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

NORTHEASTERN UNIVERSITY and JARG CORPORATION	§	
	§	
Plaintiffs,	§	
	§	Case No. 2:07-CV-486 (TJW)(CE)
v.	§	
	§	**JURY TRIAL DEMANDED**
GOOGLE INC.	§	
	§	
Defendant.	§	

Disclosure of Asserted Claims and Infringement Contentions

In accordance with Patent Rule 3-1, Plaintiffs Northeastern University and Jarg Corporation (collectively, "Northeastern") submit the following Disclosure of Asserted Claims and Infringement Contentions.

Northeastern makes these infringement contentions based on reasonably available information. Northeastern's investigation and analysis is ongoing, and Northeastern will obtain additional information through discovery. Additionally, the Court has not issued a Claim Construction Ruling. Consequently, Northeastern may amend or supplement this disclosure in accordance with Patent Rule 3-6.

I. Patent Rule 3-1(a)

Defendant Google Inc. ("Google") has infringed and continues to infringe at least claims 1, 2, 3, 8, 9 and 13 of U.S. Pat. No. 5,694,593 ("the '593 patent") entitled "Distributed Computer Database System and Method."

II. Patent Rule 3-1(b)

Google's internet search service, Google Web Search, has infringed and continues to infringe the asserted claims of the '593 patent.

III. Patent Rule 3-1(c)

Charts identifying where each element of each of the asserted claims is found within the Google Web Search service are attached as Exhibit A.

IV. Patent Rule 3-1(d)

Each element of the each of the asserted claims is present literally, or in the alternative, present under the doctrine of equivalents. See the claim chart submitted in accordance with PR 3-1(c) for further details.

V. Patent Rule 3-1(e)

The asserted claims are entitled to a priority date of no later than October 5, 1994.

VI. Patent Rule 3-1(f)

A prototype system that practiced at least the asserted claims was described in the '593 patent.

Dated: September 9, 2008

Respectfully submitted,

/s/ Stephen C. Stout

William B. Dawson (TX Bar No. 05603600)
VINSON & ELKINS L.L.P.
3700 Trammel Crow Center
3001 Ross Avenue
Dallas, Texas 75201-2975
Tel: (214) 220-7926
Fax: (214) 999-7926

David B. Weaver (TX Bar No. 00798576)
Christopher V. Ryan (TX Bar No. 24037412)
Michael Valek (TX Bar No.24044028)
R. Floyd Walker (TX Bar No. 24044751)
Stephen C. Stout (TX Bar No. 24060672)
VINSON & ELKINS L.L.P.
2801 Via Fortuna, Suite 100
Austin, Texas 78746
Tel: (512) 542-8400
Fax: (512) 236-3338
E-mail: dweaver@velaw.com
cryan@velaw.com
mvalek@velaw.com
fwalker@velaw.com
sstout@velaw.com

EXHIBIT A

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
1.	A method for information retrieval using fuzzy queries in a non-relational, distributed database system having	To the extent the preamble is a limitation on the scope of the claim, this element is present. *See, e.g.,*: "When all phases are complete, a GWS generates the appropriate HTML for the output page and returns it to the user's browser. In addition to the indexing and document serving phases, a GWS also initiates several other other ancillary tasks upon receiving a query, such as sending the query to a spell-checking system and to an ad-serving system to generate relevant advertisements (if any)." Luiz André Barroso, Jeffrey Dean, and Urs Hölzle, *Web Search for a Planet: The Google Cluster Architecture*, 23 IEEE MICRO 22, 24, col. 1 (2003) [hereinafter *Architecture*]. "Amenable to extensive parallelization, Google's Web Search Application lets different queries run on different processors and, by partitioning the overall index, also lets a single query use multiple processors." *Architecture*, at 22. "To provide sufficient capacity to handle query traffic, our service consists of multiple clusters distributed worldwide." *Architecture*, at 23, col. 1. "As with the index lookup phase, the strategy is to partition the processing of all documents by randomly distributing documents into smaller shards." *Architecture*, at 24, col. 1. Google's Web Search Application satisfies the preamble literally, and also satisfies the preamble under the doctrine of equivalents.
1.	a plurality of home nodes and	This element is present. *See, e.g.,*: "A hardware-based load balancer in each cluster monitors the available set of Google Web Servers (GWSs) and performs local load balancing across a set of them." *Architecture*, at 23, col. 1. "Each cluster has around a few thousand machines." *Architecture*, at 23, col. 1. "Fortunately, the search is highly parallelizable by dividing the index into pieces (index shards), each having a randomly chosen subset of documents from the full index. A pool of machines servers requests for each shard." *Architecture*, at 23, col. 2. The Google Web Servers and Index Servers literally meet the limitation of a plurality of home nodes. These severs also meet the limitation under the doctrine of equivalents.

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
1.	a plurality of query nodes	This element is present. *See, e.g.*,: "Document servers (docservers) handle this job, fetching each document from disk to extract the title and the keyword-in-context snippet." *Architecture*, at 24, col. 1. "Each cluster has around a few thousand machines." *Architecture*, at 23, col. 1. "[T]he strategy is to partition the processing of all documents by . . . having multiple server replicas responsible for handling each shard." *Architecture*, at 24, col. 1. Google's document servers literally meet the limitation of a plurality of query nodes. These servers also meet the limitation under the doctrine of equivalents.
1.	connected by a network, said method comprising the steps of:	This element is present. *See, e.g.*,: "The servers on each side of a rack interconnect via a 100-Mbps Ethernet switch that has one or two gigabit uplinks to a core gigabit switch that connects all racks together." *Architecture*, at 25, col. 1. The network that interconnects the home nodes and the query nodes literally meets this claim limitation. It also meets the limitation under the doctrine of equivalents.
1.	randomly selecting a first one of said plurality of home nodes;	This element is present. *See, e.g.*,: "The load-balancing system minimizes round-trip time for the user's request, while also considering the available capacity at the various clusters. The user's browser then sends a hypertext transport protocol (HTTP) request to one of these clusters, and thereafter, the processing of that query is entirely local to that cluster." *Architecture*, at 23, col. 1. Google's Web Search Application satisfies this limitation literally, and also satisfies it under the doctrine of equivalents.
1.	fragmenting, by said selected home node, a query from a user into a plurality of query fragments;	This element is present. *See, e.g.*,: "In the first phase, the index servers consult an inverted index that maps each query word to a matching list of documents (the hit list). The index servers then determine a set of relevant documents by intersecting the hits lists of the individual query words." *Architecture*, at 23, col. 2. Google's home node literally fragments the query into a plurality of query fragments. Google also meets this limitation under the

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
		doctrine of equivalents.
1.	hashing, by said selected home node, each said query fragment of said plurality of query fragments, said hashed query fragment having a first portion and a second portion;	This element is present. *See, e.g.*,: "The final result of this first phase of query execution is an ordered list of document identifiers (*docids*)." *Architecture*, at 23, col. 2. "In the first phase, the index servers consult an inverted index that maps each query word to a matching list of documents (the hit list). The index servers then determine a set of relevant documents by intersecting the hits lists of the individual query words." *Architecture*, at 23, col. 2. "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." The Google Linux Cluster (Presentation by Urs Hölzle, University of Washington Department of Computer Science and Engineering Colloquia, November 5, 2002) [hereinafter *Hölzle Presentation*]. Google's process of transforming the query fragments into hashed query fragments (docids) literally meets this element of the claim. Google's process also performs this claimed step under the doctrine of equivalents.
1.	transmitting, by said selected home node, each said hashed query fragment of said plurality of query fragments to a respective one of said plurality of query nodes indicated by said first portion of each said hashed query fragment;	This element is present. *See, e.g.*,: "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." *Hölzle Presentation.* "The final result of this first phase of query execution is an ordered list of document identifiers (*docids*). As Figure 1 shows, the second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary." *Architecture*, at 23-24. Google's home nodes literally meets this limitation. Google's home node also meets this limitation under the doctrine of equivalents.
1.	using, by said query node, said second portion of said respective hashed query fragment to access	This element is present. *See, e.g.*,: "[T]he second phase involves taking this list of docids and

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
	data according to a local hash table located on said query node; and	computing the actual title and uniform resource locator of these documents, along with a query-specific document summary. "Document servers (docservers) handle this job, fetching each document from disk to extract the title and keyword-in-context snippet." *Architecture*, at 23-24. "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." *Hölzle Presentation.* Google's query nodes literally meet this limitation. Google's query nodes also meet this limitation under the doctrine of equivalents.
1.	returning, by each said query node accessing data according to said respective hashed query fragment, an object identifier corresponding to said accessed data to said selected home node.	This element is present. *See, e.g.*,: "[T]he second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary. "Document servers (docservers) handle this job, fetching each document from disk to extract the title and keyword-in-context snippet." *Architecture*, at 23-24. Google's query nodes literally meet this limitation. Google's query nodes also meet this limitation under the doctrine of equivalents.
2.	The method of claim 1 further comprising the step of receiving, at said home node, said query from said user, prior to the step of fragmenting said query.	This element is present. *See, e.g.*,: "After receiving a query, a GWS machine coordinates the query execution and formats the results into a Hypertext Markup Language (HTML) response to the user's browser." *Architecture*, at 23, col. 1. Google's home node literally performs the step of receiving the query from the user prior to the step of fragmenting the query. Google's home node also meets this limitation under the doctrine of equivalents.
3.	The method of claim 1 further comprising the steps of: determining, by said home nodc, a measure of relevance between said accessed data and said query; and	This element is present. *See, e.g.*,: "The index servers then determine a set of relevant documents by intersecting the hit lists of the individual query words, and they compute a relevance score for each document." *Architecture*, at 23, col. 2. Google's home nodes literally perform this claim limitation, and

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
		also do so under the doctrine of equivalents.
	returning, to said user, by said home node, accessed data having a predetermined degree of relevance, subsequent to the step of returning said object identifier.	This element is present. *See, e.g.*,: "This relevance score determines the order of results on the output page." *Architecture*, at 23, col. 2. Google's home nodes literally perform this claim limitation, and also do so under the doctrine of equivalents.
8.	A non-relational, distributed database system having an information retrieval tool for handling queries from a user, comprising:	To the extent the preamble is a limitation on the scope of the claim, this element is present. *See, e.g.*,: "Amenable to extensive parallelization, Google's Web Search Application lets different queries run on different processors and, by partitioning the overall index, also lets a single query use multiple processors." *Architecture*, at 22. "To provide sufficient capacity to handle query traffic, our service consists of multiple clusters distributed worldwide." *Architecture*, at 23, col. 1. "As with the index lookup phase, the strategy is to partition the processing of all documents by randomly distributing documents into smaller shards." *Architecture*, at 24, col. 1. Google's Web Search Application satisfies the preamble literally, and also satisfies the preamble under the doctrine of equivalents.
8.	a plurality of home nodes; and	This element is present. *See, e.g.*,: "Each cluster has around a few thousand machines." *Architecture*, at 23, col. 1. "A hardware-based load balancer in each cluster monitors the available set of Google Web Servers (GWSs) and performs local load balancing across a set of them." *Architecture*, at 23, col. 1. "Fortunately, the search is highly parallelizable by dividing the index into pieces (index shards), each having a randomly chosen subset of documents from the full index. A pool of machines servers requests for each shard." *Architecture*, at 23, col. 2. The Google Web Servers and Index Servers literally meet the limitation of a plurality of home nodes. These severs also meet the limitation under the doctrine of equivalents.
8.	a plurality of query nodes;	This element is present. *See, e.g.*,: "Document servers (docservers) handle this job, fetching each document from disk to extract the title and the keyword-in-context

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
		snippet." *Architecture*, at 24, col. 1. "Each cluster has around a few thousand machines." *Architecture*, at 23, col. 1. "The docserver cluster must have access to an online, low-latency copy of the entire Web." *Architecture*, at 24, col. 1. "[T]he strategy is to partition the processing of all documents by . . . having multiple server replicas responsible for handling each shard." *Architecture*, at 24, col. 1. Google's document servers literally meet the limitation of a plurality of query nodes. These servers also meet the limitation under the doctrine of equivalents.
8.	said plurality of home nodes and said plurality of query nodes connected by a network,	This element is present. *See, e.g.*,: "The servers on each side of a rack interconnect via a 100-Mbps Ethernet switch that has one or two gigabit uplinks to a core gigabit switch that connects all racks together." *Architecture*, at 25, col. 1. *See also* e.g. Fig. 1., *Architecture* at 24, col. 2. The network that interconnects the home nodes and the query nodes literally meets this claim limitation. It also meets the limitation under the doctrine of equivalents.
8.	wherein each said home node, upon receiving a query from a user, fragments said query into a plurality of query fragments,	This element is present. *See, e.g.*,: "[A] GWS machine coordinates the query execution and formats the results into a Hypertext Markup Language (HTML) response to the user's browser. Figure 1 illustrates these steps." *Architecture*, at 23, col. 1. "In the first phase, the index servers consult an inverted index that maps each query word to a matching list of documents (the hit list). The index servers then determine a set of relevant documents by intersecting the hits lists of the individual query words." *Architecture*, at 23, col. 2. Google's home node literally fragments the query into a plurality of query fragments. Google also meets this limitation under the doctrine of equivalents.
8.	hashes each said query fragment of said plurality of query fragments into a hashed query fragment having a first portion and a second portion,	This element is literally present. *See, e.g.*,: "The final result of this first phase of query execution is an ordered list of document identifiers (*docids)*." *Architecture*, at 23, col. 2. "In the first phase, the index servers consult an inverted index that maps each query word to a matching list of documents (the hit

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
		list). The index servers then determine a set of relevant documents by intersecting the hits lists of the individual query words." *Architecture*, at 23, col. 2. "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." *Hölzle Presentation.* Google's process of transforming the query fragments into hashed query fragments (docids) literally meets this element of the claim. Google's process also performs this claimed step under the doctrine of equivalents.
8.	and transmits each said hashed query fragment to a respective one of said plurality of query nodes indicated by said first portion of said hashed query fragment, and	This element is present. *See, e.g.*,: "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." *Hölzle Presentation.* "The final result of this first phase of query execution is an ordered list of document identifiers (*docids*). As Figure 1 shows, the second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary." *Architecture*, at 23-24. Google's home nodes literally meets this limitation. Google's home node also meets this limitation under the doctrine of equivalents.
8.	further wherein each said query node uses said second portion of said hashed query fragment to access data according to a local hash table located on said query node and	This element is present. *See, e.g.*,: "[T]he second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary. "Document servers (docservers) handle this job, fetching each document from disk to extract the title and keyword-in-context snippet." *Architecture*, at 23-24. "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." *Hölzle Presentation.*

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
		Google's query nodes literally meet this limitation. Google's query nodes also meet this limitation under the doctrine of equivalents.
8.	returns an object identifier corresponding to said accessed data to said home node.	This element is present. *See, e.g.*,: "[T]he second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary. "Document servers (docservers) handle this job, fetching each document from disk to extract the title and keyword-in-context snippet." *Architecture*, at 23-24. Google's query nodes literally meet this limitation. Google's query nodes also meet this limitation under the doctrine of equivalents.
9.	The distributed database system of claim 8 wherein said home node determines a measure of relevance between said accessed data and said query and returns to said user accessed data having a predetermined degree of relevance.	This element is present. *See, e.g.*,: "The index servers then determine a set of relevant documents by intersecting the hit lists of the individual query words, and they compute a relevance score for each document. This relevance score determines the order of results on the output page." *Architecture*, at 23, col. 2. Google's Web Search literally meets this limitation. Google's Web Search also meets this limitation under the doctrine of equivalents.
13.	A non-relational, distributed database system having an information retrieval tool for handling queries from a user, comprising:	To the extent the preamble is a limitation on the scope of the claim, this element is present. *See, e.g.*,: "Amenable to extensive parallelization, Google's Web Search Application lets different queries run on different processors and, by partitioning the overall index, also lets a single query use multiple processors." *Architecture*, at 22. "To provide sufficient capacity to handle query traffic, our service consists of multiple clusters distributed worldwide." *Architecture*, at 23, col. 1. "As with the index lookup phase, the strategy is to partition the processing of all documents by randomly distributing documents into smaller shards." *Architecture*, at 24, col. 1. Google's Web Search Application satisfies the preamble literally, and also satisfies the preamble under the doctrine of equivalents.
13.	a plurality of home nodes; and	This element is present. *See, e.g.*,:

Northeastern University and Jarg Corp. v. Google Inc., **Case No. 2:07-CV-486 (TJW)(CE)**		
Claim #	**Element**	**First Infringement Contention**
		"Each cluster has around a few thousand machines." *Architecture*, at 23, col. 1. "A hardware-based load balancer in each cluster monitors the available set of Google Web Servers (GWSs) and performs local load balancing across a set of them." *Architecture*, at 23, col. 1. "Fortunately, the search is highly parallelizable by dividing the index into pieces (index shards), each having a randomly chosen subset of documents from the full index. A pool of machines servers requests for each shard." *Architecture*, at 23, col. 2. The Google Web Servers and Index Servers literally meet the limitation of a plurality of home nodes. These severs also meet the limitation under the doctrine of equivalents.
13.	a plurality of query nodes,	This element is present. *See, e.g.*,: "Document servers (docservers) handle this job, fetching each document from disk to extract the title and the keyword-in-context snippet." *Architecture*, at 24, col. 1. "Each cluster has around a few thousand machines." *Architecture*, at 23, col. 1. "[T]he strategy is to partition the processing of all documents by . . . having multiple server replicas responsible for handling each shard." *Architecture*, at 24, col. 1. Google's document servers literally meet the limitation of a plurality of query nodes. These servers also meet the limitation under the doctrine of equivalents.
13.	said plurality of home nodes and said plurality of query nodes connected by a network,	This element is present. *See, e.g.*,: "The servers on each side of a rack interconnect via a 100-Mbps Ethernet switch that has one or two gigabit uplinks to a core gigabit switch that connects all racks together." *Architecture*, at 25, col. 1. *See also* e.g. Fig. 1. *Architecture* at 24, col. 2. The network that interconnects the home nodes and the query nodes literally meets this claim limitation. It also meets the limitation under the doctrine of equivalents.
13.	each said home node, upon receiving a command from a user, enqueueing a predetermined task in response to said command,	This element is present. *See, e.g.*,: "After receiving a query, a GWS machine coordinates the query execution and formats the results into a Hypertext Markup Language (HTML) response to the user's browser. *Architecture*, at 23, col. 1.

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
		Google's home node literally meets this claim limitation. Google's home node also satisfies this limitation under the doctrine of equivalents.
13.	a query task enqueued being resultant in, in response to a query command from said user, fragmenting a query contained in said query command into a plurality of query fragments,	This element is present. See, e.g.,: "The user's browser then sends a hypertext transport protocol (HTTP) request to one of these clusters, and thereafter, the processing of that query is entirely local to that cluster. A hardware-based load balancer in each cluster monitors the available set of Google Web servers (GWSs) and performs local load balancing of requests across a set of them. After receiving a query, a GWS machine coordinates the query execution and formats the results into a Hypertext Markup Language (HTML) response to the user's browser. Figure 1 illustrates these steps." *Architecture*, at 23, col. 1. "In the first phase, the index servers consult an inverted index that maps each query word to a matching list of documents (the hit list). The index servers then determine a set of relevant documents by intersecting the hits lists of the individual query words." *Architecture*, at 23, col. 2. Google's Web Search Application satisfies this element literally, and also satisfies it under the doctrine of equivalents.
13.	hashing each said query fragment of said plurality of query fragments into a hashed query fragment having a first portion and a second portion, and	This element is present. *See, e.g.,*: "The final result of this first phase of query execution is an ordered list of document identifiers (*docids).*" *Architecture*, at 23, col. 2. "In the first phase, the index servers consult an inverted index that maps each query word to a matching list of documents (the hit list). The index servers then determine a set of relevant documents by intersecting the hits lists of the individual query words." *Architecture*, at 23, col. 2. "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." *Hölzle Presentation*. Google's process of transforming the query fragments into hashed query fragments (docids) literally meets this element of the claim. Google's process also performs this claimed step under the doctrine of equivalents.

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**First Infringement Contention**
13.	transmitting a query message containing each said hashed query fragment to a respective one of said plurality of query nodes indicated by said first portion of said hashed query fragment,	This element is present. *See, e.g.*,: "The final result of this first phase of query execution is an ordered list of document identifiers (*docids*). As Figure 1 shows, the second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary." *Architecture*, at 23-24. "So, for example, on the document server side, where we have the cached web, we basically hash the URL into, you know, a 128 bit number, or something like that, and then that number modulo whatever, you know, your server, number of server shards you have, gives you the server, the clone set number, that contains this document." *Hölzle Presentation.* Google's home node literally meets this limitation. Google's home node also meets this limitation under the doctrine of equivalents.
13.	said query node, upon receipt of said query message, using said second portion of said hashed query fragment to access data according to a local hash table located on said query node and	This element is present. *See, e.g.*,: "[T]he second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary. "Document servers (docservers) handle this job, fetching each document from disk to extract the title and keyword-in-context snippet." *Architecture*, at 23-24. Google's query nodes literally perform this claim limitation. Google's query nodes also perform this limitation under the doctrine of equivalents.
13.	transmitting a message returning an object identifier corresponding to said accessed data to said home node.	This element is present. *See, e.g.*,: "[T]he second phase involves taking this list of docids and computing the actual title and uniform resource locator of these documents, along with a query-specific document summary. "Document servers (docservers) handle this job, fetching each document from disk to extract the title and keyword-in-context snippet." *Architecture*, at 23-24. Google's query node literally meets this limitation. Google's query nodes also meet this limitation under the doctrine of equivalents.

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**Second Infringement Contention**
8.	A non-relational, distributed database system having an information retrieval tool for handling queries from a user, comprising:	To the extent the preamble is a limitation on the scope of the claim, this element is present. *See, e.g.,*: "Google is designed to be a scalable search engine. The primary goal is to provide high quality search results over a rapidly growing World Wide Web. Google employs a number of techniques to improve search quality including page rank, anchor text, and proximity information. Furthermore, Google is a complete architecture for gathering web pages, indexing them, and performing search queries over them." Sergey Brin and Lawrence Page, *The Anatomy of a Large-Scale Hypertextual Web Search Engine,* 30 COMPUTER NETWORKS 107, Section 6 (1998) [Hereinafter, *Anatomy*]. "BigFiles are virtual files spanning multiple file systems and are addressable by 64 bit integers. The allocation among multiple file systems is handled automatically." *Anatomy*, Section 4.2.1. "The current version of Google answers most queries in between 1 and 10 seconds. This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. Google's Web Search Application satisfies the preamble literally, and also satisfies the preamble under the doctrine of equivalents.
8.	a plurality of home nodes; and	This element is present. *See, e.g.,*: "The searcher is run by a web server and uses the lexicon built by DumpLexicon together with the inverted index and the PageRanks to answer queries." *Anatomy*, Section 4.1. "This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. "We intend to speed up Google considerably through distribution and hardware, software, and algorithmic improvements." *Anatomy*, Section 5.3. "In the buying spree, Google expanded from having about 300 computers when Reese joined [in 1999] to 2,000 one month later, and twice that number by the following summer. . . . At the time Google had two data centers in northern California and a third in the Washington DC area, later adding many more across the U.S. and overseas." DAVID A. VISE AND MARK MALSEED, THE GOOGLE STORY 80 (2005). The Google web servers literally meet the limitation of a plurality

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**Second Infringement Contention**
		of home nodes. These severs also meet the limitation under the doctrine of equivalents.
8.	a plurality of query nodes;	This element is present. *See, e.g.*,: "The inverted index consists of the same barrels as the forward index, except that they have been processed by the sorter. For every valid wordID, the lexicon contains a pointer into the barrel that wordID falls into." *Anatomy*, Section 4.2.7. "The current version of Google answers most queries in between 1 and 10 seconds. This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. "We intend to speed up Google considerably through distribution and hardware, software, and algorithmic improvements." *Anatomy*, Section 5.3. Google's barrels literally meet the limitation of a plurality of query nodes. The barrels also meet the limitation under the doctrine of equivalents.
8.	said plurality of home nodes and said plurality of query nodes connected by a network,	This element is present. *See, e.g.*,: "BigFiles are virtual files spanning multiple file systems and are addressable by 64 bit integers. The allocation among multiple file systems is handled automatically." *Anatomy*, Section 4.2.1. "This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. "We intend to speed up Google considerably through distribution and hardware, software, and algorithmic improvements." *Anatomy*, Section 5.3. The network that interconnects the home nodes and the query nodes literally meets this claim limitation. It also meets the limitation under the doctrine of equivalents.
8.	wherein each said home node, upon receiving a query from a user, fragments said query into a plurality of query fragments,	This element is present. *See, e.g.*,: "1. Parse the query." *Anatomy*, Fig. 4. "Every word is converted into a wordID by using an in-memory hash table – the lexicon." *Anatomy*, Section 4.4. Google's home node literally fragments the query into a plurality of query fragments. Google also meets this limitation under the doctrine of equivalents.

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**Second Infringement Contention**
8.	hashes each said query fragment of said plurality of query fragments into a hashed query fragment having a first portion and a second portion,	This element is present. *See, e.g.*,: "2. Convert words into wordIDs." *Anatomy*, Fig. 4. "Every word is converted into a wordID by using an in-memory hash table – the lexicon." *Anatomy*, Section 4.4. "For every valid wordID, the lexicon contains a pointer into the barrel that wordID falls into." *Anatomy*, Section 4.2.7 Google's process of converting words to wordIDs literally meets this element of the claim. Google's process also performs this claimed step under the doctrine of equivalents.
8.	and transmits each said hashed query fragment to a respective one of said plurality of query nodes indicated by said first portion of said hashed query fragment, and	This element is present. *See, e.g.*,: "The forward index is actually already partially sorted. It is stored in a number of barrels (we used 64). Each barrel holds a range of wordID's." *Anatomy*, Section 4.2.6. "The sorter takes the barrels, which are sorted by docID (this is a simplification, see Section 4.2.5), and resorts them by wordID to generate the inverted index." *Anatomy*, Section 4.1. "The inverted index consists of the same barrels as the forward index, except that they have been processed by the sorter. For every valid wordID, the lexicon contains a pointer into the barrel that wordID falls into." *Anatomy*, Section 4.2.7. "3. Seek to the start of the doclist in the short barrel for every word." *Anatomy*, Fig. 4. Google's home nodes literally meets this limitation. Google's home node also meets this limitation under the doctrine of equivalents.
8.	further wherein each said query node uses said second portion of said hashed query fragment to access data according to a local hash table located on said query node and	This element is present. *See, e.g.*,: "The inverted index consists of the same barrels as the forward index, except that they have been processed by the sorter. For every valid wordID, the lexicon contains a pointer into the barrel that wordID falls into. It points to a doclist of docID's together with their corresponding hit lists. This doclist represents all the occurrences of that word in all documents." *Anatomy*, Section 4.2.7. "3. Seek to the start of the doclist in the short barrel for every word." *Anatomy*, Fig. 4. Google's query nodes literally meet this limitation. Google's

Northeastern University and Jarg Corp. v. Google Inc., **Case No. 2:07-CV-486 (TJW)(CE)**		
Claim #	**Element**	**Second Infringement Contention**
		query nodes also meet this limitation under the doctrine of equivalents.
8.	returns an object identifier corresponding to said accessed data to said home node.	This element is present. *See, e.g.,*: "4. Scan through the doclists until there is a document that matches all the search terms." *Anatomy*, Fig. 4. "Sort the documents that have matched by rank and return the top k." *Anatomy*, Fig. 4. Google's query nodes literally meets this limitation. Google's query nodes also meet this limitation under the doctrine of equivalents.
9.	The distributed database system of claim 8 wherein said home node determines a measure of relevance between said accessed data and said query and returns to said user accessed data having a predetermined degree of relevance.	This element is present. *See, e.g.,*: "5. Compute the rank of that document for the query." *Anatomy*, Fig. 4. "Sort the documents that have matched by rank and return the top k." *Anatomy*, Fig. 4. Google's Web Search literally meets this limitation. Google's Web Search also meets this limitation under the doctrine of equivalents.
13.	A non-relational, distributed database system having an information retrieval tool for handling queries from a user, comprising:	To the extent the preamble is a limitation on the scope of the claim, this element is present. *See, e.g.,*: "Google is designed to be a scalable search engine. The primary goal is to provide high quality search results over a rapidly growing World Wide Web. Google employs a number of techniques to improve search quality including page rank, anchor text, and proximity information. Furthermore, Google is a complete architecture for gathering web pages, indexing them, and performing search queries over them." *Anatomy*, Section 6. "BigFiles are virtual files spanning multiple file systems and are addressable by 64 bit integers. The allocation among multiple file systems is handled automatically." *Anatomy*, Section 4.2.1. "The current version of Google answers most queries in between 1 and 10 seconds. This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. Google's Web Search Application satisfies the preamble literally, and also satisfies the preamble under the doctrine of equivalents.

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**Second Infringement Contention**
13.	a plurality of home nodes; and	This element is present. *See, e.g.*,: "The searcher is run by a web server and uses the lexicon built by DumpLexicon together with the inverted index and the PageRanks to answer queries." *Anatomy*, Section 4.1. "This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. "We intend to speed up Google considerably through distribution and hardware, software, and algorithmic improvements." *Anatomy*, Section 5.3. The Google web servers literally meet the limitation of a plurality of home nodes. These severs also meet the limitation under the doctrine of equivalents.
13.	a plurality of query nodes,	This element is present. *See, e.g.*,: "The inverted index consists of the same barrels as the forward index, except that they have been processed by the sorter. For every valid wordID, the lexicon contains a pointer into the barrel that wordID falls into." *Anatomy*, Section 4.2.7. "This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. "We intend to speed up Google considerably through distribution and hardware, software, and algorithmic improvements." *Anatomy*, Section 5.3. Google's barrels literally meet the limitation of a plurality of query nodes. The barrels also meet the limitation under the doctrine of equivalents.
13.	said plurality of home nodes and said plurality of query nodes connected by a network,	This element is present. *See, e.g.*,: "BigFiles are virtual files spanning multiple file systems and are addressable by 64 bit integers. The allocation among multiple file systems is handled automatically." *Anatomy*, Section 4.2.1. "This time is mostly dominated by disk IO over NFS (since disks are spread over a number of machines)." *Anatomy*, Section 5.3. "We intend to speed up Google considerably through distribution and hardware, software, and algorithmic improvements." *Anatomy*, Section 5.3. The network that interconnects the home nodes and the query nodes literally meets this claim limitation. It also meets the

Northeastern University and Jarg Corp. v. Google Inc., **Case No. 2:07-CV-486 (TJW)(CE)**		
Claim #	**Element**	**Second Infringement Contention**
		limitation under the doctrine of equivalents.
13.	each said home node, upon receiving a command from a user, enqueueing a predetermined task in response to said command,	This element is present. *See, e.g.*,: "The searcher is run by a web server and uses the lexicon built by DumpLexicon together with the inverted index and the PageRanks to answer queries." *Anatomy*, Section 4.1. "The current version of Google answers most queries in between 1 and 10 seconds." *Anatomy*, Section 5.3. Google's Web Search Application satisfies this element literally, and also satisfies it under the doctrine of equivalents.
13.	a query task enqueued being resultant in, in response to a query command from said user, fragmenting a query contained in said query command into a plurality of query fragments,	This element is present. *See, e.g.*,: "1. Parse the query." *Anatomy*, Fig. 4. "Every word is converted into a wordID by using an in-memory hash table – the lexicon." *Anatomy*, Section 4.4. Google's Web Search Application satisfies this element literally, and also satisfies it under the doctrine of equivalents.
13.	hashing each said query fragment of said plurality of query fragments into a hashed query fragment having a first portion and a second portion, and	This element is present. *See, e.g.*,: "2. Convert words into wordIDs." *Anatomy*, Fig. 4. "Every word is converted into a wordID by using an in-memory hash table – the lexicon." *Anatomy*, Section 4.4. "For every valid wordID, the lexicon contains a pointer into the barrel that wordID falls into." *Anatomy*, Section 4.2.7 Google's process of converting words to wordIDs literally meets this element of the claim. Google's process also performs this claimed step under the doctrine of equivalents.
13.	transmitting a query message containing each said hashed query fragment to a respective one of said plurality of query nodes indicated by said first portion of said hashed query fragment,	This element is present. *See, e.g.*,: "The forward index is actually already partially sorted. It is stored in a number of barrels (we used 64). Each barrel holds a range of wordID's." *Anatomy*, Section 4.2.6. "The sorter takes the barrels, which are sorted by docID (this is a simplification, see Section 4.2.5), and resorts them by wordID to generate the inverted index." *Anatomy*, Section 4.1. "The inverted index consists of the same barrels as the forward index, except that they have been processed by the sorter. For every valid wordID, the lexicon contains a pointer into the barrel

Northeastern University and Jarg Corp. v. Google Inc., Case No. 2:07-CV-486 (TJW)(CE)		
Claim #	**Element**	**Second Infringement Contention**
		that wordID falls into." *Anatomy*, Section 4.2.7. "3. Seek to the start of the doclist in the short barrel for every word." *Anatomy*, Fig. 4. Google's home node literally meets this limitation. Google's home node also meets this limitation under the doctrine of equivalents.
13.	said query node, upon receipt of said query message, using said second portion of said hashed query fragment to access data according to a local hash table located on said query node and	This element is present. *See, e.g.*,: "The inverted index consists of the same barrels as the forward index, except that they have been processed by the sorter. For every valid wordID, the lexicon contains a pointer into the barrel that wordID falls into. It points to a doclist of docID's together with their corresponding hit lists. This doclist represents all the occurrences of that word in all documents." *Anatomy*, Section 4.2.7. "3. Seek to the start of the doclist in the short barrel for every word." *Anatomy*, Fig. 4. Google's query node literally meets this limitation. Google's query node also meets this limitation under the doctrine of equivalents.
13.	transmitting a message returning an object identifier corresponding to said accessed data to said home node.	This element is present. *See, e.g.*,: "4. Scan through the doclists until there is a document that matches all the search terms." *Anatomy*, Fig. 4. "Sort the documents that have matched by rank and return the top k." *Anatomy*, Fig. 4. Google's query node literally meets this limitation. Google's query node also meets this limitation under the doctrine of equivalents.