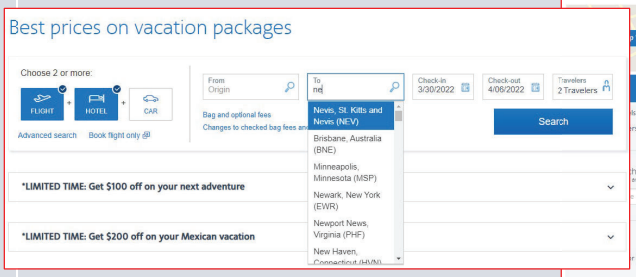
[5] DFW to LGA Flights



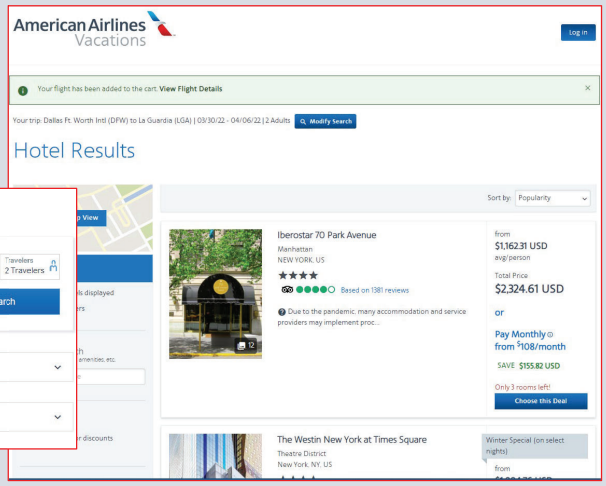
Claim Chart – Claim 10

Claim Feature	Evidence
10[c] communicating a web page including at least some of the search results from the server to the user.	The Accused Instrumentalities communicate a web page including at least some of the search results from the server to the user. For example, the Accused Instrumentalities display search result to a user on a webpage. The particular details are within AA's possession, custody, and control.

[1] AA Vacations Homepage



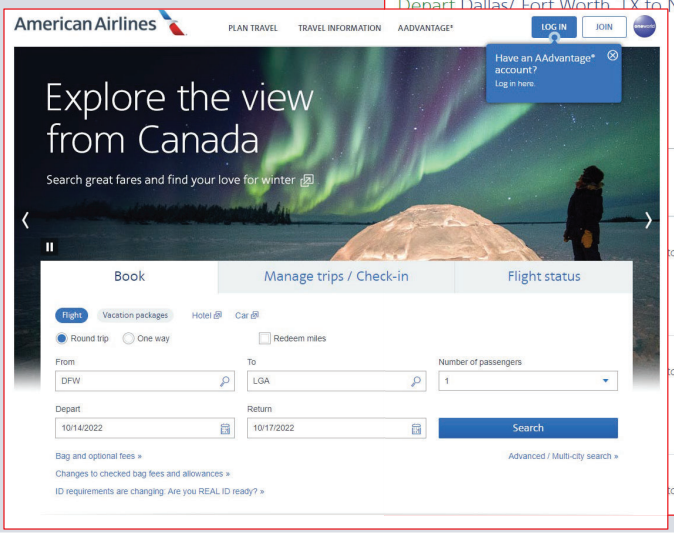
[3] DFW to LGA Search



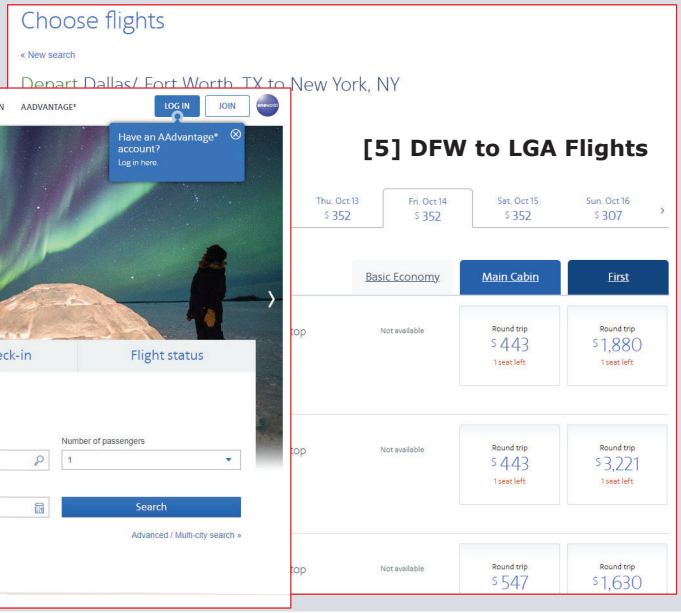
Claim Chart – Claim 10

Claim Feature	Evidence
10[c] communicating a web page including at least some of the search results from the server to the user.	For example, the Accused Instrumentalities display search result to a user on a webpage. The particular details are within AA's possession, custody, and control.

[4] AA Flights



[5] DFW to LGA Flights



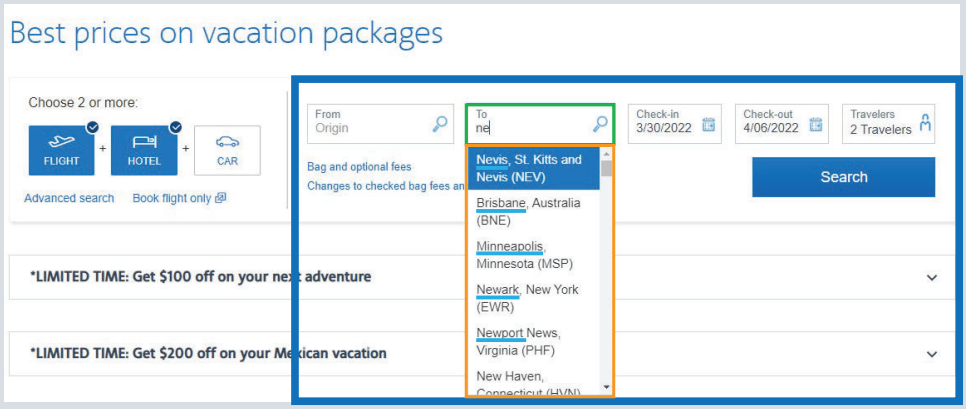
Claim 11

11[pre] The method of claim 10 further comprising:

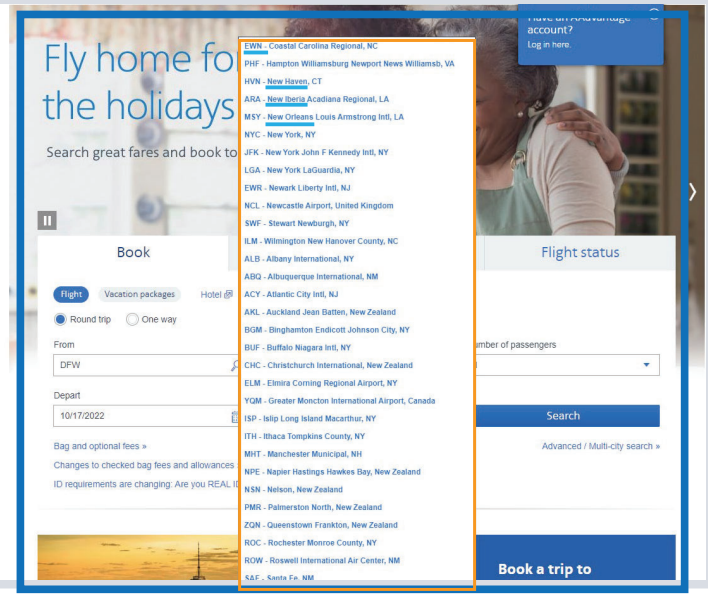
11[a] at the server, using the indexes to determine one or more alternative queries that match the submitted partial query,

11[b] and wherein communicating the web page comprises communicating a web page including at least some of the one or more alternative queries.

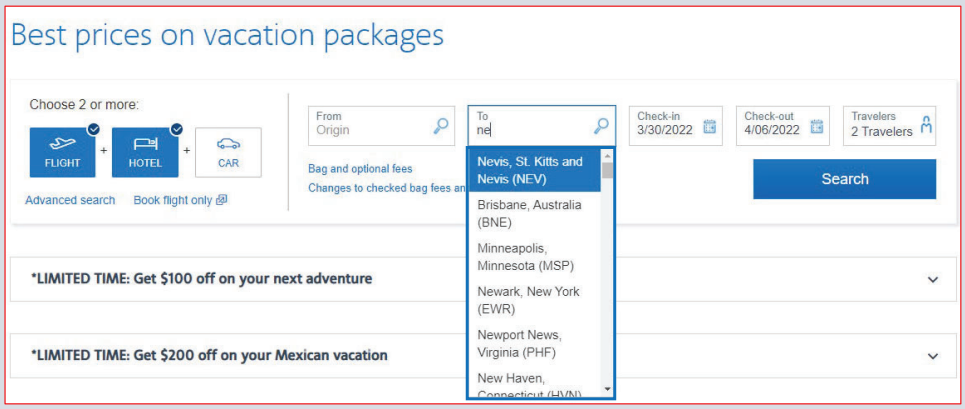
Claim Chart – Claim 11

Claim Feature	Evidence
11[pre] The method of claim 10 further comprising: **11[a]** at the server, using the indexes to determine one or more alternative queries that match the submitted partial query,	AA infringes Claim 11 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claims 8 and 10, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities, at the server, use the indexes to determine one or more alternative queries that match the submitted partial query. For example, the Accused Instrumentalities determine alternative queries that match the partial query using the indexes. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

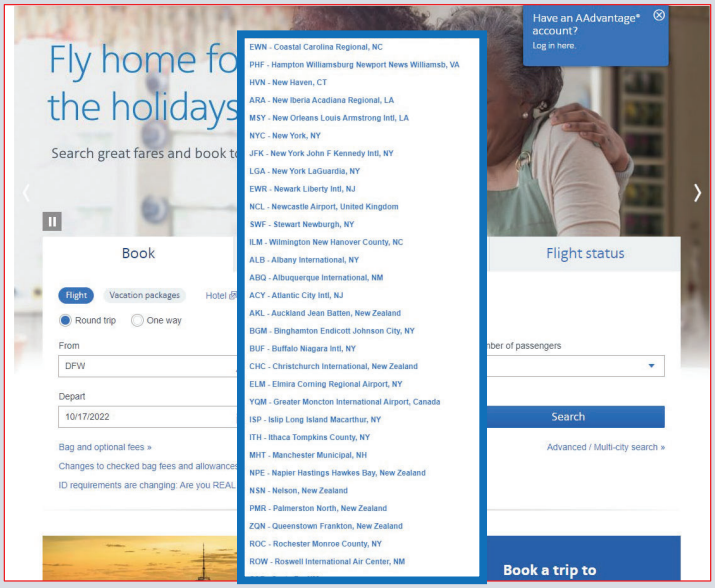
Claim Chart – Claim 11

Claim Feature	Evidence
11[pre] The method of claim 10 further comprising: **11[a]** at the server, using the indexes to determine one or more alternative queries that match the submitted partial query,	For example, the Accused Instrumentalities determine alternative queries that match the partial query using the indexes. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim Chart – Claim 11

Claim Feature	Evidence
11[b] and wherein communicating the web page comprises communicating a web page including at least some of the one or more alternative queries.	In the Accused Instrumentalities, communicating the web page comprises communicating a web page including at least some of the one or more alternative queries. For example, the Accused Instrumentalities include at least some of the alternative queries in a communicated webpage. **[1] AA Vacations Homepage** 

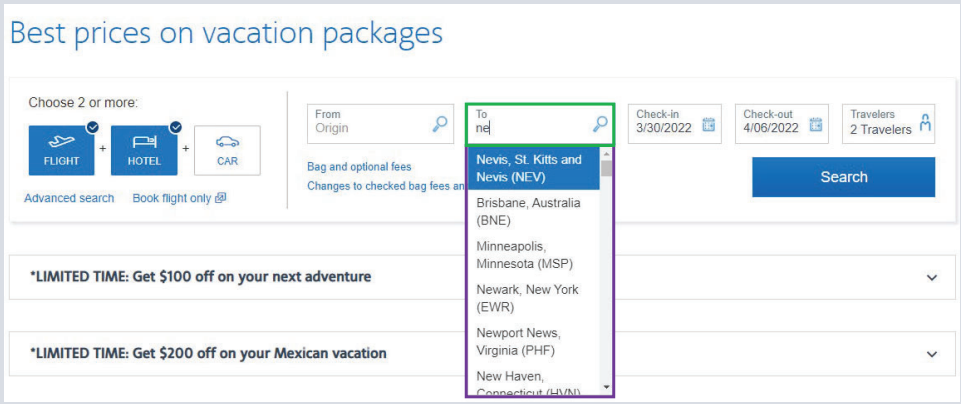
Claim Chart – Claim 11

Claim Feature	Evidence
11[b] and wherein communicating the web page comprises communicating a web page including at least some of the one or more alternative queries.	For example, the Accused Instrumentalities include at least some of the alternative queries in a communicated webpage. **[4] AA Flights** 

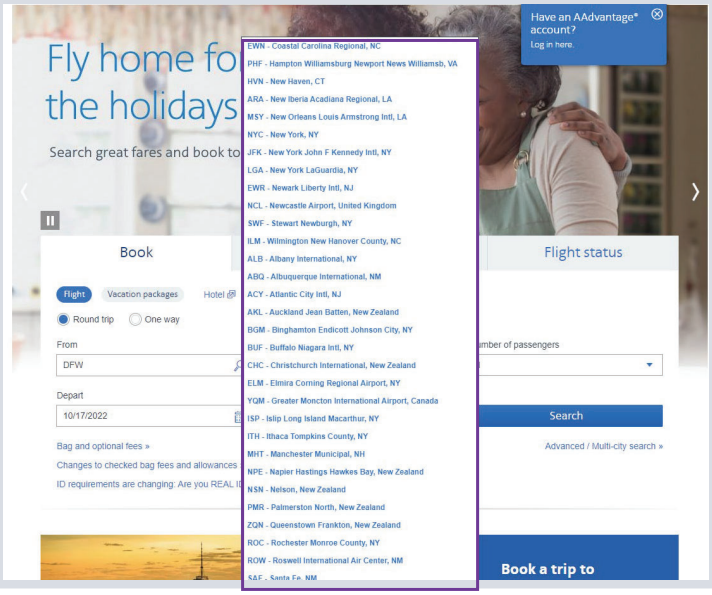
Claim 12

12 The method of claim 8 wherein receiving the partial query comprises interpreting the submitted partial query as a series of words separated by delimiters.

Claim Chart – Claim 12

Claim Feature	Evidence
12 The method of claim 8 wherein receiving the partial query comprises interpreting the submitted partial query as a series of words separated by delimiters.	AA infringes Claim 12 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claim 8, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities receive the partial query by interpreting the submitted partial query as a series of words separated by delimiters. For example, the Accused Instrumentalities can interpret a submitted partial query as including a series of words, e.g., "New" and "York" or "Haven." The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

Claim Chart – Claim 12

Claim Feature	Evidence
12 The method of claim 8 wherein receiving the partial query comprises interpreting the submitted partial query as a series of words separated by delimiters.	For example, the Accused Instrumentalities can interpret a submitted partial query as including a series of words, e.g. "New" and "York" or "Haven." The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

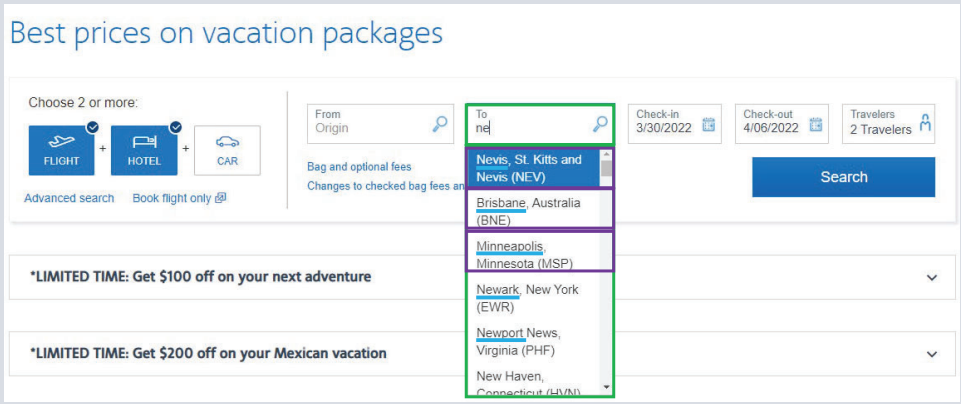
Claim 13

13[pre] The method of claim 12 further comprising:

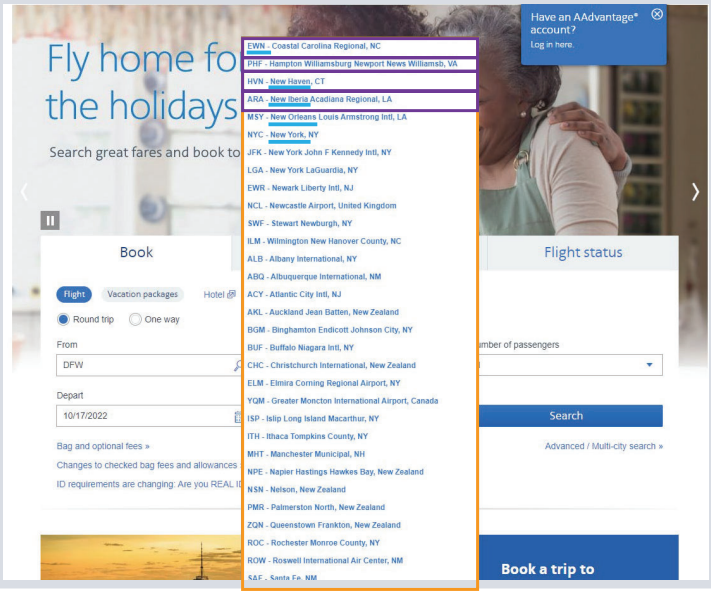
13[a] at the server, interpreting each word of the received partial query as a partial word of the entire search query desired by the user; and

13[b] using the indexes, determining a full query that matches the partial words specified in the partial query.

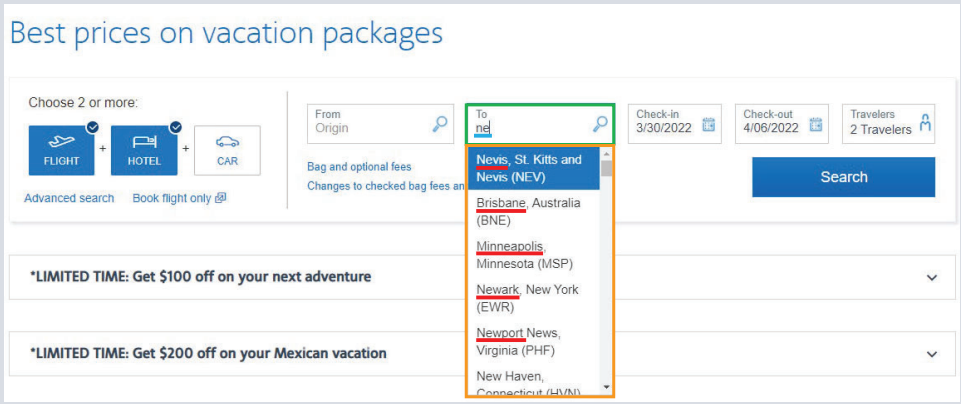
Claim Chart – Claim 13

Claim Feature	Evidence
13[pre] The method of claim 12 further comprising: **13[a]** at the server, interpreting each word of the received partial query as a partial word of the entire search query desired by the user; and	AA infringes Claim 13 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claims 8 and 12, the discussion of which is incorporated herein by reference. Furthermore, at the server, the Accused Instrumentalities interpret each word of the received partial query as a partial word of the entire search query desired by the user. For example, the Accused Instrumentalities interpret a submitted partial query as including, e.g., "Nevis" and "Newark" or "Brisbane". The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

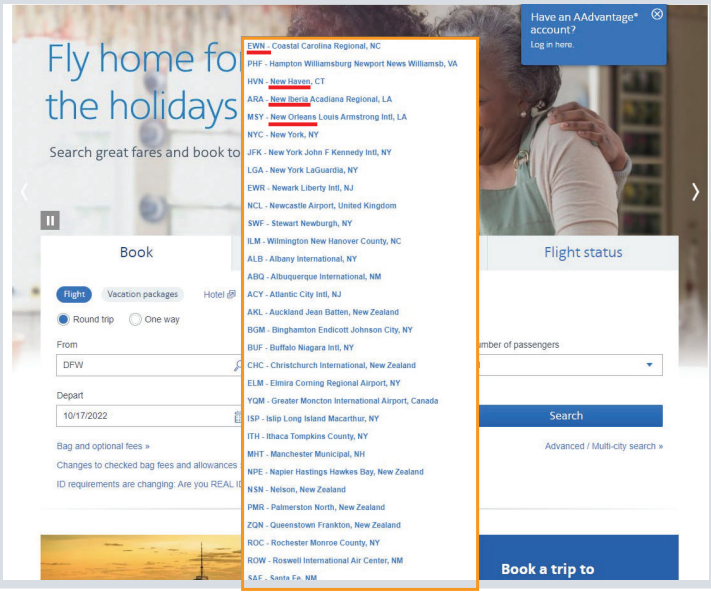
Claim Chart – Claim 13

Claim Feature	Evidence
13[pre] The method of claim 12 further comprising: **13[a]** at the server, interpreting each word of the received partial query as a partial word of the entire search query desired by the user; and	For example, the Accused Instrumentalities interpret a submitted partial query as including, e.g., "New" and "York." The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

Claim Chart – Claim 13

Claim Feature	Evidence
13[b] using the indexes, determining a full query that matches the partial words specified in the partial query.	The Accused Instrumentalities, using the indexes, determine a full query that matches the partial words specified in the partial query. For example, the Accused Instrumentalities determine a full query that matches the partial words (e.g. "ne", etc.) of the partial query. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

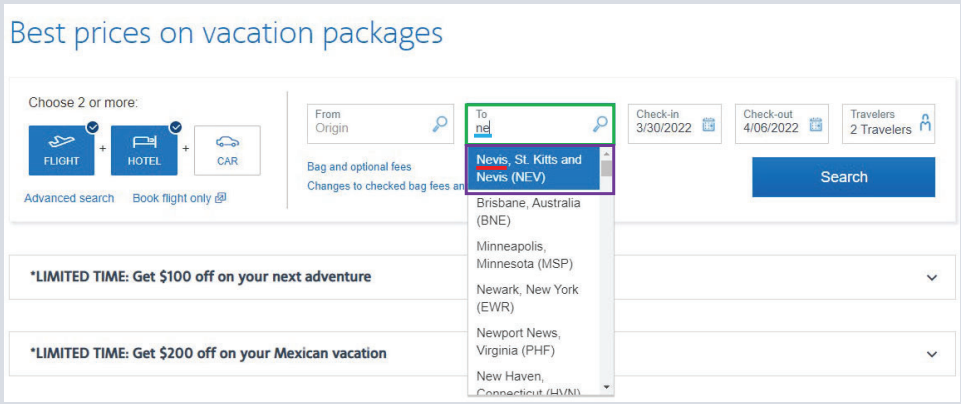
Claim Chart – Claim 13

Claim Feature	Evidence
13[b] using the indexes, determining a full query that matches the partial words specified in the partial query.	For example, the Accused Instrumentalities determine a full query that matches the partial words (e.g., "new", etc.) o the partial query. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

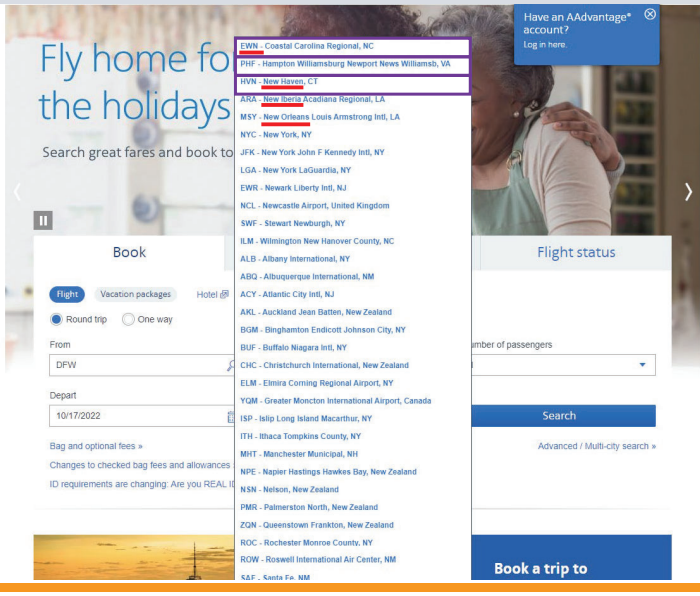
Claim 14

14 The method of claim 13 wherein determining a full query comprises determining the full query that is the most frequently submitted known query, as defined by the indexes, that matches the partial words of the received partial query in the same order.

Claim Chart – Claim 14

Claim Feature	Evidence
14 The method of claim 13 wherein determining a full query comprises determining the full query that is the most frequently submitted known query, as defined by the indexes, that matches the partial words of the received partial query in the same order.	AA infringes Claim 14 of the '317 patent. The Accused Instrumentalities satisfy each and every element of Claims 8 and 13, the discussion of which is incorporated herein by reference. Furthermore, the Accused Instrumentalities determine a full query by determining the full query that is the most frequently submitted known query, as defined by the indexes that matches the partial words of the received partial query in the same order. For example, searching "ne" generates results for "Nevis, St. Kitts and Nevis (NEV)", which (upon information and belief) is the full query that is most frequently submitted and matches the partial words of the received partial query in the same order. The particular details are within AA's possession, custody, and control. **[1] AA Vacations Homepage** 

Claim Chart – Claim 14

Claim Feature	Evidence
14 The method of claim 13 wherein determining a full query comprises determining the full query that is the most frequently submitted known query, as defined by the indexes, that matches the partial words of the received partial query in the same order.	For example, searching "new" generates results for "EWN – Coastal Carolina Regional, NC" and "PHF – Hampton Williamsburg Newport News Williamsb, VA" which (upon information and belief) are the full queries that are most frequently submitted and matches the partial words of the received partial query in the same order. The particular details are within AA's possession, custody, and control. **[4] AA Flights** 

END