Filed on Behalf of NETWORK-1 TECHNOLOGIES, INC.

By: Charles R. Macedo (Reg. No. 32,781)
Brian A. Comack (Reg. No. 45,343)
Amster, Rothstein & Ebenstein LLP
90 Park Avenue
New York, NY 10016
Facsimile: (212) 336–8001
cmacedo@arelaw.com
N1-Google-IPR@arelaw.com

Gregory Dovel (admitted *pro hac vice*)
Dovel & Luner, LLP
201 Santa Monica Blvd., Suite 600
Santa Monica, CA 90401
Telephone: (310) 656-7066
greg@dovellaw.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE INC.
Petitioner

v.

NETWORK-1 TECHNOLOGIES, INC.
Patent Owner

Cases IPR2015-00348
Patent 8,656,441

PATENT OWNER'S RESPONSE PURSUANT TO 37 C.F.R. § 42.120

Table of Contents

TABLE OF AUTHORITIES

Page(s)

CASES

OTHER AUTHORITIES

Exhibit List

Exhibit Number	Description
2001	"Brute-force search"—http://en.wikipedia.org/wiki/Bruteforce_ search (3/19/2015)
2002	U.S. Patent 8,447,762 (Brendel)
2003	U.S. Patent 7,167,984 (Graveman)
2004	Declaration of Greg Dovel in Support for *Pro Hac Vice* Admission
2005	Declaration of Dr. George Karypis
2006	Deposition of Dr. Pierre Moulin, dated August 19-20, 2015
2007	Modern Dictionary of Electronics 425-426 (1999).
2008	http://en.wikipedia.org/wiki/Nearest_neighbor_search#Approximate_nearest_neighbor
2009	http://en.wikipedia.org/wiki/Big_O_notation
2010	P. N. Yianilos, "Locally lifting the curse of Dimensionality for nearest Neighbor Search" SODA 2000: 361-370

Patent Owner Network-1 respectfully submits this response to the Petition as instituted by the decision to institute an *Inter Partes* Review of Patent No. 8,656,441 (the "'441 patent"). IPR2015-00348, paper 6 ("Decision").

Network-1 appreciates the guidance and concerns that the Board provided and raised in the Decision. In this Response, Network-1 addresses the Board's specific concerns. In doing so, Network-1 provides an extensive and detailed technical analysis in the accompanying Declaration of expert Dr. George Karypis (ex. 2005 "Karypis Decl.") that it was not permitted to submit along with its prior filings.[1]

In addition, Nework-1 quotes dozens of admissions from the deposition of Petitioner's own Declarant, Dr. Pierre Moulin (ex. 2006), to corroborate most of the arguments made in this Response.

First, the Response addresses claim constructions relevant to this IPR.

[1] To avoid duplication, to make the proceeding more efficient, and to avoid confusion (by citing multiple versions of Karypis Declarations), Patent Owner provides a single Karypis Declaration (ex. 2005) for the four related IPRs (IPR2015-00343, IPR2015-00345, IPR2015-00347, and IPR2015-00348). As a result, certain paragraphs of Declaration (¶¶15-24, 34-37) are not directly relevant to this IPR.

Second, the Response addresses each of the two grounds at issue in this IPR.

Third, the Response demonstrates that Petitioner cannot supplement its evidence and arguments attempting to establish its *prima facia* case.

I. Claim Constructions.

The Board identified the following claim constructions in its Decision:

non-exhaustive search	"a search that locates a match without a comparison of all possible matches" Decision at 7.
neighbor search	"a search that identifies a close, but not necessarily exact or closest, match" Decision at 8

Each construction is addressed in turn.

A. non-exhaustive search.

1. The Board's construction—"a search that locates a match without a comparison of all possible matches"—is correct.

A non-exhaustive search is a search that uses an intelligent algorithm to locate a match without comparing the work to all records in the database by narrowing the database to a subset of potential matches. Karypis Decl. ¶¶79-88. Moulin Decl. ¶12 ("algorithms that increased efficiency by intelligently searching only a subset of potential matches (*i.e.*, 'non-exhaustive' algorithms)"). For example, if there are 100 records in a database, a non-exhaustive search could use an intelligent algorithm to exclude 75 records from the search such that only 25

would be searched. Karypis Decl. ¶80. As the '411 specification states, these non-exhaustive "forms of matching include those based on clustering, kd-trees, vantage point trees and excluded middle vantage point forests" that do not systematically compare the work to be identified to each record. '441, 8:65-9:1. Each example uses an intelligent algorithm to narrow the database to a subset of potential matches. Karypis Decl. ¶80.

A non-exhaustive search can be contrasted with an exhaustive search. An exhaustive search systematically checks whether each record matches the work to be identified until a match is found, "perhaps halting the search when the first match is found." '441, 8:60-62; Karypis Decl. ¶81. An exhaustive search "consists of systematically enumerating all possible candidates for the solution and checking whether each candidate satisfies the problem's statement." Ex. 2001. If there are 100 records in the database, an exhaustive search does not narrow the potential matches but instead systematically compares the work with each record to determine a match (if there is one). Karypis Decl. ¶82.

2. The Board properly rejected Petitioner's assertion that a non-exhaustive search should be construed as "a search that locates a match without conducting a brute force comparison of all possible matches, and all data within all possible matches.[2]

[2] *See* Pet. at 5-6; Decision at 6-7; Karypis Decl. ¶¶83-88.

The "all data" clause (underlined above) in Petitioner's construction would improperly include as a non-exhaustive search any search that did not compare "all data" in each record, even if the search were a brute force comparison of each record in the database. As an illustrative example, assume the work to be identified "ABC" is compared with all records in a library, including record "DEF." When comparing "ABC" with "DEF," the algorithm determines that there is no match between "ABC" and "DEF" after just comparing the first letter of the work "A" with the first letter of the record "D." If the algorithm does not unnecessarily compare the second and third letters, then according to Petitioner, the search is non-exhaustive even though every record is compared. Karypis Decl. ¶83.

Petitioner's "all data" clause is improper because it is:

- inconsistent with how non-exhaustive is used in the '441 patent which describes an exhaustive search as one where the search compares the work to all "N entries," not all data within all "N entries" (*see e.g.,* '441, 8:55-64); and
- not part of the ordinary meaning of non-exhaustive search (*see* ex. 2001); Karypis Decl. ¶85.

Objective sources confirm that an exhaustive search systematically compares the work with each record in a database, not all data within each record, for example:

> In computer science, brute-force search or exhaustive search, also known as generate and test, is a very general problem-solving technique that consists of systematically enumerating all possible candidates for the solution and checking whether each candidate satisfies the problem's statement.

Ex. 2001[3] —each candidate is checked, not "all data" within each candidate.

Petitioner's Declarant confirmed that a non-exhaustive search searches a subset of potential matches, not a subset of all data within all potential matches:

> content recognition schemes typically employed search algorithms that increased efficiency by intelligently searching only a subset of potential matches (*i.e.*, 'non-exhaustive' algorithms).

Moulin Decl. ¶12.

> persons skilled in the art routinely employed more efficient searches that did not conduct a comparison of every single item in a database, sometimes referred to as non-exhaustive searches.

Moulin Decl. ¶40.

To support its proposed construction, Petitioner relies solely on the conclusory assertions in its Declaration (Pet. 5-6) and fails to present any objective evidence (*e.g.*, citations to the '441 patent's specification and prosecution history, dictionary definitions, or treatises). Accordingly, the Board properly rejected Petitioner's "all data" clause:

> Patent Owner points out … the written description of the '441 patent

[3] Emphasis added throughout this Response, unless otherwise noted.

> discusses both exhaustive and nonexhaustive searches. .... Patent Owner also supplies an example it contends shows the ordinary meaning of "exhaustive search" or "brute-force search." Neither discussion mentions the evaluation of all data within each possible match.

Decision at 6-7.

B. neighbor search.

The Board properly construed a neighbor search as "identifying a close, but not necessarily exact or closest, match." Decision at 7; Karypis Decl. ¶89-93. Petitioner and its Declarant agree with the Board's construction of neighbor search. *See e.g.*, Pet. at 6; Moulin Decl. ¶45; Moulin Depo. 250:2-5. There are two relevant features of a neighbor search based on this construction:

Feature 1: If a search necessarily identifies an exact or the closest match (*i.e.*, the search is designed to guarantee that an exact or the closest march is identified each time the search is performed), it is not a neighbor or near neighbor search because it is not a search that "identif[ies] a close, but not necessarily exact or closest, match." Rather, such a search necessarily identifies an exact or the closest match. Karypis Decl. ¶92.

Feature 2: If a search that necessarily identifies an exact or the closest match (*e.g.*, Match 1) but also identifies other matches that, by definition, are not the closest match (Match 2, Match 3, Match 4), the search still necessarily

identifies an exact or the closest match (Match 1) and therefore cannot be the claimed neighbor or near neighbor search. Karypis Decl. ¶93.

II. '441 Ground 1: The instituted claims of the '441 patent are not anticipated by Conwell.

Each independent claim of the '441 patent includes a limitation "comparing [the extracted features of the work to be identified with extracted features of the reference works] using a non-exhaustive neighbor search:"

> (c) identifying, by the computer system, the first electronic work by comparing the extracted features of the first electronic work with the first electronic data in the database using a non-exhaustive neighbor search;

'441 claim 1, element [c];

> (c) identifying, by the computer system, a matching reference electronic work that matches the first electronic work by comparing the first electronic data with the second digitally created compact electronic representation using a non-exhaustive neighbor search;

'441 claim 13, element [c];

> (c) identifying, by the computer system, a matching reference electronic work that matches the first electronic work by comparing the first electronic data with the second digitally created compact electronic representation of the first electronic work using a non-exhaustive neighbor search;

'441 claim 25, element [c]. Ground 1 fails because the search disclosed in Conwell is neither (1) a neighbor search, nor (2) a non-exhaustive search. Karypis

Decl. ¶¶331-385, 409-419. Each deficiency is addressed in turn.

A. neighbor search (claims 1, 13, 25).

The search disclosed in Conwell does not "compar[e] [the extracted features] using a …neighbor search." As explained above, a neighbor search is a search "identifying a close, but not necessarily exact or closest, match." Section I(B); Decision at 8; Karypis Decl. ¶89. Each independent claim of the '441 patent requires "comparing [the extracted features] using [a] neighbor search"—that is:

- Claim 1: "comparing [1] the extracted features of the first electronic work with [2] the first electronic data in the database using a non-exhaustive neighbor search."
- Claim 13: "comparing [1] the first electronic data with [2] the second digitally created compact electronic representation using a non-exhaustive neighbor search."
- Claim 25: "comparing [1] the first electronic data with [2] the second digitally created compact electronic representation of the first electronic work using a non-exhaustive neighbor search."[4]

[4] In Petitioner's analysis, the "extracted features" ('441, claim 1) and "compact electronic representations" ('441, claims 13 and 25) are "synonymous"—both constitute the hashed identifiers of the extracted features of

Using claim 1 as an example:

(c) identifying, by the computer system, the first electronic work by comparing the extracted features of the first electronic work with the first electronic data in the database using a non-exhaustive neighbor search;

'449, claim 1

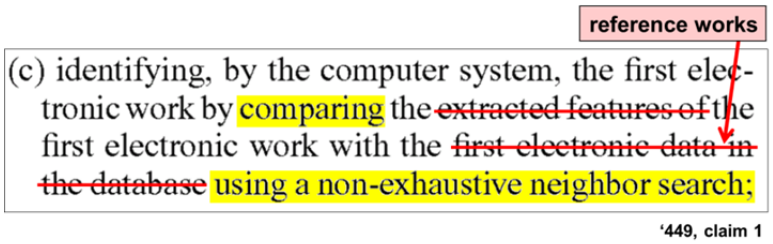
the claim requires "comparing… using a … neighbor search" (highlighted in yellow) [1] "the extracted features of the first electronic work" (highlighted in green); with [2] "the first electronic data in the database" (highlighted in orange).

The claimed comparing does not compare (a) the work to be identified, with (b) a record or records in the database. Karypis Decl. ¶335. Again, using claim 1 as an example, the claimed "comparing… using [a] neighbor search" does not compare: (a) "one or more reference electronic works" (*i.e.*, the references in the database); with (b) "a first electronic work" (the work to be identified). Such comparing would require redrafting the '441 claims. For claim 1, "the extracted features of the first electronic work" would need to be replaced with the "first electronic work;" and "first electronic data in the database" would need to be

---

the underlying reference works in the database of reference works and works to be identified. Moulin Decl. ¶85; Moulin Depo. 182:5-10. For simplicity, this Response refers to both the "extracted features" ('441, claim 1) and "compact electronic representations" ('441, claims 13 and 25) as the "extracted features."

replaced with the "reference electronic works," as reflected in the following redraft:



Karypis Decl. ¶335. The redrafted claim is not the claimed invention. Instead, comparing the extracted features—[1] the extracted features of the work to be identified with [2] the extracted features of the reference works—using a neighbor search. Karypis Decl. ¶335.

Moreover, the claims do not claim a process that simply results in or has the effect of identifying a neighbor of the work to be identified using any possible comparison. Karypis Decl. ¶336. Rather, the claims require comparing [1] the extracted features of the work to be identified, and [2] the extracted feature of the reference works "using [a] neighbor search."

The Petitioner and Declarant (and the Board in instituting Ground 1) relied on the hashes of the extracted features of the work to be identified and the records in the database (reference works) in Conwell (the "identifiers") as the extracted features that are compared to establish the neighbor search. Karypis Decl. ¶337; *see* Pet. at 23 (the hashed "identifiers extracted from a reference works" are

compared to the hashed "extract identifiers from content"); Moulin Decl. ¶85; Conwell, 1:65-67 ("some or all of the content data is processed by a hashing algorithm to yield a 128 bit identifier corresponding to that content.")

15 So you start from the content data, and
16 you map that to a 128-bit identifier, which is a
17 compact electronic representation of the content.

Moulin Depo. 180:7-17.

22 Is it your testimony that a 128-bit hash
23 identifier constitutes a digitally created compact
24 electronic representation?
25 A Yes.

Moulin Depo. 180:22-25.

The comparison relied on by Petitioner as disclosing the claimed neighbor search is illustrated in the following diagram (*see* Karypis Decl. ¶338):

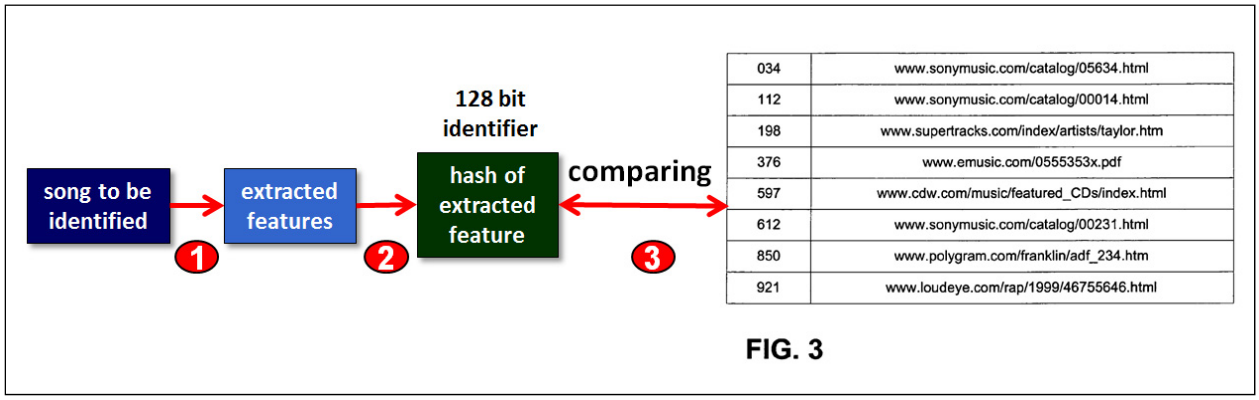


FIG. 3

1 features are extracted from the work—the song to be identified.

2 the extracted features are hashed to generate a 128 bit identifier.

3 the hashed identifier is compared to the hashed extracted features of the records using a lookup table (as illustrated in Figure 3 of Conwell). The 3-digit numbers on the left side of Figure 3 (*e.g.*, "034," "112," "198") are examples of decoded hashed identifiers of the records in the database ("decod[ed] identifier from audio content") that is compared with the hashed identifier of a work to be identified. Conwell, Figure 3; 3:46-50; Karypis Decl. ¶338. Petitioner's Declarant, Dr. Moulin, confirmed that Petitioner's anticipation theory is based on comparing the hashed extracted features of the work to be identified with the hashed extracted features of the reference works (this comparing is labeled 3 in the diagram above).

> Is it your testimony that Conwell teaches Element (c) when it teaches using a robust hash approach, where it uses a lookup table to compare one hash value of the unknown work to hash values that are in the database?
>
> A Yes.

Moulin Depo. 194:10-15; 263:22-264:9; 199:23-200:5; 191:8-12; 173:25-174:6; 170:14-18; 185:20-24. Extracting the features is part of a preprocessing step rather than comparing features:

> A As I said -- okay. You start from the work, let's say audio. You extract features. That is a representation. Okay? That's a preprocessing step. You obtain the features. Then you extract a hash, which is a compacted electronic representation. It's bits.

Moulin Depo. 169:4-13.

Conwell exclusively teaches comparing the hashed extracted features using a lookup table which uses an exact match comparison rather than a neighbor search. Karypis Decl. ¶341; Conwell, Figure 3; 3:43-62 ("Referring to FIG. 3, an exemplary Registry database can be conceptualized as a large look-up table."). A device decodes a hashed "identifier from audio content" to be identified which is compared to the hashed identifiers in the lookup table. Conwell, 3:43-62. If the decoded identifier of the work to be identified matches one of the decoded identifiers in the lookup table, "the user's web browser is then directed to that URL." Conwell, 3:43-62.

The lookup table disclosed in Conwell used to compare the hashed extracted features of the reference works and the work to be identified uses an exact search rather than a neighbor search. Karypis Decl. ¶341. If there is an exact match, the search identifies that match. If there is a neighbor—*i.e.*, "a close, but not necessarily exact or closest, match," the search will not identify such neighbor.

A neighbor search—a search that identifies "a close, but not necessarily

exact or closest, match" —can identify an exact match; however, it must also be able to identify a close match. The lookup table search in Conwell cannot identify a close match. Karypis Decl. ¶342.

If the hashed identifier of the work to be identified does not have an exact match with the hashed identifiers in the reference database, the result will not be an exact match and no match will be identified even if there is a close or closest match. Karypis Decl. ¶342. For example, as illustrated in Figure 3 of Conwell, if the identifier of the work to be identified is 199, it will not result in a match even though it is very close to 198, a match for the URL www.supertracks.com... in Figure 3. Moulin Depo. 188:13-20.

Petitioner's Declarant confirmed that comparing the hashed extracted features of the reference database with the hashed extracted features of the work to be identified "will never" identify a neighbor—it is will never identify "a close, but not necessarily exact or closest, match:"

Q If -- in Conwell, when it does a comparison of the extracted features of the first electronic work with the first electronic data, it's comparing a hash value with a set of hash values; right?
A Yes.
Q That comparison is always going to produce either an exact match or no match; right?
A Yes.
Q That -- that comparison will never return a suggestion that, "Here's something that doesn't quite match, but it's close"?
A That's correct.

Moulin Depo. 264:13-25; 259:20-260:6. Moreover, this exact match lookup table is the only search disclosed in Conwell. Conwell, 3:43-62; Karypis Decl. ¶343.

So the only search that's disclosed in -- in Conwell using the hash values is one where we look for an exact match, and it's either there or it isn't; is that right?
A Well, an exact match of that identifier. Like 198, is it there or not? If I have 199, I would say it's not there.

Moulin Depo. 201:4-10; 199:18-21.

As confirmed by Petitioner's Declarant, this lookup search disclosed in Conwell is not a neighbor search because the hashed identifiers disclosed in Conwell are never used to perform a neighbor search as required by each claim of the '441 patent:

If we look at the search you've identified as constituting the neighbor search, it's a search wherein we start with a hash value and then go to a lookup table to see if it appears in that table; is that right?

A Yes.

Q Is the result of that going to be either an exact match of the hash or a determination that it does not exist?

A Yeah. It's either the table or it's not.

Q Are we ever going to have a circumstance where we look in the table for that hash value and we conclude that the hash value does not appear, but here's one that's pretty close, and we'll return that one?

A It's not -- it does not -- Conwell does not disclose that.

Moulin Depo. 200:11-201:2;

Q I want you to assume that we define a neighbor search as one that produces a close but not necessarily exact match, so that a search that always produces an exact match would not be a neighbor search.

With that definition, if we have a search that consists of a hash value lookup in a table of reference hash values, is that lookup a neighbor search?

MR. ELACQUA: Objection.

THE WITNESS: Under your flawed definition, it will not be a neighbor search.

Moulin Depo. 262:10-263:14.

While the search disclosed in Conwell may have the result of identifying a neighbor of the work to be identified, it does not do so by "comparing [the extracted features] using [a] neighbor search" as required by the '441 claims. Karypis Decl. ¶346.

> A prior art reference can only anticipate a claim if it discloses all the claimed limitations "arranged or combined in the same way as in the claim." *Wm. Wrigley Jr. Co. v. Cadbury Adams USA LLC,* 683 F.3d 1356, 1361 (Fed.Cir.2012) (*quoting Net MoneyIN, Inc. v. VeriSign, Inc.,* 545 F.3d 1359, 1370 (Fed.Cir.2008)).

Kennametal, Inc. v. Ingersoll Cutting Tool Co., 780 F.3d 1376, 1381 (Fed. Cir. 2015). "Even where a prior art device is the 'functional equivalent' of a patented product, it does not anticipate unless it discloses the structure required by the asserted claims." *Old Reliable Wholesale, Inc. v. Cornell Corp*., 635 F.3d 539, 545 (Fed. Cir. 2011). If a process disclosed in a reference achieves the same result as a claimed invention, the reference does not anticipate the claimed process unless the disclosed process achieves that same result using the claimed steps. As Petitioner's expert acknowledged:

21	Q You might be able to have a certain
22	process that would result in identifying neighbors,
23	but it wouldn't use a neighbor search; right?
24	A That's possible.

Moulin Depo. 266:21-24; 266:11-15. Had the search process taught in Conwell

identified a neighbor of the work to be identified (*i.e.*, achieves the same result as the '441 claims), it would still not anticipate the '441 claims because it would achieve that result using a different process—*i.e.*, using an exact match comparison of hashed extracted features. Karypis Decl. ¶¶347-348; *see* Moulin Depo. 260:7-21.

The Petition, Declaration, and corresponding charts fail to demonstrate that Conwell teaches a neighbor search. As support that Conwell discloses the claimed "comparing [the extracted features] using [a] neighbor search," the Petition relies on the highlighted sentence in the following paragraph (and the corresponding citations to Conwell):

> Third, the '441 Claims require "identifying . . . the first electronic work by comparing the extracted features of the first electronic work with the first electronic data in the database using a non-exhaustive neighbor search." Ex. 1001 at Claims 1, 13, 25. Conwell teaches table look-up using the identifier decoded from the electronic work. *See* Ex. 1009 at 3:43-62, Figs. 1, 3. Because the hash algorithm generates the same output identifier for similar, but non-identical, inputs, the table look-up will return similar "neighbor" results even when the input work is not identical to the reference work. *Id.* at 4:64-5:3; Ex. 1004 at ¶ 86. This search is non-

Pet. at 23. The Petition, Declaration, and corresponding charts fail to establish that Conwell teaches the claimed "comparing [the extracted features] using [a]

neighbor search." Karypis Decl. ¶354. In fact, the Petition, Declaration, and charts fail to even address the actual claim language. The claims require:

> (c) identifying, by the computer system, the first electronic work by comparing the extracted features of the first electronic work with the first electronic data in the database using a non-exhaustive neighbor search;

'441 claim 1, element [c]. This claim language and concept is completely absent from Petitioner's analysis. The Petition and Declaration fail to address what is actually being compared using the neighbor search. The only comparing identified in the Petition is "performing a lookup in a hash table" (Pet. at 24), which, as demonstrated above, is an exact match comparison, not a neighbor search. Karypis Decl. ¶355.

Instead, the Petition states that "the '441 Claims require 'identifying…the first electronic data in the database 'using a non-exhaustive neighbor search.'" Pet. at 23. This mischaracterization of the claims ignores the actual claim language. Karypis Decl. ¶355. The claims do not simply require "'identifying' the unknown work 'using a non-exhaustive neighbor search;'" rather, they require:

> (c) identifying, by the computer system, the first electronic work by comparing the extracted features of the first electronic work with the first electronic data in the database using a non-exhaustive neighbor search;

And the only comparing of extracted features identified in the Petition (and disclosed in Conwell) is an exact match comparison—performing a lookup in a

hash table—"a single lookup in a numerically sorted lookup table that uses hash identifiers as an index" (Pet. at 23-24)—not a non-exhaustive neighbor search as required by the '441 claims

Petitioner observes that the system disclosed in Conwell may achieve the same result as the result of the '441 claims: "Because the hash algorithm generates the same output identifier for similar, but non-identical, inputs, the table look-up will return similar 'neighbor' results even when the input work is not identical to the reference work." Pet. at 23 (*citing* Conwell, 4:64-5:3). But achieving the same result using a process that is different than that claimed in the '441 claims does not anticipate the '441 claims. The Declarant states that "because Conwell also teaches the use of a hash algorithm such that similar, but non-identical inputs generate the same output identifier, the table look-up will return similar 'neighbor' results…" Moulin Decl. ¶86. That the search generates similar neighbor results, however, does not address the claim language and, in particular, what is actually being compared using the claimed neighbor search. Karypis Decl. ¶356.

The two passages from Conwell cited in the Petition and Declaration do not demonstrate the claimed "comparing [the extracted features] using [a] neighbor search." Each passage is addressed in turn:

Conwell, 3:43-62: This passage identifies the exact match look-up table illustrated in Figure 3 which, as described above, is an exact match search rather

than a neighbor search. Karypis Decl. ¶358. This passage does not teach (and the Petition and Declaration do not contend that it teaches) "comparing [the extracted features] using [a] neighbor search."

Conwell, 4:64-5:3: Conwell, 4:64-5:3. This passage states that similar works might correspond to the same hash, *e.g.*, the 4th entry in the table of Figure 3: "376." The passage does not state that the hash identifier (*e.g.*, extracted feature 376) is then compared to the hash of the reference work using the claimed neighbor search. Rather, the hash identifier (extracted feature) is compared using an exact match lookup table. Karypis Decl. ¶359.

The Board's concerns: The following addresses the Board's specific concerns (identified in its Decision) with respect to whether Conwell discloses a neighbor search. In instituting Ground 1, the Board recognized that Conwell discloses a search that compares the hashed output identifier (*i.e.*, "digitally created compact electronic representation of the first electronic work") with the records in the database using an exact match lookup but instituted Ground 1 because the "user would be directed to the same URL for both Song A and Song A_1, thereby matching both Song A and Song A_1":

> We are not persuaded by Patent Owner's argument. In contrast to Levy, Conwell discloses that its hash algorithm generates the same output identifier for similar, but non-identical, inputs, as discussed by Petitioner. *See* Pet. 23 (citing Ex. 1009, 4:64–5:3). Patent Owner acknowledges the citation, but argues that the same "hashed output identifier is compared to the records using an exact match lookup." Prelim. Resp. 29. Patent Owner continues that such an exact match search is not the same as a "neighbor" search as recited in the claims. On the present record, however, we are persuaded that similar, but non-identical, inputs (e.g., Song A and Song A_1, in Patent Owner's example, *id*. at 28), would produce the same identifier, and the hash table lookup in Conwell would provide the same URL for both. In other words, the search, or lookup, would "identify[] a close, but not necessarily exact or closest, match," such that a user would be directed to the same URL for both Song A and Song A_1, thereby matching both Song A and Song A_1. Although it appears that such a search process is different from that disclosed in the written description of the '441 patent, we are persuaded, on the present record, that it meets the discussed limitations recited in the independent claims.

Decision at 11-12. As demonstrated above, the search process disclosed in Conwell is not only "different from that disclosed in the written description of the '441 Patent" (Decision at 12), it is also different than the process claimed in the '441 patent. Karypis Decl. ¶361.

Each independent claim requires "comparing [the extracted features] using [a] neighbor search:"

(c) identifying, by the computer system, the first electronic work by comparing the extracted features of the first electronic work with the first electronic data in the database using a non-exhaustive neighbor search;

'441 claim 1, element [c]. Conwell exclusively discloses doing such comparing using an exact match lookup table. Conwell, 3:43-62; Moulin Depo. 200-11-201:2; Karypis Decl. ¶361. The Board's analysis addresses alternative claim language that does not require comparing the extracted features using a neighbor search but instead simply identifies a neighbor (*e.g.*, Song A_1) independent of the actual search being performed. That the system disclosed in Conwell may achieve the same result—*i.e.*, map both Song A and Song A_1 to the same identifier—does not convert the exact match comparison of the hashed identifiers disclosed in Conwell into a neighbor search of such identifiers as required by each independent claim. Karypis Decl. ¶362.

B. non-exhaustive search (claims 1, 13, 25).

In instituting Ground 1, the Board did not specifically address whether Conwell discloses the claimed non-exhaustive search. Decision at 11-12. Conwell does not disclose the claimed non-exhaustive search. Karypis Decl. ¶¶363-385, 441-419.

As explained above, a non-exhaustive search is "a search that locates a match without a comparison of all possible matches." Section I(A); Decision at 6-7; Karypis Decl. ¶79. A non-exhaustive search uses an intelligent algorithm to

reduce the number of potential matches. By contrast, an exhaustive search uses brute force to compare the work to be identified with each record in the database, "perhaps halting the search when the first match is found." '441, 8:60-62; *see* Section I(A)(1)); Karypis Decl. ¶81. Conwell does not teach a non-exhaustive search.

As illustrated in Figure 3, Conwell teaches identifying a match using a look-up table. Conwell, 3:43-44 ("Referring to FIG. 3, an exemplary Registry database can be conceptualized as a large look-up table."). Conwell also discloses (in a section addressing how the table can be maintained) that the look-up table can include "entries … sorted, by identifier." Conwell, 5:59-61. These disclosures in Conwell are neither an expressed nor an inherent disclosure of a non-exhaustive search. Karypis Decl. ¶365.

034	www.sonymusic.com/catalog/05634.html
112	www.sonymusic.com/catalog/00014.html
198	www.supertracks.com/index/artists/taylor.htm
376	www.emusic.com/0555353x.pdf
597	www.cdw.com/music/featured_CDs/index.html
612	www.sonymusic.com/catalog/00231.html
850	www.polygram.com/franklin/adf_234.htm
921	www.loudeye.com/rap/1999/46755646.html

FIG. 3

Express: Conwell does not expressly disclose using the look-up table to conduct a non-exhaustive search—*i.e.*, using an algorithm that increases efficiency by intelligently searching only a subset of potential matches rather than a brute force search. There are many potential ways to use Conwell's look-up table to

identify a match. Karypis Decl. ¶366. For example, one possible approach would be to compare the hashed identifier of the extracted features of the work to be matched with the first entry in the look-up table, then with the second entry, and so on—*i.e.*, an exhaustive search.

Using Figure 3 as an illustrative example, if the hashed identifier of the work to be identified is 612, the identifier could be compared with the first entry 034, resulting in no match; the search would then compare the identifier to the second entry 112, again resulting in no match; and so on until the search reached 612—a match. Conwell, Figure 3; Karypis Decl. ¶366. Such a search would be exhaustive rather than non-exhaustive because it systematically checks whether each record matches the work to be identified until a match is found.[5]

As another example, if the hashed identifier of the work to be identified were 744, the identifier could be compared with the first entry 034, resulting in no match; the search would then compare the identifier to the second entry 112, again

[5] This illustrative search would be exhaustive (rather than non-exhaustive) whether or not the searched stopped after identifying a match. Karypis Decl. ¶367. An exhaustive search systematically checks whether each record matches the work to be identified until a match is found, "perhaps halting the search when the first match is found." '441, 8:60-62.

resulting in no match; and so on until the entire table is compared. Because there is no exact match, no match would be identified. Karypis Decl. ¶368. This approach of using the sorted lookup table is an exhaustive (rather than non-exhaustive) search, because its searches all entries until a match is found rather than using an algorithm that increases efficiency by intelligently searching only a subset of potential matches. Karypis Decl. ¶368.

While there are ways to search the lookup table disclosed in Conwell using a non-exhaustive approach, Conwell does not disclose such an approach. Karypis Decl. ¶369. In fact, Conwell does not teach any specific method of conducting the exact match comparison using the disclosed lookup table. Karypis Decl. ¶369. The only descriptions in Conwell regarding the search to be used are the following generic statements: (1) the "database responds…."—which does not disclose a non-exhaustive search; and (2) "queries the database"—which also does not disclose a non-exhaustive search:

> Referring to FIG. **3**, an exemplary Registry database can be conceptualized as a large look-up table. Each active record includes an identifier and a corresponding URL. When a consumer uses a suitably equipped device (e.g., a personal computer, or wireless internet appliance) to decode an identifier from audio content and send the identifier to the database, the database responds by returning the URL corresponding to that identifier back to the user device. The user device then directs an internet browser to that URL. By such arrangements, music (e.g., in MP3 format) can serve as a portal to a web site dedicated to the music artist, a web site giving concert schedules for the artist, a web site offering CDs, etc.
>
> In the FIG. **3** example, if the device decodes the identifier '376' from an MP3 file, and queries the database with this data, the database returns the URL www.emusic.com/0555353x.pdf. The user's web browser is then directed to that URL. (For expository convenience, the identifiers are assumed to be in the range 0–1023. In actual implementations, a much larger range would usually be used.)

Conwell, 3:43-62; Karypis Decl. ¶369.

Inherent: Conwell also does not inherently disclose using the look-up table to conduct a non-exhaustive search. *First*, the Petition does not rely on any theory that Conwell inherently discloses a non-exhaustive search. Pet. at 1-60. Accordingly, because the Petitioner did not present this theory, it cannot be a basis for finding the '441 claims unpatentable. *Second*, Conwell does not inherently disclose a non-exhaustive search. "A claim limitation is inherent in the prior art if it is necessarily present in the prior art, not merely probably or possibly present." *Akamai Technologies, Inc. v. Cable & Wireless Internet Servs., Inc.*, 344 F.3d 1186, 1192 (Fed. Cir. 2003). As noted above, Conwell does not expressly disclose comparing the hashed identifiers to the potential matches in the look-up table using

a non-exhaustive search. Karypis Decl. ¶370. And, as illustrated in the example above, such a search is not "necessarily present"—there are many ways to compare a hashed identifier to the reference hashed identifiers in a look-up table other than using a non-exhaustive search. While it might be possible or even probable that one skilled in the art would use a non-exhaustive approach to conduct a search using the look-up table disclosed in Conwell, such a possibility or even probability does not establish an inherent disclosure. Karypis Decl. ¶371.

The Petition, Declaration, and corresponding charts fail to demonstrate that Conwell teaches comparing the extracted features using a non-exhaustive search. As support that Conwell discloses the claimed "comparing [the extracted features] using a non-exhaustive … search," the Petition relies on the following:

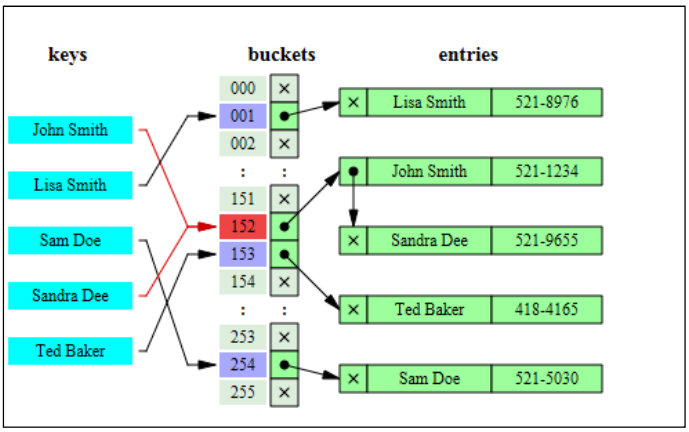
> identical to the reference work. *Id.* at 4:64-5:3; Ex. 1004 at ¶ 86. This search is non-exhaustive because it does not require a brute force comparison of all possible hashes, but rather requires a single lookup in a numerically sorted lookup table that uses hash identifiers as an index. Ex. 1009 at 3:43-50, 5:58-64; Ex. 1004 at ¶ 86.

Pet. at 23-24. Petitioner asserts that Conwell discloses a non-exhaustive search because using the lookup-table "does not require a brute force comparison of all possible hashes," but instead "requires a single lookup in a numerically sorted lookup table." Pet. at 23-24. As explained above, Conwell does not teach any particular search algorithm for searching the lookup-table and certainly does not

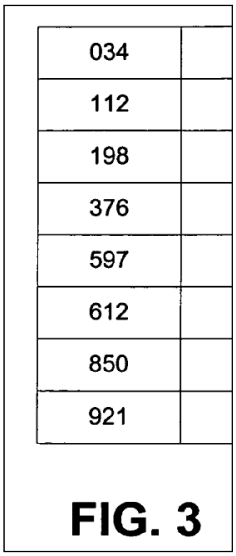
disclose a search algorithm that is not a brute force algorithm or an algorithm that requires a "single lookup." Conwell, 3:43-62; Karypis Decl. ¶377. As explained above, Conwell provides no details as to how the exact-match search between the hashed identifier of the work to be identified and the hashed identifiers in the reference database is actually performed. The entire description of the exact match search is these two generic statements highlighted in the Conwell passage presented above. Conwell, 3:43-62. Neither of these statements—"the database responds" or "queries the database"—discloses any particular exact match search, much less a "single lookup" search. Karypis Decl. ¶378.

Moreover, the illustrative example disclosed in Conwell cannot be used to identify a match using a single lookup. Hashes can be stored in a standard database structure using [1] just records for the identified keys (as in Figure 3 of Conwell), or [2] pre-allocating records for all possible keys (often referred to as a hash table). The second approach—the hash table—could possibly be used to identify a match using a single lookup. For example, assume that a hash algorithm generates values from 1-10 and that the reference works in the reference library hash to 1, 4, 7, and 10. A hash table that could be used to perform a "single lookup" would look like this:

Key	URL
1	www....
null	null
null	null
4	www...
null	null
null	null
7	www...
null	null
null	null
10	www....



Karypis Decl. ¶380. The hash table can be used to look up a record directly because there is a line item for every possible number. For example, if the query hashes to a value of 5, the search could go directly to the 5th record.



Figures 3 and 4 of Conwell, however, do not have this hash table structure but instead disclose a lookup table that includes gaps (*e.g.*, gaps between 034 and 112). As a result, the lookup table in Conwell does not disclose identifying a match using a single lookup as suggested by Petitioner. Karypis Decl. ¶381.

The two cited passages from Conwell relied on in the Petition do not disclose any particular search algorithm using the sorted lookup-table, much less an algorithm that is a non-exhaustive search. Each passage is addressed in turn.

Conwell, 3:43-50: This passage does not disclose the claimed non-exhaustive search and does not disclose a "single lookup." Rather, this passage states that the "Registry database can be conceptualized as a large look-up table" and states that the "database responds…." Karypis Decl. ¶383. The passage says nothing about how the "Registry database" is searched, and specifically does not disclose a non-exhaustive search or a single lookup as suggested by Petitioner. Karypis Decl. ¶383. Rather, the passage simply states that, when a consumer uses a device to "send the identifier to the database, the database responds by returning the URL corresponding to that identifier back to the user device." Conwell, 3:43-50. The passage says nothing about the search process that uses the "conceptualized … large look-up table" to identify the corresponding URL. Karypis Decl. ¶383.

Conwell, 5:58-64: Like the first passage, this passage also does not disclose the claimed non-exhaustive search. Karypis Decl. ¶384. Rather, this passage states that the (1) the maintenance of the table 12 is well understood by those skilled in data structures, (2) the entries can be sorted by identifier, and (3) the system may be keyed by identifier, sound, and artist. None of these disclose how the search is actually performed, much less that the undisclosed search technique is non-exhaustive. Karypis Decl. ¶385. The first three sentences simply describe how the table 12 is maintained and say nothing about how the table is searched.

Karypis Decl. ¶385. The final sentence addresses how the system is "keyed by identifier, song, and artist." Again this sentence says nothing about how the table is actually searched but rather refers to how the database is maintained. Karypis Decl. ¶385.

III. '441 Ground 2: The instituted claims of the '441 Patent are not obvious over Ghias and Philyaw.

If a combination of two references fails to teach an important claimed element, it is not possible for that combination to render the claim obvious. That is, assuming one of ordinary skill would have thought to combine prior art references, that combination would still be missing an important claim element and therefore, even with the combination, one of ordinary skill would still not possess the invention.

All independent claims of the '441 patent include the limitation "non-exhaustive neighbor search." '441, claims 1, 13, and 25. For Ground 2, Petitioner relies exclusively on Ghias for the clamed "non-exhaustive neighbor search." Pet. at 49-60; Moulin Depo. 375:18-20; 373:23-374:3; 374:20-25. Ground 2 fails because Ghias does not disclose (1) a non-exhaustive search, and (2) a neighbor search. Karypis Decl. ¶¶387-407, 421-423, 236-245, 266-290. Each deficiency is addressed in turn.

A. non-exhaustive search (claims 1, 13, 25).

As explained above, a non-exhaustive search is "a search that locates a match without a comparison of all possible matches." Section I(A)(1); Decision at 7; Karypis Decl. ¶¶79-88. Ghias teaches an exhaustive search that compares the work to be identified (user input 23) with "all the songs" in the database—*i.e.*, "all possible matches:"

> In order to search the database, songs in the database **14** are preprocessed to convert the melody into a stream of the previously discussed U,D,S characters, and the converted user input (the key **23**) is compared with all the songs.

Ghias, 5:66-6:2; Karypis Decl. ¶388. As Petitioner's Declarant acknowledged when addressing the paragraph from Ghias quoted above:

> Q But what you've identified in Ghias, in each case, Ghias is going to be searching each of the records in the database; right?
> A Yes.
> Q Would it be the case that the Ghias search is going to be performing a comparison to each of the melodies that are possible matches in the database?
> A It does a comparison, yes, to each of them.

Moulin Depo. 323:4-13; 339:23:-340:5; 340:6-9. The user input (23) is not compared with some songs in melody database (14); rather, it "is compared with all the songs." Ghias does not disclose any search algorithm which would not compare the query to every record in the reference data set. Karypis Decl. ¶267.

Petitioner's Declarant confirmed that the search disclosed in Ghias compares the song to be identified with each record in the database and is therefore non-exhaustive under the Board's proper correct construction of the phrase—"a search that locates a match without a comparison of all possible matches" (*see* Section I(A)(1)); Decision at 6-7):

Let's assume we define "exhaustive search" as a search that does a comparison to each of the possible matches in the database that we're searching over.

In that instance, if we have that definition, would you agree that Ghias does perform an exhaustive search?

MR. ELACQUA: Objection.

THE WITNESS: Under your flawed definition, yes, it would.

Moulin Depo. 327:3-12.

Q Now, if it's the case that -- if we define -- let's define "nonexhaustive" to be the opposite; that is, "nonexhaustive" is a search that does not perform a comparison to each of the possible matches in our record database.

Would you agree, then, that Ghias does not disclose a nonexhaustive search?

MR. ELACQUA: Objection.

THE WITNESS: I would have to think -- to read the patent again if he considers the possibility of not searching the entire database. Clearly, it's intended to search the entire database.

BY MR. DOVEL:

Q Ghias is intended to search the entire database?

A Yes.

Moulin Depo. 327:14-328:4.

The Petition fails to demonstrate that Ghias teaches comparing the extracted features using a non-exhaustive search. As support for this element, the Petition exclusively relies on the following:

between successive notes of the melody."). Ghias further discloses that this search may be non-exhaustive: "it is considered desirable to use an efficient approximate pattern matching algorithm" rather than an algorithm that is guaranteed to yield a match. *Id.* at 6:7-11, 6:23-35 ("Several Algorithms have been developed that address the problem of approximate string matching. Running times have ranged from 0(mn) for the brute force algorithm to O(kn) or O(nlog(m)"). This search

Pet. at 47-48; Moulin Depo. 332:17-333:2. The Petition fails to: (a) apply Petitioner's construction (or any other construction) of non-exhaustive to Ghias; or (b) explain how an approximate string matching algorithm is expressly or inherently a non-exhaustive search. Karypis Decl. ¶275. Neither the assertions nor the passages from Ghias disclose the claimed non-exhaustive search. Karypis Decl. ¶273. Each is addressed in turn.

Passage 1: Ghias, 6:7-11.

> For performing the key-search within the database **14**, it is considered desirable to use an efficient approximate pattern matching algorithm. By "approximate" is meant that the algorithm should be able to take into account various forms of errors.

This passage does not state that all matches are not considered, or even that all data in all possible matches is not considered. Karypis Decl. ¶274. In particular, the reference to an approximate pattern matching algorithm does not disclose a non-exhaustive search.

13 You would agree that the term "approximate
14 matching melody" doesn't expressly say that we're
15 going to be using a search approach where we leave
16 out all -- some of the data; right?
17 A I agree.

Moulin Depo. 347:13-17; Karypis Decl. ¶274.

Passage 2: Ghias, 6:23-35.

> Several Algorithms have been developed that address the problem of approximate string matching. Running times have ranged from 0(mn) for the brute force algorithm to 0(kn) or 0(nlog(m), where "0" means "on the order of," m is the number of pitch differences in the query, and n is the size of the string (song). See Ricardo Baeza-Yates and G. H. Gonnet, "Fast String Matching with Mismatches," *Information and Computation,* 1992. A preferred algorithm which is considered to offer better performance in general for this purpose is that described in Ricardo A. Baesa-Yates and Chris H. Perieberg, "Fast and Practical Approximate String Matching, "*Combinatorial Pattern Matching, Third Annual Symposium,"* pages 185–192, 1992.

This passage discusses comparing the work with a single record in the database. Karypis Decl. ¶275. The approximate string matching algorithms involves matching a work with a record in the database that includes an "error" so that "various forms of errors" would not prevent a proper match from being identified. Karypis Decl. ¶275. The approximate string matching algorithm is applied when the work melody "is compared with all the songs" in the database and all of the data within each record. Ghias, 5:66-6:2; Moulin Depo. 347:13-17; Karypis Decl. ¶275. Accordingly, Ghias does not disclose a search that would even meet Petitioner's improper construction of non-exhaustive search, because Ghias does not search less than all possible matches or even not less than all data within all possible matches.

Board concerns: The following addresses the Board's specific concerns (identified in the Decision) with respect to whether Ghias discloses the claimed non-exhaustive search. In instituting Ground 2, the Board did not rely on the

arguments presented by Petitioner attempting to establish the claimed non-exhaustive search. Instead, the Board found that Ghias disclosed the non-exhaustive search because the search disclosed in Ghias could produce a list of matches based on an error-tolerance and the user can perform a "new query on a restricted search list consisting of songs just retrieved":

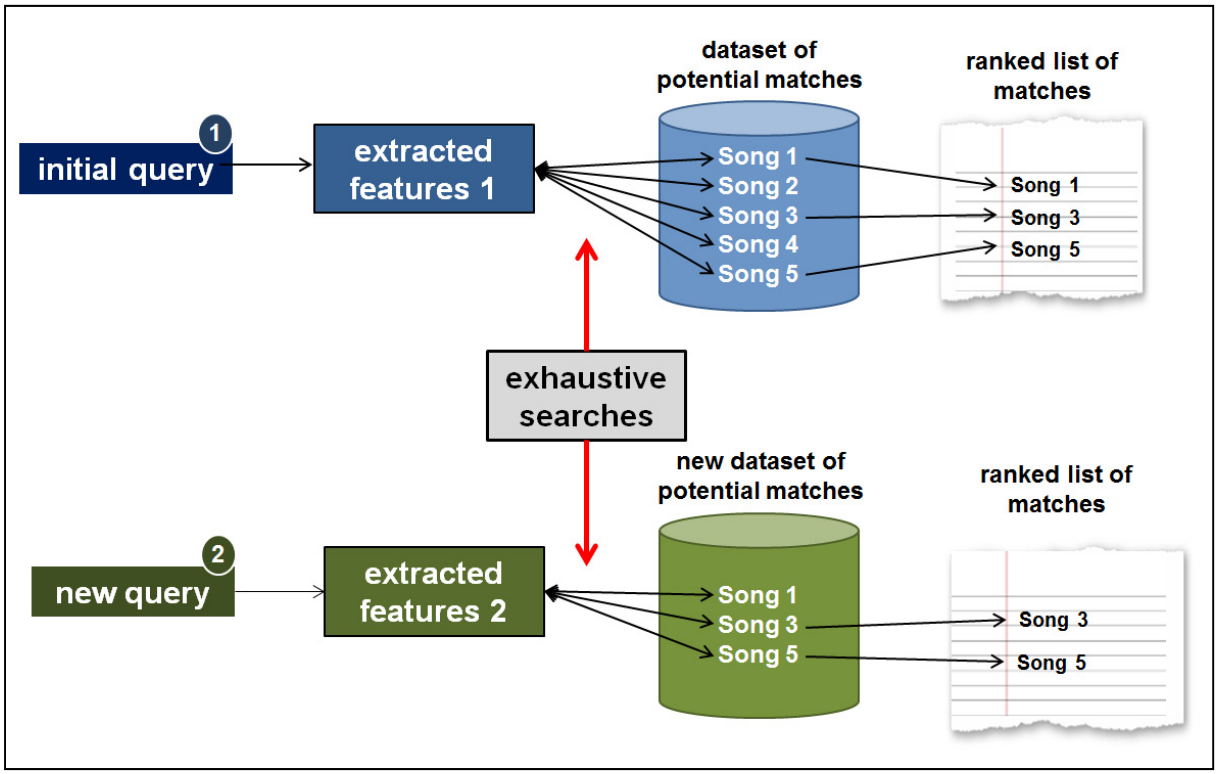
> arguments. Ghias provides that "[t]he number of matches that the database 14 should retrieve depends upon the *error-tolerance* used during the key-search." Ex. 1010, 6:63–65 (emphasis added). Ghias further provides that "the user can perform a *new query on a restricted search list consisting of songs just retrieved*. This allows the user to identify sets of songs that contain similar melodies." *Id.* at 7:5–8 (emphasis added). Thus, Ghias makes clear that the search need not be exhaustive, as Patent Owner argues, and will act to "identify[] a close, but not necessarily exact or closest, match," per our claim construction. Additionally, given the

Decision at 14. There are two reasons why the Board's reliance on the "new query on a restricted search list" does not satisfy Petitioner's burden of demonstrating that the instituted clams are unpatentable based on Ghias.

First, using a "new query" on the "restricted search list consisting of songs just received" does not disclose the claimed non-exhaustive search. A non-exhaustive search is "a search that locates a match without a comparison of all possible matches." *See* Section I(A)), Karypis Decl. ¶283. The restricted search

can be viewed in one of two ways.[6] Under either view, Ghias does not disclose a non-exhaustive search. Each view is addressed in turn.

First view: The second search could be viewed as a separate independent second search. As disclosed in Ghias, the second search is based on a "new query"—"the user can perform a new query on a restricted search list." Ghias, 7:4-8. The two searches, the first based on an initial query (1), and the second, based on a second "new query" (2), is reflected in this diagram:



To refine the list of potential matches, the "new query" (2) disclosed in Ghias must be different from the initial query (1). This is because Ghias does not teach an

[6] Ghias provides no details about the search on the restricted search list.

alternative search algorithm (different from the initial search algorithm) for searching the restricted list. Rather, Ghias teaches that the same search algorithm is applied to the new query (2) that was applied to the initial query (1). If the initial query (1) (rather than a new query (2)) is applied to the restricted list using the same search algorithm, the search would produce the same restricted list rather than refine the search as intended by Ghias.[7] Although the details of the new query (2) are not disclosed, the new query could theoretically be, *e.g.*, (a) a different portion of the song to be identified, or (b) a better hummed version of the same portion of the song to be identified.

If the second new query (2) is viewed as a second separate search, each search (1 and 2) would be exhaustive because (1) the initial search compares the query to all possible matches in the database—"all the songs" (Ghias, 5:66-6:2; Moulin Depo. 336:9-15), and (2) the restricted search also compares the query to all "possible matches." The records that are not on the restricted list (*i.e.*, in the blue dataset but not in the green dataset) are not possible matches. The first search excludes from the list of ranked songs those songs that are not possible matches.

[7] If the query remains constant (*i.e.*, the query is not changed) but a different algorithm is applied to the restricted list, this would constitute a single search with two stages because the query do not change.

The second search compares a new query to a new data set and compares that query to all record in the new data set—all possible matches. Moulin Depo. 336:13-327:6; 335:13-336:12.

Second view: Alternatively, under a second view, the search is viewed as a single search with two stages. Under this view, the second search on the restricted list is not an independent search. Rather, the search on the restricted list is the second stage of a two-stage search, dependent on the first stage. *See* Moulin Depo. 336:9-15. The second search depends on the first to generate a candidate set. A single work *i.e.*, the song to be identified (not two or more works) is being identified in the two-stage search. The two stages refine the identified matches; the second stage does not identify any new matches.

To constitute a non-exhaustive search under this view, the two-stage search process disclosed in Ghias would have to conduct the search without comparing the work to be identified with all possible matches in the dataset. The two-stage search disclosed in Ghias is exhaustive because the first stage compares the query to all possible matches in the dataset —"all the songs." Ghias, 5:66-6:2.

Q And the user -- the user search has already done a comparison to the other songs in the database; right?
A In the first stage, you know, the one that produced the list, the search was on the entire database presumably, yes.
Q So we have a two-stage search. The first stage, we do a comparison of all possible matches in the database; right?
A Yes.

Moulin 336:3-336:12.

Q Does it teach doing a search without doing a comparison of all the possible matches?
A You mean all the possible entries in the original database?
Q Yes.
A No, it does not teach that.
Q The entries in the original database, those are the possible matches?
A Yes.

Moulin Depo. 338:6-14. Under this view, the query on a restricted search list is part of a broader search of every record in the database, which compares the work to be identified with all possible matches—all records in the data set.

The only algorithm Ghias teaches for conducting a search is to compare a query statement against every record in the data set against which the algorithm is to be run—and is thus always an exhaustive search. Accordingly, whether the two-

stage search is viewed as a single search or two separate searches, the searches compare the work to be identified with "all possible matches" and are therefore exhaustive searches.

Second, because the passages cited by the Board do not establish the claimed non-exhaustive search, these passages were not identified by the Petitioner as support for the non-exhaustive search in the Petition or Declaration and should therefore not be a basis for finding the '441 claims unpatentable.

To establish a *prima facie* case of unpatentability, the Petition must specifically point to where in the reference the element is disclosed. 42 C.F.R. 42.104(b)(4) ("The petition must specify where each element of the claim is found in the prior art patent or printed publications relied upon."). It is not the Board's job to find where a reference teaches an element. *3M Company, Inc., v. Andover Healthcare, Inc.*, IPR2014-00630, Paper 11 at 5, (February 20, 2015) ("Petitioner fails to point out where the inherency argument was presented in the Petition."); *Purdue Pharma L.P., v. Depomed, Inc*., IPR2014-00378, Paper 18 at 5 (August 6, 2014) ("It is not the Board's role to play archeologist to uncover any additional support in the record that is not raised and discussed in the Petition.")

The only references to Ghias presented by the Petitioner for the claimed non-exhaustive search are Ghias, 6:7-11 and 6:23-35 addressing approximate string matching, not a performing a "new query on a restricted search list

consisting of songs just retrieved" based on an error tolerance. Petitioner's Declarant did not "cite anything in [his] Declaration that teaches, in Ghias, performing a search that returns a list of ranked matching songs and then performing a second search on that list." Moulin Depo. 146:21-147:21.

The Board, however, relied exclusively on two passages from Ghias not cited by Petitioner—Ghias, 6:63-65 and 7:5-8. These passages address a different concept that the approximate pattern matching concept identify by Petitioner as support for the non-exhaustive search element. Karypis Decl. ¶282. Accordingly, the Board's reasoning in its Decision should not be a basis for finding the instituted claims unpatentable because the instituted claims should not be found unpatentable based on arguments and passages from Ghias that were not identified in the Petition as support for this claim element.

B. neighbor search (claims 1, 13, 25).

In instituting this Ground, the Board did not specifically address whether Ghias discloses the claimed neighbor search. As explained above (Section I(B)), a neighbor search is "a search that identifies a close, but not necessarily exact or closest, match." *See* Section I(B); Decision at 6-7; Karypis Decl. ¶89. Ghias does not disclose a neighbor search because Ghias does not disclose identifying a close

match that is not necessarily the closest match. Karypis Decl. ¶¶429-430, 339-407, 236-245-.

To disclose a neighbor search, Ghias must disclose a search which does not necessarily find the closest match. *See* Section I(B). Karypis Decl. ¶¶399, 237. A search that always (necessarily) identifies an exact or the closest match is not a neighbor search because a neighbor search identifies a "close, but not necessarily exact or closest, match." Decision at 6-7; Karypis Decl. ¶¶399, 237. Ghias discloses a search algorithm which necessarily finds the closest match. Karypis Decl. ¶¶399, 238. Ghias does not expressly disclose a search that does not necessarily identify an exact or closest match. And such a search is not inherent (necessarily present) in Ghias. Karypis Decl. ¶238.

Ghias teaches a search that generates three possible outputs:

(1) an exact match (Ghias 2:53-59 ("exact matching melody"));

(2) a "ranked list of approximately matching melodies" (Ghias, 2:50-59; Ghias, 6:60-63 ("a list of songs ranked by how well they matched the query"); Moulin Depo. 118:9-22); or

(3) "the single most approximate matching melody" (Ghias, 2:50-59); Karypis Decl. ¶239.

Petitioner's Declarant confirmed that Ghias teaches these three potential outputs:

Would you agree that Ghias teaches sort of three options? We've got -- it's going to produce the exact match; it's going to produce the most approximate matching; or it's going to produce this ranked list?

A Yes.

Moulin Depo. 341:16-21.

For all three outputs, Ghias always identifies an exact or the closest match:

(1) exact match: If the search produces an exact match, it necessary produces an "exact or closest, match" and therefore does not disclose a neighbor search. Moulin Depo. 341:23-342:1; Karypis Decl. ¶241.

(2) ranked list: If the search produces a ranked list, it necessarily identifies as part of the ranked list either an exact match (if there is one) or the closest match—*i.e.*, the top ranked match and therefore does not disclose a neighbor search that does not necessarily identify an exact or the closest match. Karypis Decl. ¶242. At the top of the ranked list (*i.e.*, the number 1 ranked match in the list) is an exact or the closest match. Petitioner Declarant admitted that the ranked list approach identifies the closest match. Moulin Depo. 356: 2-21.

(3) single most approximate matching melody: If the search identifies the single most approximate matching melody, it necessarily identifies the closet match and is therefore not the claimed neighbor search. Moulin Depo. 345:16-346:11.

Petitioner's Declarant confirmed that for all three outputs, Ghias teaches a system that will always (necessarily) identify the closest match. Moulin Depo. 352:22-353:2. Accordingly, for all three potential outputs, Ghias necessarily identifies an exact or the closest match. Karypis Decl. ¶245. Ghias does not disclose a neighbor search which identifies "a close, but not necessarily exact or closest, match." Karypis Decl. ¶245.

The Petition, Declaration, and corresponding charts fail to demonstrate that Ghias teaches a neighbor search. As support for this element, the Petition, Declaration, and corresponding charts rely on the following:

> from 0(mn) for the brute force algorithm to O(kn) or O(nlog(m)"). This search locates a neighbor by determining "a ranked list of approximately matching melodies, as illustrated at 26" or "the single most approximate matching melody." *Id.* at 2:50-59, 6:60-63.

Pet. at 47-48. Neither of the cited passages from Ghias discloses the claimed neighbor search because, as described above, searches that produce either the ranked list or single most approximate matching melody always identify the closet match. Karypis Decl. ¶252. Each is addressed in turn:

Passage 1: Ghias, 2:50-59. This passage states that the search "outputs a ranked list of approximately matching melodies, as illustrated at 26" or "the single most approximate matching melody." As explained above, neither approach

discloses the claimed neighbor search. A neighbor search is "a search that identifies a close, but not necessarily exact or closest, match." Decision at 8. Karypis Decl. ¶253. Both of the two searches disclosed in this passage necessarily disclose an exact or the closest match and, therefore are not neighbor searches. Karypis Decl. ¶253.

Passage 2: Ghias, 6:60-63. This passage does not disclose a neighbor search. As explained above, the "list of songs ranked by how well they matched the query" necessarily identifies an exact or the closest match, and specifically identifies such song as the top-ranked song.

IV. Petitioner cannot supplement its analysis and evidence.

Apparently realizing that the Petition and Declaration fail to identify where key limitations of the '441 patent are found in the asserted art, Dr. Moulin suggested in his deposition that he might be able to come up with alternative undisclosed reasons why the asserted art teaches the missing claim elements. Dr. Moulin repeatedly suggested that he may try to "supplement" his opinions. Moulin Depo. 66:24-67:7; 103:23-104:12 ("I would like to supplement my – my opinion"); 84:22-85:17 ("I can supplement them"); 70:24-71:13 ("I supplemented my views now"); 293:3-10 ("I want to supplement my opinion if I did not write it down.").

Q You have mentioned several times in the course of my cross-examination of you that you've got some opinions that you would like to supplement. Have you prepared any supplemental opinions?
A You mean prepare in writing?
Q Yes.
A No, I have not written anything about that.
Q They're just in your head?
A Yes.

Moulin Depo. 376:11-21. Any attempt by Petitioner to rely on some previously unidentified disclosures in the asserted references to establish the claim elements beyond the specific analysis in the Petition and Declaration should be rejected.

First, the rules are clear: PTAB regulations require that new evidence or arguments submitted after the Petition—*i.e.*, with a reply brief or at oral argument—cannot be considered for purposes of making out a *prima facie* case of unpatentability. A Reply and Reply Declaration may only be considered to rebut arguments submitted by the patent holder. Rules of Practice for Trials Before the Patent Trial and Appeal Board and Judicial Review of Patent Trial and Appeal Board Decisions, Final Rule, 77 Fed. Reg. 48,612, 48,620 (Aug. 14, 2012) ("Reply evidence … must be responsive and not merely new evidence that could have been presented earlier to support the movant's motion."); Office Patent Trial Practice

Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012) ("[A] reply that raises a new issue or belatedly presents evidence will not be considered. … Examples of indications that a new issue has been raised in a reply include new evidence necessary to make out a *prima facie* case for the … unpatentability of an original… claim, and new evidence that could have been presented in a prior filing.").

Final Written Decisions issued by the Board consistently hold that Petitioners cannot rely on passages from a reference that were not presented and analyzed in an initial Petition:

> Petitioner did not present an analysis in the Petition that relies on paragraph 69 of Kamada as teaching [a limitation]. Instead, the Petition focused on the applicable of paragraphs 45 and 47 of Kamada to the limitation… Accordingly, Petitioner's argument that paragraph 69 teaches this limitation is a new argument, and we will not consider it for purposes of this Decision.

Eizo Corporation v. Barco N.V., IPR2014-00358, (citing Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756).

> we find that Petitioner's Reply exceeds the proper scope of a reply. As provided in 37 C.F.R. § 42.23(b), a "reply may only respond to arguments raised in the corresponding opposition or patent owner response." Thus, "a reply that raises a new issue or belatedly presents evidence will not be considered and may be returned. The Board will not attempt to sort proper from improper portions of the reply."

Intelligent Bio-Systems, Inc. v. Illumina Cambridge LTD, IPR2013-00517, Paper 87 at 14-16 (Feb. 11, 2015) (*quoting* Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012) (rejecting a "distinct new line of reasoning… changing the unpatentability rationale… as well as presenting new evidence to support the new rationale.").

> One indication that a new issue has been raised in a reply is where a petitioner submits "new evidence necessary to make out a *prima facie* case" of unpatentability of an original claim."

Intelligent Bio-Systms, Paper 87 at 14-16 (*quoting* Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012).

> none of the points raised by Dr. Zimmerman in his reply declaration … were made in his initial declaration … served with the Petition, or in the Petition itself. …. We have not considered it as part of Avaya's attempt to make out a *prima facie* case of unpatentability of the challenged claims."

Avaya Inc. v. Network-1, IPR2013-00071 Paper 103 at 28 (May 22, 2014) (*citing* Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012).

> None of the analysis in Dr. Wang's reply declaration, however, was included with Shaw's Petition. *See* 37 C.F.R. § 42.23(b); Rules of Practice for Trials Before the Patent Trial and Appeal Board and

> Judicial Review of Patent Trial and Appeal Board Decisions; Final Rule, 77 Fed. Reg. 48,612, 48,620 (Aug. 14, 2012) ("Oppositions and replies may rely upon appropriate evidence to support the positions asserted. Reply evidence, however, must be responsive and not merely new evidence that could have been presented earlier to support the movant's motion."); Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012) ("While replies can help crystalize issues for decision, a reply that raises a new issue or belatedly presents evidence will not be considered and may be returned. . . . Examples of indications that a new issue has been raised in a reply include new evidence necessary to make out a *prima facie* case for the patentability or unpatentability of an original or proposed substitute claim, and new evidence that could have been presented in a prior filing.").

Shaw Industries Group, Inc., v. Automated Creel Systems, Inc., IPR2013-00584, Paper 42 at 22-23 (July 24, 2014).

> We will not consider a new argument for unpatentability raised for the first time in a Reply. *See* 37 C.F.R. § 42.23(b). Likewise, we will not consider new evidence submitted for the first time in a Reply. *Id., see also* Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug.14, 2012) (A Reply that belatedly presents new evidence will not be considered).

Greene's Energy Group, LLC v. Oil States Energy Services, LLC, IPR2014-00364, Paper 37 at 30 (May 1, 2015).

> we note that Toyota introduces this argument for the first time in its Reply, along with a supporting Reply Declaration …. A reply may only respond to arguments raised in the patent owner response. 37 C.F.R. § 42.23(b). Furthermore, "a reply that raises a new issue or belatedly presents evidence will not be considered." Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 12, 2014).

Toyota Motor Corporation v. American Vehicular Sciences LLC, IPR2013-00419, Paper 59 at 15 (January 12, 2015).

<u>*Second*</u>, this Response and corresponding Declaration is the only opportunity for Patent Owner and its declarant, Dr. Karypis, to effectively address Petitioner's arguments and provide responsive evidence. The Patent Owner would be prejudiced if Petitioner presents new evidence and cites to alternative passages from the asserted art in its Reply and Reply Declaration when the controlling rules do not provide Patent Owner an opportunity to effectively respond to such new arguments:

> "this argument is newly raised in the Reply, and is based on a portion of Yanagawa that TRW alleges, for the first time, corresponds to the *enhanced* limitation. Although the argument is based on Magna's proposed claim construction rather than the construction that TRW

> proposes in its Reply, we consider the dispute over the proper construction of this term to be sufficiently foreseeable so that TRW should have presented the argument and evidence in the Petition. TRW's failure to do so at that time has deprived Magna of the opportunity to respond to it, as Magna could not respond to TRW's Reply. For this reason, we decline to consider it. *See* Office Patent Trial Practice Guide, 77 Fed. Reg. 48,756, 48,767 (Aug. 14, 2012) (stating that a reply that belatedly presents evidence will not be considered, and that one indication that a new issue has been raised in a reply is where the petitioner submits 'new evidence necessary to make out a *prima facie* case' of unpatentability of an original claim)."

TRW Automotive v. Magna Electronics, IPR2014-00262, Paper 37 at 12-13 (June 25, 2015).

Respectfully submitted,

AMSTER, ROTHSTEIN & EBENSTEIN LLP
Attorneys for Patent Owner
90 Park Avenue
New York, NY 10016
(212) 336-8000

Dated: September 18, 2015

By: */Charles R. Macedo/*
Charles R. Macedo
Registration No. 32,781

Gregory Dovel (admitted *pro hac vice*)
Dovel & Luner, LLP
201 Santa Monica Blvd., Suite 600
Santa Monica, CA 90401
(310) 656-7066

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. § 42.6(e), the undersigned hereby certifies that on this 18th day of September, 2015, a copy of the foregoing PATENT OWNER'S RESPONSE PURSUANT TO 37 C.F.R. § 42,120, together with Patent Owner's Exhibit List and Exhibit Nos. 2005-2010, was served via e-mail on the counsel of record for Petitioner at the following e-mail addresses:

DLGOOGLE1N@skadden.com
James J. Elacqua (James.Elacqua@skadden.com)
Douglas R. Nemec (Douglas.Nemec@skadden.com)

Dated: September 18, 2015

By: */Charles R. Macedo/*
Charles R. Macedo
Registration No. 32,781